# Análise de dados utilizando a medida de tempo de consenso em redes complexas

***Jean Pierre Huertas Lopez***

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura:_____

# Análise de dados utilizando a medida de tempo de consenso em redes complexas

***Jean Pierre Huertas Lopez***

**Orientador:** ***Prof. Dr. Zhao Liang***

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências - Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*.

USP – São Carlos
Abril/2011

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados fornecidos pelo(a) autor(a)

H887 Huertas Lopez, Jean Pierre
/ Jean Pierre Huertas Lopez; orientador Liang Zhao -- São Carlos, 2011.
74 p.

Dissertação (Mestrado - Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2011.

1. Análise de Dados. 2. Redes Complexas. 3. Consenso. I. Zhao, Liang, orient. II. Título.

Aos meus pais,
Carlos e Lourdes.

# Agradecimentos

Existem muitas pessoas as quais queremos e precisamos agradecer, pois a conclusão de um trabalho só é possível com a ajuda, suporte e orientação direta o indireta delas.

Agradeço primeiramente a Deus, por ter colocado no meu caminho cada uma das pessoas que contribuíram na minha formação acadêmica e pessoal, e pela serenidade e fortaleza proporcionadas em todo momento.

Aos meus pais, Lourdes e Carlos por todo amor, incentivo e apoio brindado apesar da distância. Eles sempre foram meus guias, meu modelo a seguir em todo momento da vida.

Ao meu orientador, professor e amigo Zhao Liang, primeiramente por ter aceitado me orientar, por todo o apoio concedido, a motivação transmitida, e o conhecimento brindado que me guiou nesses dois anos de trabalho.

A minha amiga, companheira e amor, Flor, pela compreensão e alegria transmitida apesar da distância.

Aos meus amigos e colegas de estudo em São Carlos. Especialmente ao Bilzã e à Lilian, parceiros de pesquisa com os quais foram discutidas as inquietudes que fortaleceram os nossos trabalhos.

Aos professores e funcionários do ICMC-USP. Especialmente ao Prof. Ricardo Campello, Prof. Eduardo Hruschka e ao Prof. Thiago Pardo, pelo atenção prestada e conhecimentos transmitidos.

Finalmente, gostaria de agradecer à FAPESP pelo apoio financeiro concedido.

# Resumo

Redes são representações poderosas para muitos sistemas complexos, onde vértices representam elementos do sistema e arestas representam conexões entre eles. Redes Complexas podem ser definidas como grafos de grande escala que possuem distribuição não trivial de conexões. Um tópico importante em redes complexas é a detecção de comunidades. Embora a detecção de comunidades tenha revelado bons resultados na análise de agrupamento de dados com grupos de diversos formatos, existem ainda algumas dificuldades na representação em rede de um conjunto de dados. Outro tópico recente é a caracterização de simplicidade em redes complexas. Existem poucos trabalhos nessa área, no entanto, o tema tem muita relevância, pois permite analisar a simplicidade da estrutura de conexões de uma região de vértices, ou de toda a rede. Além disso, mediante a análise de simplicidade de redes dinâmicas no tempo, é possível conhecer como vem se comportando a evolução da rede em termos de simplicidade. Considerando a rede como um sistema dinâmico de agentes acoplados, foi proposto neste trabalho uma medida de distância baseada no tempo de consenso na presença de um líder em uma rede acoplada. Utilizando essa medida de distância, foi proposto um método de detecção de comunidades para análise de agrupamento de dados, e um método de análise de simplicidade em redes complexas. Além disso, foi proposto uma técnica de construção de redes esparsas para agrupamento de dados. Os métodos têm sido testados com dados artificiais e reais, obtendo resultados promissores.

# Abstract

Networks are powerful representations for many complex systems, where nodes represent elements of the system and edges represent connections between them. Complex networks can be defined as graphs with no trivial distribution of connections. An important topic in complex networks is the community detection. Although the community detection have reported good results in the data clustering analysis with groups of different formats, there are still some difficulties in the representation of a data set as a network. Another recent topic is the characterization of simplicity in complex networks. There are few studies reported in this area, however, the topic has much relevance, since it allows analyzing the simplicity of the structure of connections between nodes of a region or connections of the entire network. Moreover, by analyzing simplicity of dynamic networks in time, it is possible to know the behavior in the network evolution in terms of simplicity. Considering the network as a coupled dynamic system of agents, we proposed a distance measure based on the consensus time in the presence of a leader in a coupled network. Using this distance measure, we proposed a method for detecting communities to analyze data clustering, and a method for simplicity analysis in complex networks. Furthermore, we propose a technique to build sparse networks for data clustering. The methods have been tested with artificial and real data, obtaining promising results.

# Conteúdo

# Lista de Figuras

Capítulo

1

# Introdução

As Redes têm sido amplamente utilizadas no modelamento de sistemas complexos. Uma das áreas onde as redes têm sido utilizadas com grande sucesso é na análise de dados, pois as redes podem revelar informações de estrutura topológicas dos dados. Um tema recente que vem sendo amplamente estudado, com muitos resultados são as Redes Complexas, as quais podem ser definidas como grafos de grande escala que possuem distribuição não trivial de conexões. Uma propriedade saliente das redes complexas é a "estrutura de comunidades". Cada comunidade é um grupo de vértices densamente conectados e ao mesmo tempo, as conexões entre diferentes grupos são relativamente esparsas. Vários algoritmos têm sido desenvolvidos para encontrar as comunidades em uma rede complexa.

Uma das técnicas bem conhecidas é baseada na remoção iterativa de arestas com um valor alto de uma medida chamada *betweenness* (Newman & Girvan, 2004), construindo dessa maneira uma árvore hierárquica divisiva de comunidades. Outra técnica de detecção de comunidades é baseada em inteligência coletiva proposta em (de Oliveira et al., 2008). Um método de detecção de comunidades que utiliza a idéia de sincronização de osciladores acoplados via controle focalizado (*pinning control*) é apresentado em (Li et al., 2008), onde cada comunidade é sincronizado para um estado comum no espaço de fase, aplicando controle sobre alguns vértices na rede. Em (Quiles et al., 2008), os autores propuseram um método baseado em competição de partículas, nesse trabalho as partículas percorrem a rede e competem entre elas, de maneira que cada partícula tenta dominar tantos vértices como seja possível, no final, cada partícula domina uma comunidade na rede. Dois métodos de detecção de comunidades baseados em uma medida de caminhada aleatória foram propostos em (Zhou, 2003a,b), nesses trabalhos os autores apresentaram uma medida de distância para redes complexas baseada em uma caminhada aleatória de uma partícula *Browniana* na rede, para depois aplicar essa medida para detectar comu-

nidades numa rede complexa.

Durante as décadas passadas, o comportamento de um sistema dinâmico de uma rede de agentes acoplados (por exemplo osciladores) têm sido amplamente estudado (Li et al., 2010; Gu et al., 2010; Olfati-saber et al., 2007; Chen et al., 2009), especificamente dois conceitos importantes nesse tipo de redes: o problema de Consenso e Sincronização, onde o estado de todos os agentes no espaço de fase são assintoticamente iguais.

O problema de consenso em uma rede de agentes acoplados é atingir um estado final comum para todos os agentes. Existem diferentes tipos de consenso, o mais popular é o "Consenso Médio", onde o estado final de todo agente é igual à média dos estados iniciais dos agentes na rede. Sincronização em osciladores acoplados é fortemente relacionada ao problema de consenso. Uma estratégia importante, chamada "Controle Focalizado"(*pinning control*), consiste em aplicar controle sobre uma parte dos vértices na rede para atingir a sincronização dos vértices para um estado homogêneo desejado, que pode ser, um ponto de equilíbrio, uma órbita periódica, ou outros tipos de soluções do sistema dinâmico. Muitos resultados nesse tópico têm sido obtidos (Xiang et al., 2007; Porfiri & Fiorilli, 2009; Wang & Chen, 2002). Em (Chen et al., 2007), o autor provou que um solo vértice controlado pode fixar uma rede acoplada para uma solução comum.

Uma segunda área contemplada neste trabalho é o Agrupamento de Dados, que é um processo que consiste em particionar um conjunto de dados em diferentes grupos, de tal forma que a similaridade entre objetos de um mesmo grupo é maximizado e a similaridade entre grupos é minimizado (Duda et al., 2001). As aplicações de agrupamento de dados incluem a caracterização de grupos de clientes baseada em padrões de compra, a categorização de documentos da Web, o agrupamento de genes e proteínas que possuem funcionalidades similares, a aprendizagem de máquina, a mineração de dados, a análise de imagens, bem como a classificação de regiões no globo que possuem maior chance de ocorrência de terremotos baseada em dados sismológicos, entre outras (Kogan et al., 2006) (Duda et al., 2001) (Jain et al., 1999).

A maioria dos algoritmos de agrupamento de dados baseia-se em modelos estáticos tais como $K$-Médias (Jain et al., 1999), Clarans (Ng & Han, 2002), DBSCAN (Kryszkiewicz & Skonieczny, 2005), CURE (Guha et al., 1998) e ROCK (Guha et al., 2000) são muito eficientes em alguns casos, embora não sejam adequados para a resolução de todos os casos. Um exemplo em que tais algoritmos não são adequados é quando os dados contém grupos de diversas formas, densidades e tamanhos, em virtude de não se basearem na natureza dos grupos individuais a fusão entre eles acontece. Por exemplo, técnicas de agrupamento de dados baseadas em partição, tais como $K$-Médias e Clarans, tentam particionar o conjunto de dados em $N$ grupos, de forma a otimizar um critério previamente estabelecido. Nesses algoritmos é assumido que os grupos têm a forma de hiper-esféricos com tamanhos similares. De forma que fica impossível evidenciar grupos que variam em tamanho, como mostrado na Figura 1.1(a), ou grupos com formatos irregulares, como mostrado na Figura 1.1(b).

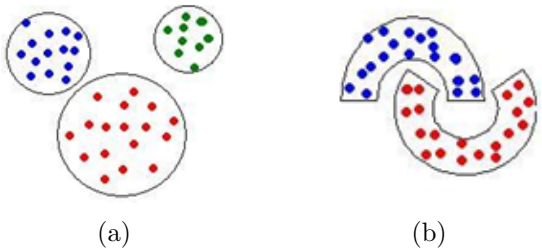



Figura 1.1: Grupos com formatos diferentes. (a)Grupos de diferente tamanho e (b)Irregulares.

Algoritmos de agrupamento de dados baseados em grafos são capazes de detectar grupos com formatos arbitrários. Esses algoritmos consistem em duas etapas. A primeira etapa é criar um grafo a partir do conjunto de dados, para depois na segunda etapa aplicar algum algoritmo que particione o grafo em grupos. Um algoritmo representativo de agrupamento de dados baseado em grafo é o CHAMELEON (Karypis et al., 1999), que utiliza o conceito de $K$-vizinhos mais próximos ($K$-nearest neighbour - KNN) sobre um grafo esparso que representa o conjunto de dados, no qual cada vértice refere-se a um ítem de dados e uma aresta entre dois vértices é um dos $K$ dados mais similares de outro vértice. O CHAMELEON realiza o processo de agrupamento de dados em duas etapas. Durante a primeira fase, o CHAMELEON usa um algoritmo de particionamento de grafos para agrupar os ítens de dados em um grande número de sub-grupos relativamente pequenos. Na segunda fase, é usado um algoritmo de agrupamento hierárquico aglomerativo para encontrar os grupos genuínos fundindo repetitivamente esses sub-grupos.

Em virtude disto, o algoritmo CHAMELEON busca a estrutura topológica do conjunto de dados de entrada. Conseqüentemente, é capaz de identificar em certo nível formas de grupos variados. No entanto, a formação da rede utilizando a técnica de $K$-vizinhos que é usualmente utilizado nos algoritmos de agrupamento de dados baseados em grafos apresenta dois problemas: 1) A rede resultante não será necessariamente conexa; e ainda pior, 2) a rede resultante pode ser densamente conectada dificultando a detecção dos grupos no grafo.

Este trabalho apresenta um algoritmo de agrupamento de dados baseado em grafo utilizando uma medida de distância de tempo de consenso proposta e um algoritmo de formação de redes a partir de um conjunto de dados, tratando os problemas mencionados acima e obtendo bons resultados.

O último assunto tratado neste trabalho é a caracterização de simplicidade em redes complexas. Esse tópico é novo e existem muito poucos trabalhos relacionados à detecção de regiões simples em redes complexas, no entanto, o tema tem muita relevância, pois permite analisar a complexidade ou simplicidade da topologia de uma região de vértices em uma rede, ou da rede toda (da F. Costa & Rodrigues, 2009). Além disso, mediante a análise de simplicidade de redes dinâmicas no tempo é possível conhecer se uma rede vem

se tornando mais simples, ou encontrar regiões com algum padrão de conexão que podem representar alguma informação importante da rede.

## 1.1 Objetivos

O presente trabalho tem como principal objetivo explorar o tema de análise de dados utilizando redes complexas. Entre os tópicos a serem tratados estão o agrupamento de dados e a análise de simplicidade em redes complexas. Um dos objetivos no tópico de agrupamento de dados é o desenvolvimento de uma nova técnica de agrupamento de dados baseada em detecção de comunidades que seja capaz de lidar com grupos de diversas densidades, tamanhos e formas. Por tal motivo foi proposto uma nova medida de distância em redes complexas baseada no tempo de consenso dos vértices em um sistema dinâmico acoplado, além de uma técnica de construção de redes a partir de um conjunto de dados que permita obter redes mais esparsas e com estrutura de comunidade mais bem definida.

No tópico de análise de simplicidade em redes complexas, o objetivo é o desenvolvimento de uma técnica para detectar as regiões de vértices com estrutura de conexões mais simples ou homogêneas de uma rede complexa. Assim, foi proposto um método de detecção de regiões simples em redes complexas baseado na medida de distância proposta, que além de detectar esse tipo de regiões na rede é capaz de tolerar alguns vértices e conexões que não pertencem à região homogênea (atuando como ruido na região homogênea) para encontrar os vértices que de fato pertencem a uma região simples.

## 1.2 Organização do documento

O presente trabalho mostra a medida de distância baseada no tempo de consenso em redes complexas e duas aplicações em análise de dados: agrupamento de dados via detecção de comunidades e a detecção de regiões homogêneas em redes complexas. Por tal motivo, no Capítulo 2 são apresentadas revisões bibliográficas sobre redes complexas e estrutura de comunidades, além de alguns métodos representativos de detecção de comunidades e alguns conceitos de sistemas acoplados dinâmicos, como consenso e controle focalizado em redes complexas. O Capítulo 3 apresenta uma revisão de agrupamento de dados, além de algumas das técnicas de agrupamento de dados mais conhecidas, e alguns trabalhos relevantes que abordam o problema de agrupamento de dados utilizando redes complexas. O Capítulo 4 apresenta a medida de distância proposta, assim como um método de detecção de comunidades baseado nessa distância, além disso, é apresentado também um método de agrupamento de dados e um método de detecção de regiões simples em redes complexas baseados na medida proposta. Por fim, no Capítulo 5 são apresentadas as conclusões mais relevantes e as perspectivas de trabalhos futuros.

Capítulo

2

# Redes complexas e detecção de comunidades

Neste capítulo serão apresentados conceitos básicos de redes complexas, assim como algumas características das redes complexas. Além disso, serão apresentadas alguns conceitos de estrutura de comunidade e modularidade em redes complexas, os quais são fortemente ligados com agrupamento de dados, e algumas técnicas de detecção de comunidades.

## 2.1 Redes complexas

Uma rede complexa pode ser definida como uma rede cuja estrutura não segue um padrão regular (Newman, 2003). Exemplos de redes complexas podem ser: redes neurais biológicas (Sponrs, 2002), distribuição de energia elétrica (Albert et al., 2004), a Internet (Faloutsos M., 1999).

Em redes relativamente pequenas é possível calcular características de maneira exata, usando a teoria de grafos, mas em sistemas mais complexos com grande quantidade de vértices, o cálculo dessas características não é uma tarefa trivial, assim medidas estatísticas são necessárias para a melhor caracterização dessas redes. A nomenclatura de redes complexas vem do campo da Física, onde este modelo tem sido mais estudado.

Um exemplo interessante de redes complexas, são as redes sociais, onde se tem um grande número de vértices e a distribuição dos graus não segue um padrão regular, além disso, estas redes geralmente apresentam grupos de vértices, famílias, cidades, esses grupos são chamados de comunidades. A Figura 2.1 mostra uma rede de interações entre os atores principais da obra de Victor Hugo *Les Miserables*, a rede apresenta estrutura de comunidades.

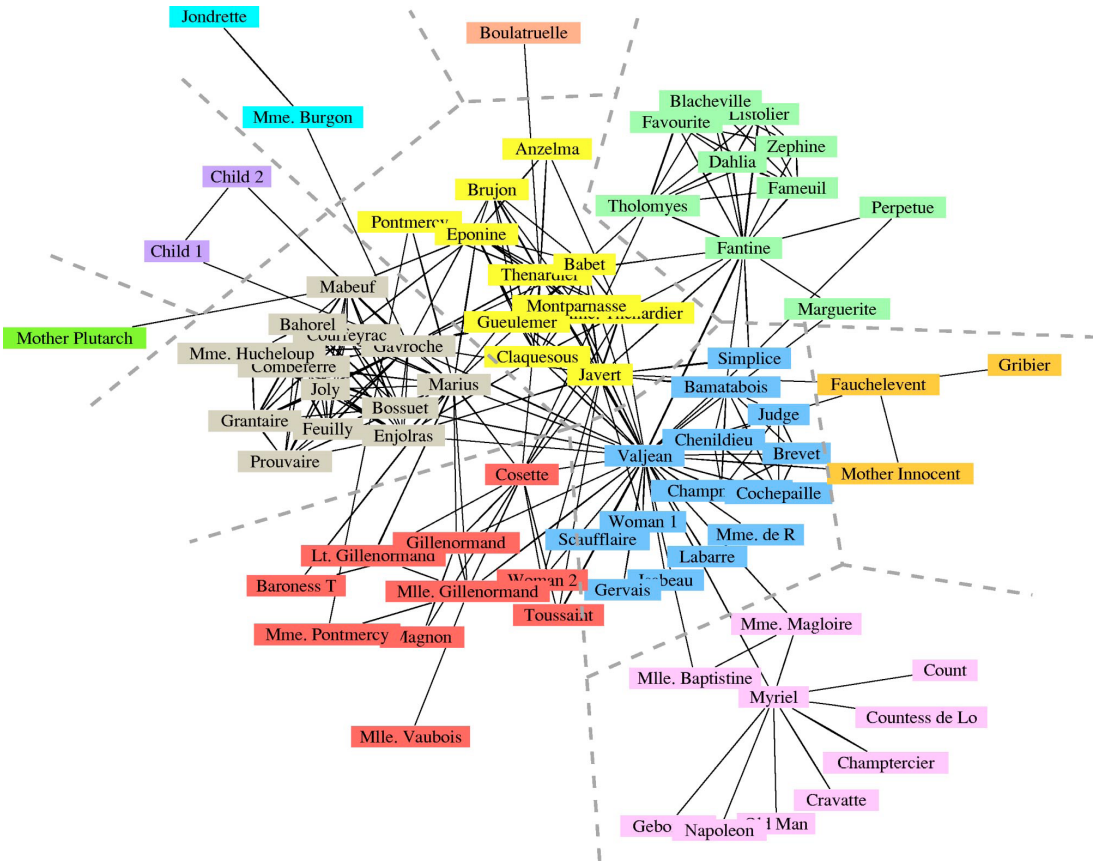



Figura 2.1: Rede de interações entre os atores mais importantes da novela *Les Miserables* de Victor Hugo. A rede possui uma separação bem definida em grupos ou comunidades. Figura obtida de (Newman & Girvan, 2004)

## 2.2 Modelos de redes complexas

Nas últimas décadas têm sido desenvolvidos diversos modelos de redes. Erdös & Rényi propuseram em (Erdös & Rényi, 1959) o modelo de rede aleatória, que é uma rede onde as arestas são criadas de maneira aleatória. Nesse tipo de redes a maioria de vértices possui um mesmo numero de arestas, assim a distribuição dos graus nas redes aleatórias segue uma distribuição binomial. Esses tipos de modelos são geralmente utilizados como redes artificiais, pois não representa muitas características das redes reais. A Figura 2.2(a) mostra um modelo de rede aleatória.

Em 1998 foi apresentado um modelo de rede que representava melhor as características das redes reais. Uma propriedade importante desse modelo é a propriedade de pequeno mundo, propriedade que diz que a média da distância geodésica (caminho mínimo) entre todos os vértices é pequena. Em outras palavras, é muito provável encontrar um caminho curto entre dois vértices. Esse modelo foi chamado de rede de pequeno mundo (Watts & Strogatz, 1998a) mostrado na Figura 2.2(b).

Outro modelo é o modelo de redes livres de escala (Albert et al., 1999), onde notaram que a distribuição dos graus da *world wide web* segue a lei de potência descrita na Equação (2.1).

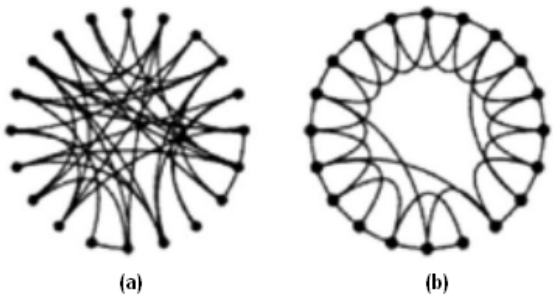



Figura 2.2: Modelos de redes. (a) Rede aleatória e (b) Rede de pequeno mundo. Figura obtida e adaptada de (Watts & Strogatz, 1998a)

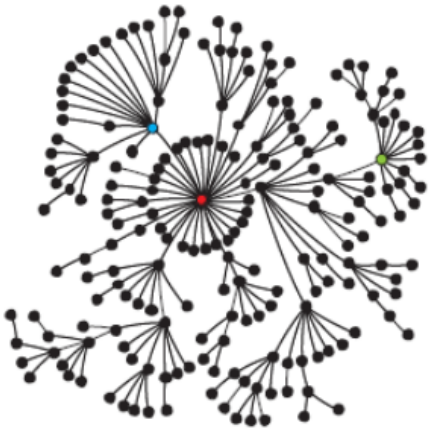


Figura 2.3: Rede livre de escala. Note-se a presença de vértices com grau alto chamados *hubs*. Figura obtida de (Strogatz, 2001)

$$p(k) \sim k^{-\lambda}, \tag{2.1}$$

onde $\lambda$ é o expoente da escala, $k$ é o grau do vértice. Segundo a lei de potência a maioria dos vértices da rede tem grau baixo, enquanto poucos vértices na rede tem grau alto, chamados *hubs*. A Figura 2.3 mostra uma rede livre de escala.

## 2.3 Propriedades das redes complexas

Nesta seção são apresentadas algumas propriedades importantes das redes complexas, essas propriedades consistem em medidas estatísticas que revelam alguma característica particular da rede. Considere o grafo não direcionado $G = (V, E)$, onde $V$ é o conjunto de $n$ vértices do grafo e $E$ representa o conjunto de $m$ arestas entre os vértices de $G$. Uma forma de representar um grafo $G$ é utilizando uma matriz de adjacência $A$ com elementos $a_{ij}$, sendo que $a_{ij} \neq 0$ representa uma aresta entre os vértices $i$ e $j$ que pertencem a $V$. O grau $k_i$ de cada vértice $i$ é definido pelo número de vizinhos de $i$ como mostrado na Equação (2.2).

$$k_i = \sum_j a_{ij}. \tag{2.2}$$

Assim o grau médio da rede é dado por $\langle K \rangle = \frac{1}{n}\sum_i k_i$, sendo $n$ o número de vértices na rede.

### 2.3.1 Coeficiente de agrupamento

O coeficiente de agrupamento mede a presença de ciclos de ordem três na rede (Newman, 2003), a Equação (2.3) define o coeficiente de agrupamento.

$$C = \frac{3N_\triangle}{N_3}, \tag{2.3}$$

onde $N_\triangle$ é o número de triângulos na rede e $N_3$ é o número de triplas conectadas. Um triângulo é um conjunto de três vértices com arestas entre cada par de vértices, e uma tripla conectada é um conjunto de três vértices onde cada vértice pode ser alcançado por qualquer outro. Assim o número de triângulos e o número de triplas conectadas podem ser calculados como:

$$N_\triangle = \sum_{k>j>i} a_{ij}a_{ik}a_{jk}, \tag{2.4}$$

$$N_3 = \sum_{k>j>i} (a_{ij}a_{ik} + a_{ji}a_{jk} + a_{ki}a_{kj}), \tag{2.5}$$

onde $i, j, k$ são vértices da rede e o somatório é realizado sobre todas as triplas de vértices diferentes $i, j, k$.

Outra definição de coeficiente de agrupamento (Watts & Strogatz, 1998a) é dada por:

$$C_i = \frac{3N_\triangle(i)}{N_3(i)}, \tag{2.6}$$

onde $N_\triangle(i)$ é o número de triângulos que contem o vértice $i$ e $N_3(i)$ é o número de triplas conectadas que tem como vértice central o vértice $i$. $N_\triangle(i)$ e $N_3(i)$ são definidos como:

$$N_\triangle(i) = \sum_{k>j} a_{ij}a_{ik}a_{jk}, \tag{2.7}$$

$$N_3(i) = \sum_{k>j} a_{ij}a_{ik}. \tag{2.8}$$

Logo o coeficiente de agrupamento nesta outra abordagem é definido como:

$$C = \frac{1}{n}\sum_i C_i, \tag{2.9}$$

onde $n$ é o numero de vértice e $C_i$ é o coeficiente de agrupamento do vértice $i$.

### 2.3.2 Distribuição de grau

O grau de um vértice é uma característica importante individual (Newman, 2003), mas a distribuição do grau é uma característica importante global da rede. A distribuição do grau $P(k)$ é a fração de vértices que possuem grau $k$, ou expressado de outra forma, a probabilidade de que um vértice escolhido aleatoriamente na rede seja de grau $k$. Geralmente a distribuição cumulativa do grau pode dar uma idéia da conectividade da rede, assim as rede aleatórias geralmente possuem uma distribuição Binomial, e uma distribuição de Poisson no limite do número de vértices, nas redes regulares é evidenciado uma distribuição de Poisson (Newman, 2003), entanto redes livres de escala seguem a lei de potência. A Equação (2.10) mostra uma definição da função de distribuição cumulativa.

$$P(k) = \sum_{k'=k}^{\infty} P(k'), \tag{2.10}$$

onde $P(k)$ representa a probabilidade de que o grau escolhido aleatoriamente seja maior ou igual que $k$.

### 2.3.3 Entropia da distribuição de grau

A entropia da distribuição de grau mede a heterogeneidade da rede, essa medida é aplicada para conhecer quão robusta é a rede (Luciano et al., 2006). A entropia da distribuição de grau é definida na Equação (2.11).

$$H = -\sum_{k} P(k) \log P(k). \tag{2.11}$$

O valor máximo para essa medida é dado para uma distribuição de grau uniforme, entanto o valor mínimo $H = 0$ é atingido quando todos os vértices possuem grau igual na rede.

### 2.3.4 Centralidade

Centralidade é uma medida que tenta medir quão importante é um vértice na rede (Luciano et al., 2006). Se muitos caminhos passam por um vértice ou aresta, essa aresta ou vértice é considerado mais importante. Considerando que as interações entre os vértices seguem sempre o caminho mais curto, então a importância de um vértice ou aresta é dada pela centralidade *betweenness* definida na Equação (2.12).

$$B_u = \sum_{ij} \frac{\sigma(i,u,j)}{\sigma(i,j)}, \tag{2.12}$$

onde $\sigma(i,u,j)$ é o número de caminhos mínimos desde o vértice $i$ até $j$ passando pelo vértice ou aresta $u$, $\sigma(i,j)$ é o número total de caminhos mínimos entre o vértice $i$ e $j$.

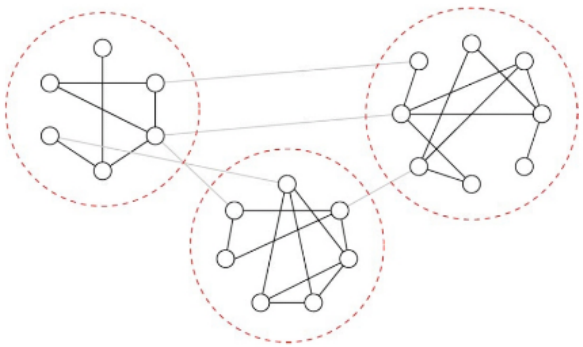


Figura 2.4: Estrutura de comunidades. Os vértices das comunidades estão densamente conectados, enquanto vértices de diferentes comunidades estão pouco conectados. Figura obtida de (Newman & Girvan, 2004).

## 2.4 Estrutura de comunidade

As redes na natureza possuem muitas características de estrutura, essa estrutura corresponde geralmente a entidades que compartilham funções. Uma comunidade é um subconjunto densamente conectado de vértices que é pouco conectado aos demais vértices na rede (Newman, 2004a). Em redes biológicas, comunidades correspondem a módulos com funções definidas, onde as células do modulo colaboram com algum processo importante. Outro exemplo, no caso das redes criadas pelo homem, como a *world wide web*, páginas com temas semelhantes tendem a estar bem conectadas, enquanto páginas com temas diferentes tendem a estar pouco ligadas (Flake et al., 2002).

O estudo da estrutura de comunidades em redes complexas é importante em diversos campos. Na Bioinformática, onde as comunidades são conjuntos de genes ou proteínas que participam em algum processo biológicos, é possível classificar proteínas ou genes com funções desconhecidas determinando a comunidade à que pertencem.

Uma forma de medir a qualidade das comunidades encontradas por um algoritmo de detecção de comunidades é mediante a modularidade $Q$ (Newman & Girvan, 2004). A modularidade é proporcional à subtração entre o número de arestas entre os vértices de uma mesma comunidade e o número de arestas entre distintas comunidades. Considerando uma partição da rede em $l$ comunidades e uma matriz simétrica $C_{l\times l}$ tal que o elemento $c_{ij}$ é a fração de arestas que conectam vértices da comunidade $i$ com a comunidade $j$, assim o traço da matriz $C_{l\times l}$ representa a fração de arestas que conectam vértices da mesma comunidade. A modularidade é definida na Equação (2.13).

$$Q = \sum_i (c_{ii} - a_i^2), \tag{2.13}$$

onde $a_i = \sum_j c_{ij}$ que representa a fração de arestas que estão conectadas à comunidade $i$. $Q$ tem um valor próximo a 0 quando não existe estrutura de comunidade nessa partição da rede, e tem um valor próximo a 1 quando as comunidades estão bem definidas pela

partição da rede.

Existem várias técnicas de detecção de comunidades na literatura, uma técnica bastante utilizada é mediante a otimização da função de modularidade (Newman & Girvan, 2004), descrita na Equação (2.13), outras são baseadas em caminhadas aleatórias na rede (Zhou, 2003b), outros métodos são baseados na teoria da informação (Rosvall & Bergstrom, 2007), *swarm aggregation* ou inteligência coletiva (de Oliveira et al., 2008), competição de partículas (Quiles et al., 2008).

A seguir são apresentados alguns métodos de detecção de comunidades.

### 2.4.1 Método baseado em modularidade

Baseado na medida de modularidade da Equação (2.13) foi proposto um método aglomerativo de detecção de comunidades (Newman, 2004b). Assim o método junta duas comunidades quando estas levam ao maior aumento da modularidade, essa técnica começa com cada vértice sendo seu próprio grupo e vai juntando os grupos por pares. Trata-se de um algoritmo de otimização gulosa da modularidade $Q$.

A variação da modularidade quando duas comunidades são agrupadas é dada pela Equação (2.14).

$$\triangle Q = e_{ij} + e_{ji} - 2a_i a_j, \tag{2.14}$$

onde $e_{ij}$ é a fração de arestas entre vértices da comunidade $i$ até $j$, $a_i$ é a fração de arestas que incidem sobre vértices da comunidade $i$. No começo, na primeira iteração, o valor de $e_{ij}$ é igual à metade do grau de cada vértice, pois cada um deles forma seu próprio grupo. Na última iteração, todos os vértices formam um grupo só, a melhor partição será aquela que possua o maior valor da modularidade $Q$.

### 2.4.2 Método baseado em betweenness

O método baseado em *betweenness* é fundamentado na idéia da separabilidade das comunidades, se a fração de arestas que ligam comunidades, for bem menor do que a fração que ligam vértices da mesma comunidade, então, todos os caminhos que vão de um vértice de uma comunidade, para um vértice de outra comunidade, devem passar por aquela pequena fração de arestas que ligam as duas comunidades. Portanto as ligações entre as comunidades têm valor de *betweenness* alto. A idéia do algoritmo é ir removendo arestas com maior valor de *betweenness* para encontrar as comunidades, onde *betweenness* é uma medida que tenta identificar as arestas que ligam duas comunidades.

A técnica foi proposta em (Newman & Girvan, 2004), trata-se de um método divisivo que começa com toda a rede sendo uma comunidade e vai removendo arestas que possuam o maior valor de *betweenness* até que cada vértice forma sua própria comunidade. Existem duas abordagens possíveis, a primeira é calcular o valor de *betweenness* só no começo do algoritmo e remover em cada iteração aquelas arestas com o maior valor de *betweenness*,

mas quando existem várias arestas entre duas comunidades, por algum motivo, pode acontecer que a maior parte dos caminhos entre os vértices das duas comunidades passem por apenas uma aresta, ficando as outras arestas com valor baixo para o *betweenness* e sendo removidas só quase nas últimas iterações, errando a detecção de comunidades. Uma outra abordagem é recalcular o *betweenness* a cada iteração assim evita-se que arestas que ligam duas comunidades sejam mantidas, isso claramente aumenta o custo computacional, mas aumenta significativamente a eficácia do algoritmo.

Os passos a seguir do algoritmo são:

1. Calcular o valor do *betweenness* para cada aresta da rede.
2. Remover a aresta com maior valor de *betweenness*, se duas ou mais arestas possuem o maior valor remover uma aleatoriamente.
3. Calcular de novo o *betweenness* para as arestas restantes.
4. Voltar ao segundo passo.

Para calcular os possíveis caminhos entre dois vértices da rede existem diferentes abordagens, a mais simples é utilizando o caminho mais curto entre os vértices, outra abordagem é baseada em caminhadas aleatórias e outra é baseada no fluxo de corrente da teoria dos circuitos. Essas três abordagens são descritas a seguir nas seguintes subseções.

Betweenness baseado no caminho mais curto

Nessa abordagem é calculado o caminho mais curto entre cada par de vértices da rede, assim o valor do *betweenness* para cada aresta será o número desses caminhos que passam por ela. O custo computacional para calcular o caminho mais curto entre cada par de vértices da rede utilizando busca em largura é $O(mn^2)$, sendo $m$ o número de arestas e $n$ o número de vértices. Esse tempo pode ser reduzido utilizando uma modificação proposta em (Newman & Girvan, 2004) que sugere que não é necessário recalcular o *betweenness* de todas as arestas em cada iteração, pois somente o valor do *betweenness* das arestas conectadas aos vértices das arestas removidas é alterado, assim recalculando o *betweenness* a cada iteração somente das arestas alteradas o tempo computacional é reduzido a $O(mn)$. Além disso, o tempo computacional irá diminuir a cada nova divisão da rede em cada iteração.

Betweenness baseado em caminhada aleatória

Ao invés de levar em conta o menor caminho entre dois vértices é possível usar uma caminhada aleatória entre dois vértices (em cada passo do caminho é escolhido uma aresta de um vértice vizinho aleatoriamente com igual probabilidade). Assim para calcular o valor do *betweenness* de uma aresta em particular $(u, w)$, é calculado o número esperado de vezes que uma caminhada aleatória desde um vértice $s$ até um vértice $t$ passa pela

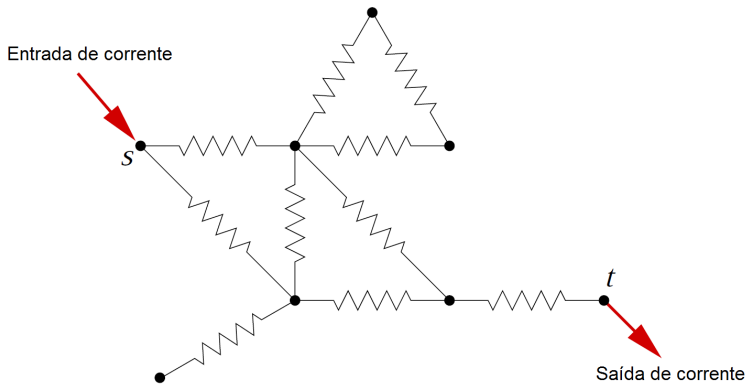



Figura 2.5: Exemplo de tipo de rede de resistências para o calculo do *betweenness* baseado no fluxo de corrente. Figura adaptada de (Newman & Girvan, 2004).

aresta $(u, w)$, calculando esse valor esperado para cada par de vértices $s$ e $t$ para a aresta $(u, w)$ e somando esses valores é obtido o valor do *betweenness* da aresta $(u, w)$.

Betweenness baseado em fluxo de corrente

Outra forma de calcular o *betweenness* de uma aresta é utilizando o conceito de fluxo de corrente da teoria dos circuitos. Considere que cada aresta é uma resistência e um vértice $s$ como unidade fonte de energia e outro vértice $t$ como sumidouro. O fluxo de corrente na rede percorrerá múltiplos caminhos em paralelo desde a fonte até o sumidouro, aqueles caminhos com menor resistência serão percorridos com maior intensidade de corrente. O *betweenness* de fluxo de corrente para uma aresta é definido como o valor absoluto da intensidade de corrente nessa aresta para todos os pares de vértices $s$ e $t$, esse valor da intensidade de corrente pode ser calculado utilizando as leis de Kirchhoff. A Figura 2.5 mostra um exemplo de rede com cada aresta sendo uma resistência.

### 2.4.3 Método baseado em caminhadas aleatórias

Este método é baseado na idéia de caminhadas aleatórias, esta caminhada pode ser entendida como o percurso que uma partícula *browniana* faz na rede, a cada novo passo a partícula escolhe qualquer vizinho com a mesma probabilidade. A técnica for proposta em (Zhou, 2003b), onde é definido uma distância de caminhada aleatória entre cada par de vértices que serve para detectar comunidades baseado em atratores locais e atratores globais. Este modelo foi estendido em (Zhou, 2003a) onde é definido um índice de dissimilaridade baseado na distância de caminhada aleatória que permite detectar comunidades com um algoritmo divisivo.

Considere uma rede $G = (V, E)$ com $n$ vértices e $m$ arestas, $V = 1, 2, 3, ..., n$. $A$ com elementos $a_{ij}$ representa a matriz de adjacência de $G$, onde $a_{ij} \geq 0$ é a intensidade da ligação entre o vértice $i$ e $j$, $a_{ij} = 0$ representa que não existe aresta entre esses vértices. Uma partícula *browniana* no vértice $i$ passa para o vértice $j$ com probabilidade $p_{ij}$, sendo

$P$ a matriz de transferência, $p_{ij}$ pode ser calculado segundo a Equação (2.15).

$$p_{ij} = \frac{A_{ij}}{\sum_{k=1}^{n} A_{ik}}. \tag{2.15}$$

Assim a distância de caminhada aleatória entre dois vértices $i$ e $j$ segundo (Zhou, 2003b) é a média de passos necessários para que a partícula no vértice $i$ chegue até o vértice $j$. A Equação (2.16) define a distância $d_{ij}$.

$$d_{ij} = \sum_{k=1}^{n} \left( \frac{1}{I - B(j)} \right)_{ik}, \tag{2.16}$$

onde $I$ é a matriz identidade $n \times n$, e $B(j)$ é a matriz de transferência modificada, onde $B_{kj} = 0, \forall k \in V$. Para calcular a distância de todos os vértices da rede até $j$ é necessário resolver o sistema de equações lineares descrito na Equação (2.17) derivada algebricamente da Equação (2.16).

$$[I - B(j)]\{d_{1j}, d_{2j}, ..., d_{nj}\}^T = [1, ..., 1]^T. \tag{2.17}$$

Dois conceitos são introduzidos pelo autor: atratores globais e atratores locais. Um vértice $j$ é atrator global de $i$ se a distância de $i$ até $j$ é a menor possível entre o vértice $i$ e qualquer outro vértice da rede, $d_{ij} \leq d_{ik}$, $\forall k \in V$. Do mesmo jeito $j$ é atrator local de $i$ se a distância entre o vértice $i$ e o seu vizinho $j$ é a menor possível entre o vértice $i$ e qualquer outro vértice vizinho, $d_{ij} \leq d_{ik}$ para todos os vizinhos $k$ do vértice $i$.

Faz sentido pensar que um vértice $i$ deve pertencer à mesma comunidade que seu atrator local $j$, pois o vizinho $j$ possui a menor distância dentre os vizinhos de $i$. Assim as comunidades, chamadas pelo autor comunidades $L$, são formadas seguindo as seguintes regras:

- Um vértice $j \in L$ se $i \in L$ e $j$ é atrator local de $i$.
- Se $i \in L$ e existe um vértice $k$ onde $i$ é seu atrator local, então $k \in L$.

Do mesmo jeito é possível formar comunidades $G$ utilizando os atratores globais ao invés dos locais.

O método descrito foi estendido em (Zhou, 2003a) mediante a introdução de um índice de dissimilaridade entre cada par de vértices baseado nas distâncias de caminhada aleatória $d_{ij}$. As distancias do vértice $i$ até os demais vértices na rede pode ser visto como a perspectiva que o vértice $i$ tem da rede, quão longe fica cada vértice na rede desde o ponto de vista de $i$, assim a diferença de perspectivas ou índice de dissimilaridade de dois vértices $i$ e $j$ é dada pela Equação (2.18).

$$\Lambda(i, j) = \sqrt{\frac{\sum_{k \neq i, k \neq j} (d_{ik} - d_{jk})^2}{n - 2}}. \tag{2.18}$$

Assim vértices que pertençam à mesma comunidade devem ter baixo valor do índice de dissimilaridade. O algoritmo começa com a rede toda sendo um grupo só e a cada iteração a rede é decomposta em subgrupos segundo uns valores de limiar inferiores e superiores para o índice de dissimilaridade que são calculados para cada grupo. Nesse sentido é formado toda uma hierarquia de partições.

### 2.4.4 Método baseado em competição de partículas

Este método foi proposto em (Quiles et al., 2008) e consiste em que diversas partículas caminham em uma rede competindo entre si para dominar seus vértices. Ao mesmo tempo, partículas intrusas são repelidas ao entrarem em territórios pertencentes a outras partículas. Depois de algum tempo de execução cada partícula se isola dentro de uma comunidade da rede. O método de competição de partículas envolve conceitos de sistemas dinâmicos, processos estocásticos e mecanismos de competição. Nesse modelo é considerado dois tipos de dinâmica, a dinâmica das partículas e a dinâmica dos vértices. Cada partícula $p_j$ é descrita por duas variáveis $p_j^v(t)$ e $p_j^\omega(t) \in [\omega_{min}, \omega_{max}]$, a primeira indica o vértice que a partícula está visitando no instante $t$, e a segunda indica o potencial da partícula, que caracteriza a sua habilidade para explorar no tempo $t$. Os valores $\omega_{min}$ e $\omega_{max}$ indicam os valores mínimo e máximo para o potencial dos vértices e das partículas. A dinâmica das partículas é expressa pelas seguintes equações:

$$p_j^\omega(t+1) = v_i \tag{2.19}$$

$$p_j^\omega(t+1) = \begin{cases} p_j^\omega(t) & \text{se } v_i^p(t) = 0 \\ p_j^\omega(t) + (\omega_{max} - p_j^\omega(t))\Delta_p & \text{se } v_i^p(t) = p_j \neq 0 \\ p_j^\omega(t) - (p_j^\omega(t) - \omega_{min})\Delta_p & \text{se } v_i^p(t) \neq p_j \neq 0 \end{cases} \tag{2.20}$$

onde o parâmetro $\Delta_p$ controla a velocidade com que o potencial de cada partícula é atualizado.

No caso da dinâmica dos vértices também existem duas variáveis $v_i^p(t)$ e $v_i^\omega(t)$. A primeira variável serve para indicar a qual partícula $p_j$ o vértice $i$ pertence no tempo $t$, entanto, a segunda variável tem o objetivo de armazenar o potencial do vértice $v_i$ no tempo $t$, que indica a força com que o a partícula $p_j$ domina esse vértice. No começo cada vértice não é dominado por nenhuma partícula assim $v_i^p(t) = 0$.

Se o potencial do vértice $v_i$ é menor que $\omega_{min}$, então esse vértice está no estado livre e pode ser dominado pela primeira partícula que o visitar. As seguintes equações definem a dinâmica dos vértices:

$$v_i^p(t+1) = \begin{cases} v_i^p(t) & \text{se } v_i^\gamma = 0 \\ p_j & \text{se } v_i^\gamma = 1 \text{ e } v_i^p(t) \leq \omega_{min} \end{cases} \tag{2.21}$$

$$v_i^{\omega}(t+1) = \begin{cases} v_i^{\omega}(t) & \text{se } v_i^{\gamma} = 0 \\ \max\left(\omega_{min}, v_i^{\omega}(t) - \Delta_v\right) & \text{se } v_i^{\gamma} = 1 \text{ e } v_i^{p}(t) \neq p_j \\ p_i^{\omega}(t+1) & \text{se } v_i^{\gamma} = 1 \text{ e } v_i^{p}(t) = p_j \end{cases} \tag{2.22}$$

onde o parâmetro $\Delta_v$ é utilizado para controlar a velocidade com que o potencial do vértice é atualizado, e $v_i^{\gamma}$ indica se o vértice está sendo visitado no instante $t$ por alguma partícula, com valores 1 ou 0.

Inicialmente, $K$ partículas são inseridas em $K$ vértices selecionados aleatoriamente. Cada uma dessas partículas e todos os vértices possuem um potencial inicial igual ao potencial mínimo. Em seguida, cada partícula escolhe um vértice vizinho para visitar, de maneira que cada uma pode encontrar uma das três situações:

1. Se o vértice atual $v_i$ não pertence a nenhuma partícula, então o potencial da partícula não é alterado e o vértice é marcado como pertencente à partícula com potencial igual ao da partícula.

2. Se o vértice atual já pertence à partícula corrente, o potencial da partícula é aumentado e ao potencial do vértice novamente é atribuído o mesmo valor do potencial da partícula.

3. Se o vértice atual pertence a uma partícula diferente da partícula que o está visitando, então ambos potenciais são diminuídos. Se for reduzido a valores menores que $\omega_{min}$, a partícula é descartada e uma nova, com potencial igual a $\omega_{min}$ é inserida em um vértice escolhido aleatoriamente. No caso em que o potencial do vértice chega a um valor menor que $\omega_{min}$, então sua pertenencia retorna ao estado inicial 0.

A idéia do algoritmo é que o grau de pertinência de um vértice é reforçado se este for visitado freqüentemente pela mesma partícula, e reduzido se for visitado por outras partículas. O processo continua até o momento em que o equilíbrio dinâmico for alcançado, isto quando a maioria dos vértices não muda mais de partícula dominante. Finalmente no equilíbrio, cada vértice dominará uma comunidade na rede. A Figura 2.6 mostra o processo de detecção de comunidades em uma rede com 128 vértices e 4 comunidades, foram inseridas 4 partículas.

Um assunto a considerar é o número de partículas a inserir, pois não é conhecido a priori o número de comunidades. Uma forma de estimar o número de comunidades é rodar o algoritmo com diferente número de partículas, e calcular a média do potencial das partículas para cada execução do algoritmo, aquela com o maior valor médio do potencial estimará o número de comunidades da rede.

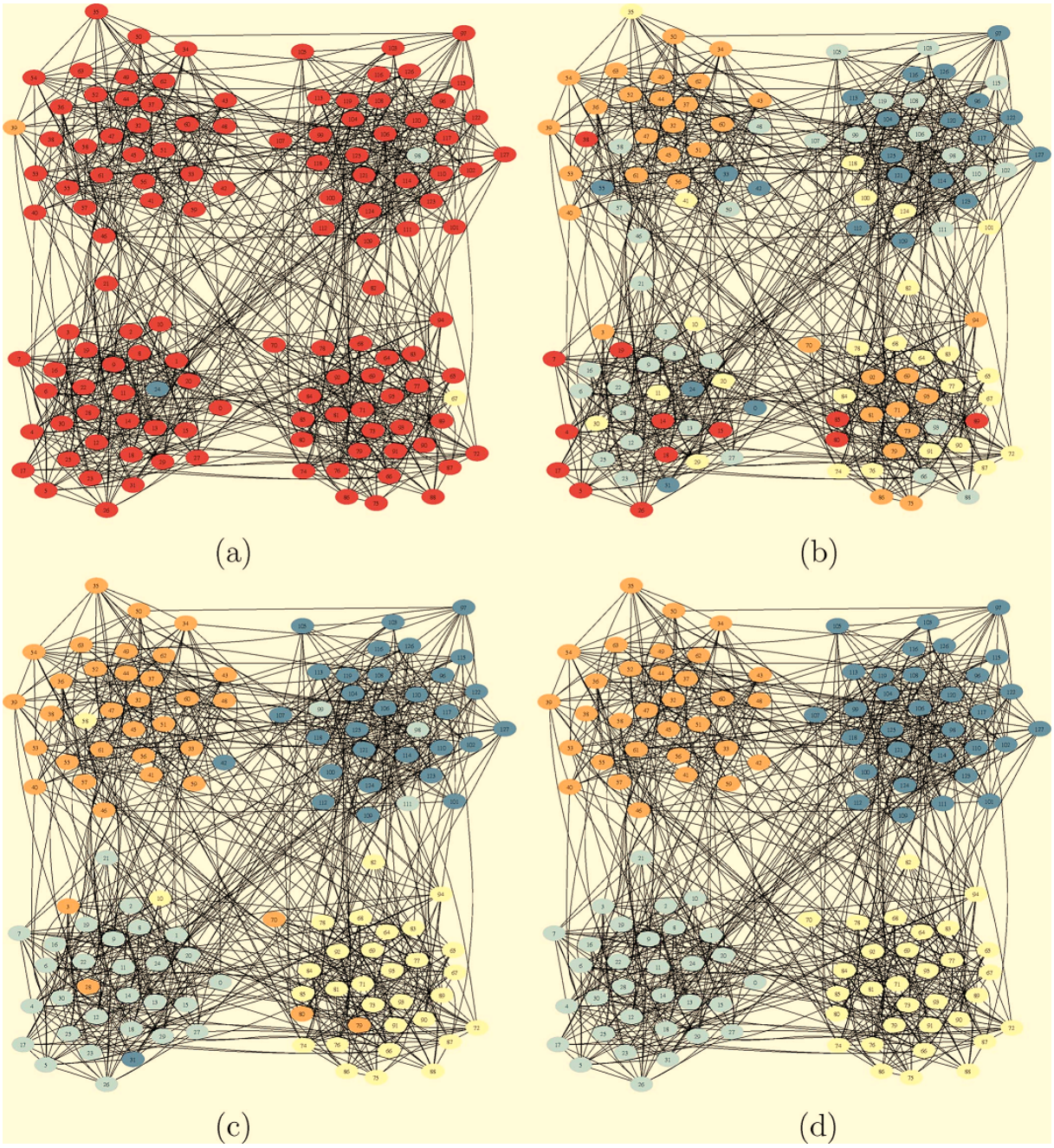



Figura 2.6: Detecção de comunidades utilizando competição de partículas em uma rede com 128 vértices e 4 comunidades. (a) Configuração inicial. Quatro partículas representadas pelas cores amarelo, azul claro, laranja e azul escuro são aleatoriamente colocadas na rede. O vermelho representa vértices livres. (b) Iteração 250, (c) iteração 3500, (d) iteração 7000. Figura obtida de (Quiles et al., 2008).

## 2.5 Consenso e controle focalizado em redes complexas

Em uma rede de sistemas dinâmicos acoplados, consenso refere-se a atingir um acordo para um valor em particular que depende do estado de todos os agentes (Olfati-saber et al., 2007). A idéia básica do consenso é que cada agente atualiza seu próprio estado baseado nos estados dos seus vizinhos. Finalmente, todos os agentes atingem um valor comum para seus estados.

Considere $A = [a_{ij}]$ sendo a matriz de adjacência da rede $G = (V, E)$, onde $V = \{1, ..., n\}$ é o conjunto de agentes ou vértices e $E$ é o conjunto de arestas da rede. O conjunto de vizinhos do vértice $i$ é definido como $N_i = \{j \in V : a_{ij} \neq 0\}$.

A Equação (2.23) define regra de evolução do estado de cada vértice $i$.

$$\dot{x}_i(t) = \sum_{j \in N_i} a_{ij}(x_j(t) - x_i(t)), \tag{2.23}$$

onde $x_i$ é a variável que representa a dinâmica do vértice $i$, e "$\cdot$"denota a derivada sobre tempo, por exemplo $\dot{x}_i(t) = d_x/d_t$. O sistema linear representado pela Equação (2.23) é um sistema dinâmico distribuído de consenso proposto em (Olfati-Saber & Murray, 2004). O sistema converge para um mesmo estado via interações locais entre vizinhos na rede. Supondo que a rede é não direcionada, a soma do valor do estado de todos os agentes é invariante e um consenso assintótico é atingido, com uma decisão coletiva $\alpha$ igual à média dos estados inicias dos vértices, $\alpha = \frac{1}{n}\sum_i x_i(0)$. Algoritmos de consenso como a propriedade de soma invariante dos estados dos vértices são chamados de algoritmos de consenso médio (*average-consensus*).

Existem muitos trabalhos que tentam controlar a dinâmica de uma rede de agentes para um estado desejado, o estado desejado pode ser um ponto de equilíbrio ou uma órbita periódica (Xiang et al., 2007). Uma das técnicas para obter tal comportamento é chamado de controle focalizado (*pinning control*) que consiste em adicionar controladores locais em alguns vértices da rede.

Considere que a rede descrita anteriormente $G = (V, E)$ represente a rede de comunicações de $N$ osciladores acoplados, e que $A = [a_{ij}]$ representa a matriz de acoplamento de toda a rede. A evolução do $i$th oscilador no tempo é descrita na Equação (2.24).

$$\dot{x}_i(t) = f(x_i(t)) - c\sum_{j=1}^{N} a_{ij}h(x_j(t) - x_i(t)), \tag{2.24}$$

onde $x_i = (x_{i1}, x_{i2}, ..., x_{im})$ representa o vetor do estado do $i$th oscilador, $f(\cdot)$ descreve a dinâmica individual do oscilador, $c > 0$ é a força global do acoplamento, $h$ é a função interna de acoplamento que caracteriza a conexão entre os estados de cada oscilador. A rede $G$ pode ser descrita utilizando o grafo Laplaciano $L = D - A$, onde $D = diag(d_1, ..., d_n)$ é a matriz diagonal dos graus dos vértices de $G$ com $d_i = \sum_{j \neq i} a_{ij}$. No controle focalizado é esperado que somente uma pequena fração de vértices da rede sejam controlados. Con-

sidere os primeiros $l$ vértices para serem controlados, adicionando controle e utilizando o grafo Laplaciano $L$ de $G$ a Equação (2.24) pode ser reescrita como:

$$\dot{x}_i(t) = f(x_i(t)) - c\sum_{j=1}^{N} l_{ij}h(x_j(t)) + u_i(t), \tag{2.25}$$

onde $u_i(t)$ representa o controle local, sendo $u_i(t) = k_i(h(s(t)) - h(x_i(t)))$, e $s(t)$ é a trajetória de referência. A trajetória de referência é descrita por um oscilador independente com dinâmica $\dot{s}(t) = f(s(t))$. $k_i = z$ quantifica a força do controle para o vértice $i$, nesse sentido se o vértice $i$ não está controlado então $k_i = 0$. Em geral a tarefa do controle focalizado é dirigir a rede complexa dinâmica para $s(t)$ quando o tempo $t \to \infty$, aplicando controle em alguns vértices.

## 2.6 Considerações finais

Neste capítulo foram apresentadas revisões bibliográficas sucintas sobre redes complexas e estrutura de comunidades. Entende-se por rede complexa uma rede que não possui uma distribuição dos graus dos vértices trivial. Foram descritos também alguns modelos de redes complexas como: o modelo de redes aleatórias, redes de pequeno mundo e redes livres de escala. É importante o estudo de algoritmos de detecção de comunidades nas redes complexas porque permite conhecer as estruturas das redes reais, inclusive no campo da Bioinformática é possível descobrir a função de algumas proteínas desconhecidas somente conhecendo a que grupo ou módulo com função conhecida elas pertencem numa rede de interação de proteínas.

Além disso, foram apresentados alguns métodos representativos de detecção de comunidades em redes complexas, como o algoritmo divisivo baseado em uma medida chamada de *betweenness* que remove interativamente arestas com maior valor dessa medida, também foi apresentado alguns métodos de detecção de comunidades baseados em caminhadas aleatórias, caminhada que representa o percurso que uma partícula *browniana* faria na rede, e um método baseado em competição de partículas.

Finalmente, foram apresentados dois conceitos muito importantes no tópico de sistemas dinâmicos de agentes acoplados em redes complexas: consenso e controle focalizado ou *pinning control*.

Capítulo

3

# Agrupamento de dados

Agrupamento de dados consiste em uma classificação não supervisionada de padrões em grupos (Jain et al., 1999), cujos elementos mais similares são mantidos em um mesmo grupo e elementos diferentes são mantidos em grupos distintos, nesse sentido tenta-se minimizar a dissimilaridade entre objetos de um mesmo grupo e maximizar a dissimilaridade entre objetos de diferentes grupos, como mostrado na Figura 3.1. O processo de agrupar dados é importante em muitas áreas de pesquisa, tais como bioinformática (Golub et al., 1999; Daxin Jiang, 2004), mineração de dados (Ester et al., 1996), segmentação de imagens (Silva & Zhao, 2007), entre outras.

Segundo (Jain et al., 1999), as técnicas de agrupamento de dados podem ser divididas em: técnicas particionais e técnicas hierárquicas. As técnicas particionais, procuram obter a melhor partição em grupos do conjunto de dados, estas técnicas geralmente precisam conhecer o número de grupos antes de tentar gerar a partição, assim a entrada destes algoritmos é o conjunto de dados e o número de grupos desejado, e a saída é uma partição

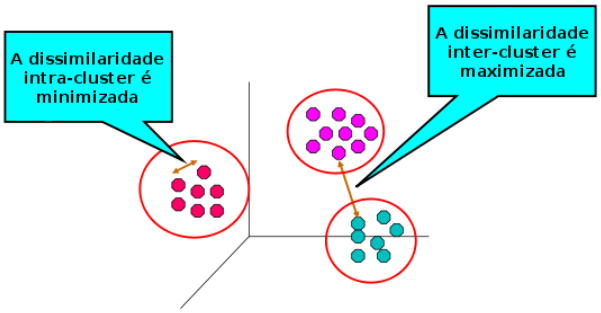



Figura 3.1: Agrupamento de dados. Tenta-se minimizar a dissimilaridade entre objetos de um mesmo grupo e maximizar a dissimilaridade entre objetos de diferentes grupos. Figura adaptada de (Tan et al., 2006)

do conjunto de dados com o número de grupos especificado. Nas técnicas de agrupamento de dados hierárquicas não é gerada somente uma partição, senão toda uma hierarquia de partições, na qual no nível mais alto todos os elementos estão agrupados em apenas um grupo, e no nível mais baixo cada elemento forma o seu próprio grupo. O resultado dos algoritmos de agrupamento de dados hierárquicos é representado numa estrutura de árvore chamada dendograma.

## 3.1 Componentes do processo de agrupamento de dados

O processo de agrupamento de dados pode ser divido em varias etapas (Jain et al., 1999):

1. Representação dos padrões
   Aqui são definidos os atributos que representarão os padrões, além do tipo e escala de cada atributo. Essa escolha dos atributos é muito importante, pois os atributos têm que possuir informações suficientes para caracterizar bem cada padrão. É possível fazer aqui um pré-processamento dos atributos, tais como: tratar valores ausentes, normalizar atributos, padronizar atributos, remover padrões duplicados.

2. Definição da medida de similaridade
   Para medir a similaridade entre dois padrões é preciso usar uma medida de distância, usualmente essa medida é a distância euclidiana que mede a dissimilaridade entre dois elementos. Existem outras medidas de similaridade/dissimilaridade: distância de Manhattan, distância de Mahalanobis, correlação de pearson, entre outras.

3. Algoritmo de agrupamento
   Nessa etapa define-se o algoritmo de agrupamento, existem algoritmos de agrupamento particionais e hierárquicos, alguns algoritmos podem gerar como saída uma partição rígida, onde cada elemento pertence unicamente a um grupo, ou uma partição difusa, onde cada elemento pode pertencer a distintos grupos com diferentes graus de pertinência, ou também gerar toda uma estrutura de partições como nos algoritmos hierárquicos.

4. Validação
   Os algoritmos de agrupamento sempre encontram grupos, mesmo que os dados sejam aleatórios e sem estrutura nenhuma. Por tanto, é necessário avaliar a qualidade da partição ou partições obtidas. Se o algoritmo de agrupamento for hierárquico, nessa etapa é escolhida a melhor partição do dendograma, escolhendo ao mesmo tempo o número de grupos. Existem três tipos de índices de validação: externos, que avaliam a qualidade da partição em relação a uma partição de referencia externa, internos, que avaliam a qualidade dos grupos formados numa partição, e os índices relativos que avaliam um conjunto de partições geradas.

Existem diferentes técnicas de agrupamento de dados, alguns algoritmos recebem como entrada a matriz de dados com os valores dos atributos para cada padrão, um exemplo tradicional desse tipo de algoritmos é o $K$-medias, outros algoritmos como o Single Linkage, Complete Linkage e Average Linkage, não precisam da matriz original dos dados, mas precisam da matriz de similaridade entre os padrões. Além disso existem algoritmos de agrupamento de dados baseados em grafos, esses algoritmos constroem um grafo a partir do conjunto de dados e aplicam alguma técnica para encontrar os grupos nesse grafo. Esses algoritmos serão descritos nas seguintes seções.

## 3.2 Algoritmos particionais

Os algoritmos particionais obtém apenas uma partição dos dados (Jain et al., 1999). Os algoritmos particionais geralmente recebem como parâmetro de entrada o número de grupos desejado, em seguida, os algoritmos particionais usualmente fornecem esse número de grupos mediante a otimização de algum critério, um dos algoritmos mais simples e mais utilizados é o $K$-medias descrito na seção a seguir.

### 3.2.1 K-Médias

K-Médias é um dos algoritmos particionais mais conhecidos, onde $K$ é o número de grupos desejado. O algoritmo começa com um conjunto de $K$ elementos chamados centróides, onde cada centróide representa um grupo, esses elementos centróides são escolhidos de forma aleatória e atualizados em cada passo do algoritmo até alcançar algum critério de convergência.

Os passos básicos do algoritmo são:

1. Escolher $K$ centróides de maneira aleatória. Os centróides podem ser escolhidos dentre os elementos do conjunto de dados.

2. Formar $K$ grupos atribuindo cada ponto ao seu centróide mais próximo.

3. Recalcular a nova posição de cada centróide dos grupos com os elementos atribuídos a cada grupo no passo anterior.

$$c_k{}' = \frac{1}{n_k} \sum_{i=1}^{n_k} x_i, \tag{3.1}$$

   onde $c_k{}'$ é o novo centróide do grupo $k$, $n_k$ é o número de elementos do grupo $K$, e o vetor $x_i$ representa o elemento $i$ do conjunto de dados.

4. Se o critério de convergência não é atingido, voltar ao passo 2. Geralmente o critério de convergência é quando os centróides dos grupos não mudam mais de posição, ou quando os elementos do conjunto de dados não mudam mais de grupo.

A Figura 3.2 mostra o resultado do $K$-médias com dois centróides em um conjunto de dados com dois grupos originais.

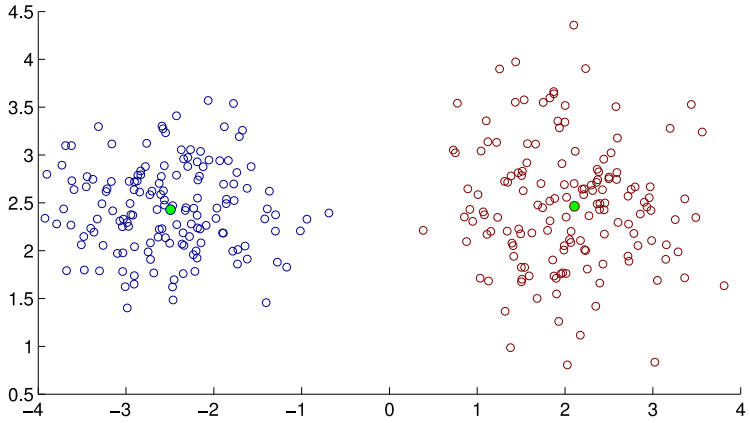



Figura 3.2: Resultado da aplicação do método $K$-médias em um conjunto de dados com dois grupos. Os pontos verdes indicam os centróides finais.

## 3.3 Algoritmos hierárquicos

Os algoritmos hierárquicos produzem uma hierarquia de partições de tipo árvore, estas técnicas não precisam da informação do número de grupos a priori (Silva & Zhao, 2007). Os algoritmos hierárquicos podem ser divididos em dois tipos: aglomerativos e divisivos. Os algoritmos divisivos partem de um só grupo, um grupo contendo todos os elementos do conjunto de dados, e os dividem em grupos cada vez menores, até que cada elemento represente um grupo. Esse tipo de algoritmos são geralmente custosos computacionalmente por causa das inúmeras divisões possíveis em cada iteração, e por tanto, pouco utilizados na prática. Nos algoritmos aglomerativos inicialmente cada elemento representa seu próprio grupo, e o algoritmo vai juntando de dois em dois os grupos mais similares até obter um grupo só contendo todos os elementos. Tanto nos algoritmos divisivos ou aglomerativos é possível montar uma representação de dendrograma, que permita visualizar em um gráfico em forma de árvore a hierarquia das partições encontradas. A Figura 3.3 mostra um esquema de dendograma.

Entre os algoritmos aglomerativos mais representativos temos o Single Linkage, Average Linkage e o Complete Linkage. A idéia do Single Linkage é juntar aqueles grupos mais próximos um de outro, onde a proximidade entre dois grupos é dada pela menor distância entre dois objetos, um objeto de cada grupo. No Complete Linkage essa proximidade entre dois grupos é dada pela maior distância entre dois objetos, onde cada objeto pertence a um grupo. Finalmente no Average Linkage a proximidade entre dois grupos é calculada como a distância média entre pares de objetos, um objeto de cada grupo. O resultado da aplicação destes algoritmos em diferentes conjuntos de dados pode ser visto na Figura 3.4.

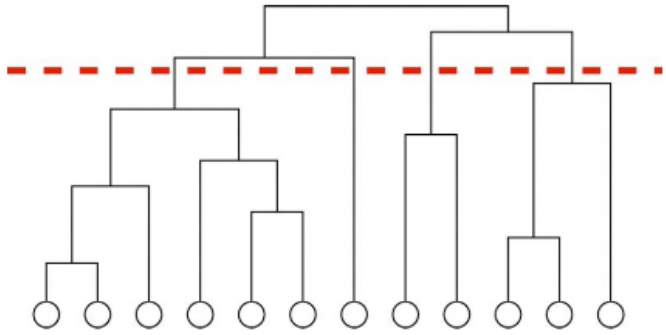


Figura 3.3: Dendograma. A linha vermelha indica um corte no dendograma, uma partição do conjunto de dados em 4 grupos. Figura obtida de (Newman & Girvan, 2004)

Os passos de agrupamento de dados a seguir no Single Linkage, Complete Linkage o Average Linkage são:

1. Colocar cada elemento no seu próprio grupo. Construir a matriz de distâncias entre os elementos do conjunto de dados.

2. Juntar os dois grupos com a menor distância.

3. Recalcular a distâncias entre o grupo formado no passo 2 e todos os demais grupos, onde essa distância será a menor ou maior distância entre dois elementos de diferentes grupos, ou a média das distâncias entre os elementos dos grupos, segundo o algoritmo escolhido, Single Linkage, Complete Linkage ou Average Linkage.

4. Se ainda todos os elementos não estiverem em um grupo só, voltar ao passo 2.

Single Linkage pode encontrar grupos de diferentes formas e tamanhos, mas é sensível a ruido e *outliers* por causa do efeito de encadeamento (Nagy, 1968), no entanto, funciona bem para grupos bem separados. O Complete Linkage é menos sensível ao ruído, mas tende a dividir grandes grupos e favorecer formas globulares (Pang-Ning Tan, 2006). O Average Linkage é um ponto de equilíbrio entre o Single e Complete Linkage, sendo menos sensível ao ruído e *outliers*, mas também é polarizado para a formação de grupos globulares.

## 3.4 Algoritmos de agrupamento de dados baseados em grafos

Existem na literatura alguns trabalhos que tentam abordar o problema de agrupamento de dados usando grafos. Esses trabalhos inicialmente criam uma representação em grafo do conjunto de dados usando diferentes técnicas, uma dessas técnicas é usar os $K$-vizinhos mais próximos, que será visto no algoritmo CHAMELEON descrito mais adiante. Após criada essa representação em grafo, esses trabalhos utilizam alguma técnica

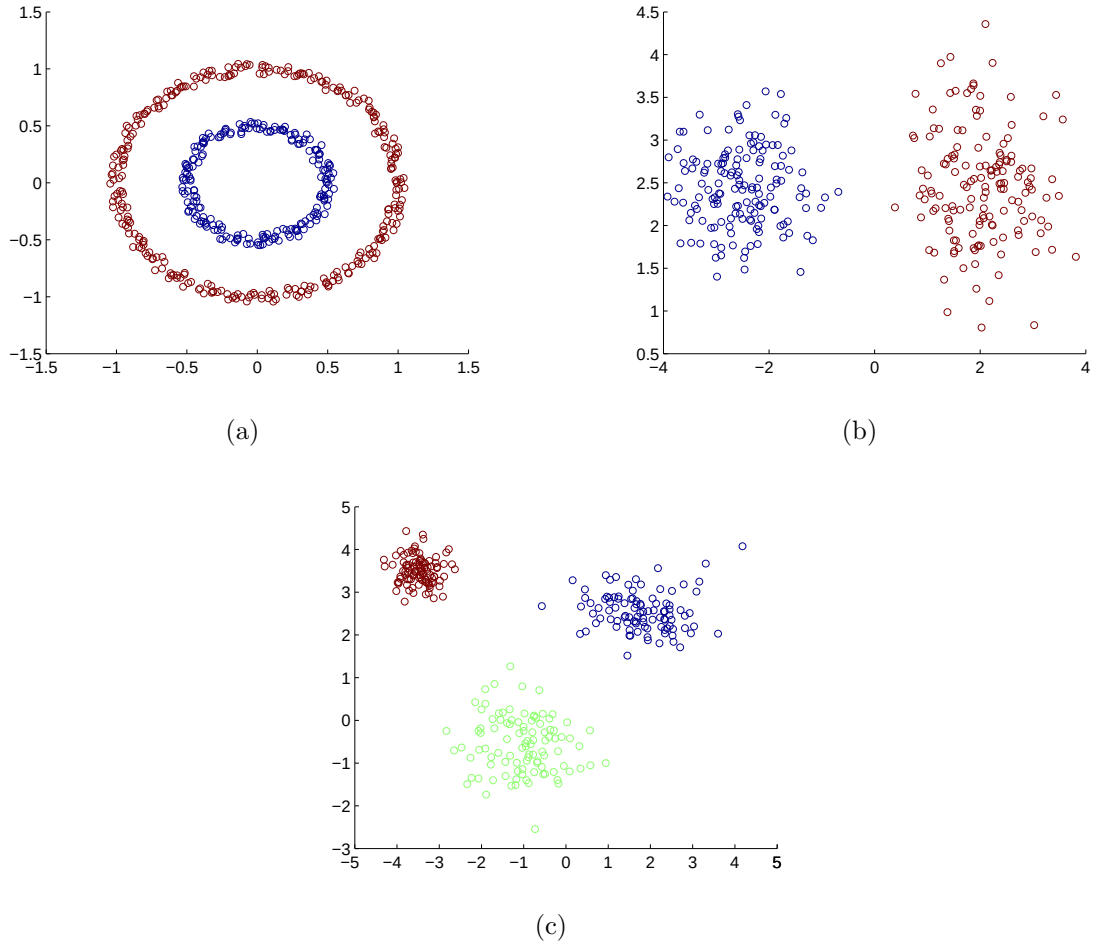



Figura 3.4: Resultado da aplicação de diferentes algoritmos de agrupamento hierárquico: (a) Single Linkkage, (b) Complete Linkage, (c) Average Linkage.

de partição de grafos ou alguma técnica de detecção de comunidades em redes complexas para encontrar os grupos reais.

A seguir são apresentadas três técnicas de agrupamento baseados em grafos.

### 3.4.1 CHAMELEON

CHAMELEON (Karypis et al., 1999), é um algoritmo de agrupamento de dados hierárquico. A principal característica do algoritmo é que leva em conta a interconectividade e proximidade para identificar os grupos mais similares, e que não depende de uma medida de similaridade como o $K$-medias, nesse sentido o algoritmo é capaz de identificar grupos com diferentes formas e tamanhos.

CHAMELEON modela o conjunto de dados como um grafo esparso, onde os vértices representam objetos e as arestas representam a similaridade entre esses objetos. O algoritmo encontra grupos no conjunto de dados usando duas fases, na primeira fase utiliza um algoritmo de particionamento de grafos para dividir o conjunto de dados em grupos pequenos; na segunda fase usa um algoritmo aglomerativo para juntar esses grupos pequenos e encontrar os grupos reais. A Figura 3.5 mostra as fases do algoritmo.


Figura 3.5: Fases do algoritmo CHAMELEON. Figura adaptada de (Karypis et al., 1999)

- Fase divisiva

  A primeira tarefa é construir um grafo a partir da matriz de similaridade, essa abordagem é muito comum em algoritmos hierárquicos (Jain & Dubes, 1988). CHAMELEON forma um grafo esparso a partir da matriz de similaridade usando a abordagem dos $K$-vizinhos mais próximos, onde cada objeto da matriz de similaridade representa um vértice, e onde $K$ arestas são criadas com cada um dos $K$-vizinhos mais próximos dele. É preciso escolher um número de $K$ vizinhos suficientemente grande para a rede não ficar desconectada. A Figura 3.6 mostra quatro exemplos de redes construídas como os 5, 10, 15 e 20 vizinhos mais próximos em um conjunto de dados com dois grupos.

  Sobre o grafo formado, CHAMELEON aplica um algoritmo de particionamento de grafos hMetis (G. Karypis, 1998), esse algoritmo tem por objetivo dividir o grafo inicial em vários grupos pequenos, onde a soma do peso das arestas que ligam esses grupos seja mínimo, essas arestas são chamadas de arestas de corte. Como o peso das arestas representa a similaridade entre objetos, a similaridade entre dois grupos de vértices é minimizado por uma partição que minimize a soma das arestas de corte entre eles.

  O algoritmo de particionamento divide recursivamente o grafo em grupos cada vez menores, até que o maior grupo tenha um número de vértices menor a um tamanho mínimo definido pelo usuário como parâmetro inicial .

- Fase Aglomerativa

  Nessa fase CHAMELEON tenta unir os pequenos grupos formados na fase anterior, segundo as medidas de interconectividade relativa (RI) e proximidade relativa (RC). CHAMELEON pode usar duas abordagens diferentes: mediante a definição de parâmetros ou mediante a otimização de uma função definida.

  Na abordagem de definição de parâmetros, o algoritmo une dois grupos quando esses parâmetros, RI e RC, são excedidos.

  Muitos algoritmos medem a interconectividade absoluta (EC) entre dois grupos $C_i$ e $C_j$ em termos de arestas de corte: a soma dos pesos das arestas que ligam os dois grupos $C_i$ e $C_j$. Interconectividade relativa é essa interconectividade absoluta

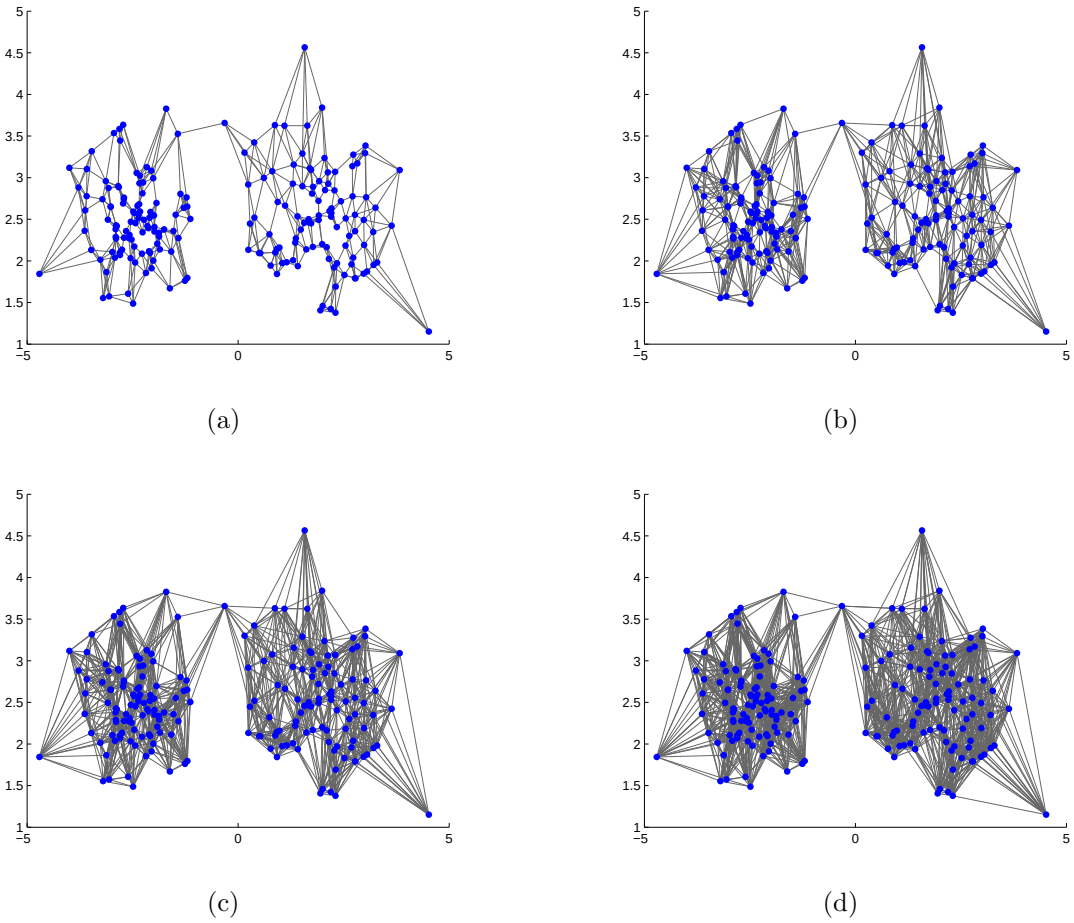



Figura 3.6: Redes construídas utilizando o algoritmo $K$-vizinhos mais próximos. (a) com $K = 5$ e (b) com $K = 10$.

normalizada em relação à interconectividade interna dos dois grupos. A interconectividade interna é a soma dos pesos das arestas de corte mínimo que dividem os dois grupos em partes iguais.

Assim, a interconectividade relativa é definida como:

$$RI(C_i, C_j) = \frac{|EC(C_i, C_j)|}{\frac{|EC(C_i)|+|EC(C_j)|}{2}}. \tag{3.2}$$

A proximidade absoluta é a média dos pesos das arestas que conectam vértices dos grupos $C_i$ e $C_j$, e a proximidade interna é definida como a média dos pesos das arestas de corte mínimo que dividem $C_i$ e $C_j$ em partes iguais. Por tanto, a proximidade relativa de $C_i$ e $C_j$ é a proximidade absoluta normalizada em relação à proximidade interna.

$$RC(C_i, C_j) = \frac{\bar{S}EC(C_i, C_j)}{\frac{|C_i|}{|C_i|+|C_j|}\bar{S}EC(C_i) + \frac{|C_j|}{|C_i|+|C_j|}\bar{S}EC(C_j)}, \tag{3.3}$$

onde $\bar{S}EC(C_i)$ e $\bar{S}EC(C_j)$ são a proximidade interna de $C_i$ e $C_j$, $\bar{S}EC(C_i, C_j)$ é a

proximidade absoluta, $|C_i|$ e $|C_j|$ é o número de vértices de cada grupo.

Assim, na abordagem de definição de parâmetros, o algoritmo irá unir dois grupos $|C_i|$ e $|C_j|$ se os valores de RI e RC excederem os valores definidos pelo usuário.

Na segunda abordagem é utilizado uma função que leva em conta a RI e RC. CHAMELEON une dois grupos que maximizam essa função. Como é desejável que a RI e RC sejam altos, é intuitivo pensar em um produto desses valores, mas nesse caso a RI e RC teriam a mesma importância, para permitir ajustar a importância de alguns desses valores é introduzido o parâmetro $\alpha$.

$$RI(C_i, C_j) \times RC(C_i, C_j)^{\alpha}. \tag{3.4}$$

Nesse sentido, CHAMELEON tenta encontrar dois grupos que maximizem a função (3.4) para ser unidos. Se $\alpha > 1$, o algoritmo dará mais importância à proximidade relativa, e quando $\alpha < 1$, dará mais importância à interconectividade relativa.

A Figura 3.7 mostra o resultado da aplicação do algoritmo em diferentes conjuntos de dados com grupos de diferentes formas.

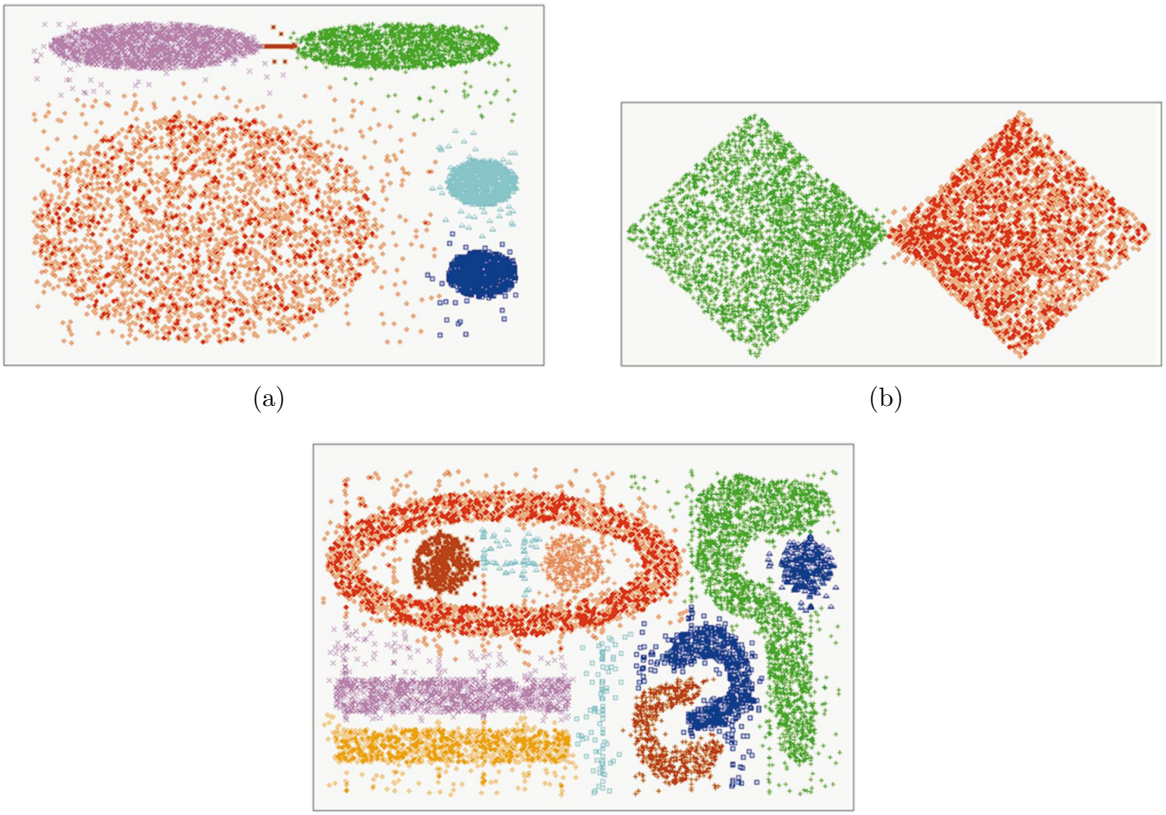



Figura 3.7: Resultado da aplicação do algoritmo CHAMELEON em diferentes conjuntos de dados. Figura adaptada de (Karypis et al., 1999)

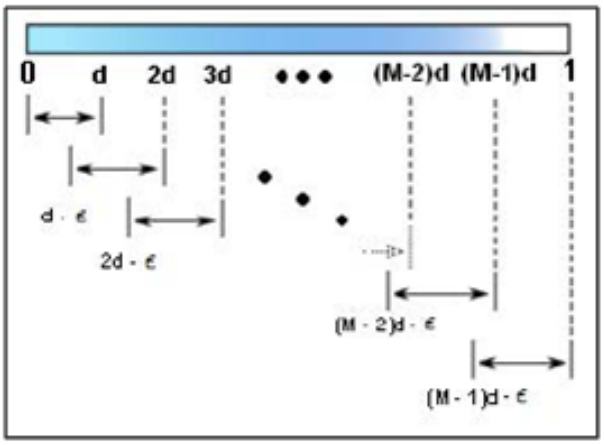



Figura 3.8: Divisão de intervalos para a formação da rede.

### 3.4.2 Agrupamento de pixels usando medida de modularidade em redes complexas

Essa técnica foi proposta em (Silva & Zhao, 2007). O objetivo da técnica é agrupar *pixels* para segmentação de imagens, assim, o conjunto de dados é os *pixels* da imagem em tons de cinza, o processo de agrupamento está composto em duas etapas: formação da rede e partição da rede.

Formação da rede

Cada vértice é um elemento do conjunto de dados. Os dados de entrada são inicialmente normalizados no intervalo $[0, 1]$. No próximo passo define-se um conjunto de intervalos sobrepostos com tamanho $d + \epsilon$, cobrindo todo o intervalo $[0, 1]$:

$$[0, d], [d - \epsilon, 2d], [i \times d - \epsilon, (i + 1) \times d], [(M - 1) \times d - \epsilon, 1], \tag{3.5}$$

onde $M$ é o número de intervalos, $d$ é o tamanho da janela no intervalo que não sofre sobreposição e $\epsilon$ é o tamanho da sobreposição, como mostra a Figura 3.8.

Depois disso, a quantidade de elementos que pertence a cada intervalo ($S_i$ para o intervalo $i$) é contado. A rede é formada conectando-se cada vértice aos seus $K_i$ vizinhos mais próximos, onde $K_i$ é calculado segundo a Equação (3.6):

$$K_i = {S_{max}}^2 - (S_{max} - S_i)^2, \tag{3.6}$$

onde $S_{max}$ é o número de elementos do intervalo que possui mais elementos, assim $S_{max}$ terá um grau maior de conectividade $K_i$, enquanto que um intervalo que possui poucos elementos terá valor baixo para $K_i$, além disso, quanto mais acentuado for o peso de $S_{max}$ no intervalo, mais conectada será a rede. Nesta técnica de formação da rede, existem dois parâmetros, $d$ e $\epsilon$. Quando $d$ e $\epsilon$ são grandes, uma rede fortemente conectada será formada, na qual grandes comunidades tendem a ser freqüentes; por outro lado, quando $d$

e $\epsilon$ são pequenos, a rede será esparsa favorecendo a formação de pequenas comunidades.

Partição da rede

A partição da rede é baseada no conceito de modularidade desta, uma medida definida em (Clauset et al., 2004). Seja $e_{ij}$ um elemento da rede definido pela Equação (3.7).

$$e_{ij} = \begin{cases} 1 & \text{se } i \text{ está conectado a } j, \\ 0 & \text{senão.} \end{cases} \tag{3.7}$$

Supondo que o vértice $i$ pertence à comunidade $c_i$, a fração de arestas que pertence à comunidade $c_i$ é dada pela Equação (3.8).

$$\frac{\sum_{ij} e_{ij}\delta(c_i, c_j)}{\sum_{ij} e_{ij}} = \frac{1}{2m}\sum_{ij} e_{ij}\delta(c_i, c_j), \tag{3.8}$$

onde $\delta(c_i, c_j)$ é 1 se $i = j$ e 0 caso contrario. $m = \frac{1}{2}\sum_{ij} e_{ij}$ é o número de arestas na rede. Uma partição é boa se esse valor é próximo de 1, mas uma partição trivial que ponha todos os vértices em apenas um grupo terá valor 1 na Equação (3.8). Para evitar isso a técnica subtrai o efeito da aleatoriedade.

Nesse sentido segundo (Clauset et al., 2004), a modularidade $Q$ é definida na Equação (3.9).

$$Q = \frac{1}{2m}\sum_{ij}(e_{ij} - \frac{K_i K_j}{2m})\delta(c_i, c_j), \text{ se } e_{ij} \neq 0, \tag{3.9}$$

onde $K_i$ é o grau do vértice $i$, e $\frac{K_i K_j}{2m}$ é a probabilidade de uma aresta existir entre os vértices $i$ e $j$ mas tomando em conta os graus de $i$ e $j$.

O algoritmo começa com cada vértice sendo seu próprio grupo, e vai juntando grupos em pares, escolhendo em cada passo aqueles grupos que resultam em maior incremento para a modularidade $Q$. É possível então montar um dendograma com o resultado de cada iteração por ser um algoritmo hierárquico aglomerativo. A Figura 3.9 mostra o resultado desta técnica.

### 3.4.3 Agrupamento de dados utilizando técnica de movimentação de vértices em redes complexas

Esse trabalho se baseia na detecção de comunidades em redes complexas. Foi proposto por (de Oliveira et al., 2008), o algoritmo está dividido em duas fases: modelamento do conjunto de dados como uma rede, e divisão iterativa da rede. Essa técnica pertence aos algoritmos hierárquicos divisivos, onde o comportamento dos vértices na rede é como se fossem partículas em um sistema auto-controlado. Cada vértice tem um ângulo inicial que é atualizado segundo o ângulo dos seus vizinhos. Depois do sistema convergir, os

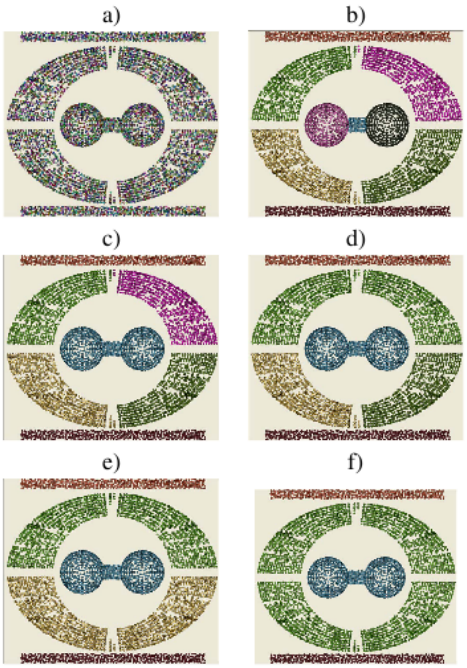



Figura 3.9: (a)Rede artificial com 7320 vértices. A modularidade atinge um valor máximo $Q = 0.8$ quando 9 grupos são formados; (b) 9 grupos formados; (c) 7 grupos formados; (d) 6 grupos formados; (e) 5 grupos formados; (f) 4 grupos formados. Figura obtida de (Silva & Zhao, 2007)

.

vértices de cada diferente grupo ficam restritos a um determinado ângulo. A Figura 3.11 mostra o processo de atualização de ângulos e convergência do algoritmo para a rede da Figura 3.10.

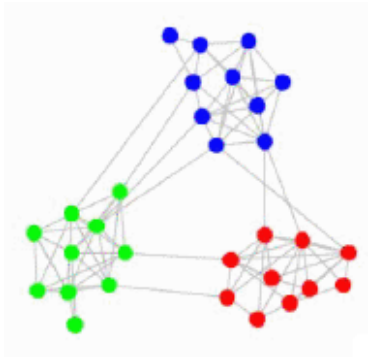


Figura 3.10: Rede original. Vértices com cores diferentes pertencem a comunidades distintas. Figura obtida de (de Oliveira et al., 2008)

.

Formação da rede

A técnica utiliza o conceito dos K-vizinhos mais próximos para formar a rede a partir de um conjunto de dados como em (Karypis et al., 1999), na qual um vértice $v_i$, que

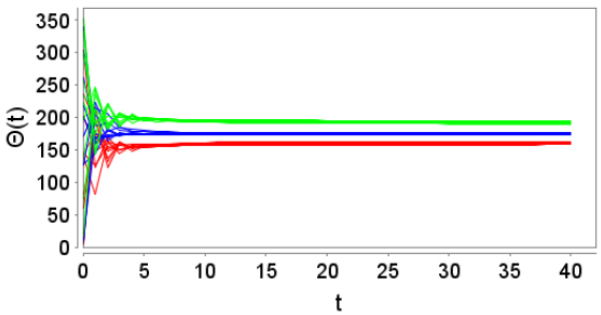



Figura 3.11: Evolução do processo de atualização de ângulos dos vértices para a rede da Figura 3.10. Figura obtida de (de Oliveira et al., 2008)

.

representa um elemento do conjunto de dados, é conectado aos seus $K$-vizinhos mais próximos, formando assim uma rede. Devido a que a escolha de valores altos de $K$ conecta fortemente dois grupos separados, e valores baixos de $K$ geram uma baixa conexão entre elementos de grandes grupos, é que se introduz um peso às arestas, definido como a distância Euclidiana entre dois objetos do conjunto de dados, Equação (3.10).

$$d_{ij} = \|v_j - v_i\|. \tag{3.10}$$

Assim, embora a rede esteja fortemente conectada, o algoritmo irá dar mais importância a vértices mais próximos.

Divisão iterativa da rede

Nessa fase o algoritmo vai atualizando o ângulo de cada vértice segundo o ângulo dos seus vizinhos, visando aproximar vértices que pertencem ao mesmo grupo. Inicialmente o ângulo de cada vértice $\theta_i(t = 0)$ é um valor aleatório entre $[0, 2\pi)$. A dinâmica de atualização dos ângulos é definido pela Equação (3.11).

$$\theta_i(t+1) = \theta_i(t) + \eta_i(t) \left[ \frac{\sum_{j=1}^{M_i} w_{ij}\theta_j(t)}{\sum_{j=1}^{M_i} w_{ij}} - \theta_i(t) \right] ., \tag{3.11}$$

onde $M_i$ é o número de vizinhos de $v_i$, $\eta_i(t)$ é a taxa de movimentação de $v_i$ no instante $t$, e $w_{ij}$ é o peso que representa a influencia de $v_j$ em $v_i$.

$$w_{ij} = CN(v_i, v_j) \times SN(v_i, v_j), \tag{3.12}$$

sendo que $CN(v_i, v_j)$ dá mais importância aos vértices que estejam mais próximos de $v_i$ e é definida na Equação (3.13), $\alpha$ é um parâmetro que define a importância dos vizinhos.

$$CN(v_i, v_j) = e^{-\alpha d_{ij}}. \tag{3.13}$$

$SN(v_i, v_j)$ dá mais importância a um vértice $v_j$ que possua mais vizinhos em comum

com $v_i$. A Equação (3.14) define $SN(v_i, v_j)$, nesta $c(v_i, v_j)$ é o número de vizinhos comuns compartilhados por $v_i$ e $v_j$.

$$SN(v_i, v_j) = \frac{c(v_i, v_j)}{M_i}. \tag{3.14}$$

A taxa de movimentação de $v_i$ é dada pela Equação (3.15), $\beta$ é um parâmetro definido pelo usuário, e $\sigma_i$ é o desvio padrão dos ângulos dos vizinhos de $v_i$. Inicialmente como os ângulos dos vértices são definidos de maneira aleatória, $\sigma_i$ é um valor grande, e $\eta_i(t)$ adquire valores próximos a 1. Após certo número de iterações o desvio padrão será pequeno e $\eta_i(t)$ tenderá a 0, indicando que o algoritmo convergiu.

$$\eta_i(t) = e^{\frac{-\beta}{\sigma_i}}. \tag{3.15}$$

O algoritmo de agrupamento e a regra de atualização de ângulos estão descritos nos Algoritmos 1 e 2.

**Algorithm 1:** Algoritmo de Agrupamento

**Require**: Uma rede não vazia *net*

No começo, a rede *net* é um só grupo, e é adicionado ao conjunto de grupos C
**repeat**
  **forall the** *grupos* $c_k \in C$ **do**
    **if** $\eta_{c_k} > 1$ **then**
      $VO_{c_k} \leftarrow$ `atualizarAngulo(`$c_k$`)`
      $id_{c_k} \leftarrow$ `argMax` $(VO_{c_k}[j] - VO_{c_k}[j+1]), \forall j \in [1, \eta_k)$
      $maxDif_{c_k} \leftarrow VO_{c_k}[id] - VO_{c_k}[id+1]$
    **end**
  **end**
  $c_{max} \leftarrow c_k$ com o valor máximo para $maxDif_{c_k}$, $\forall c_k \in C$
  $c_1 \leftarrow$ vértices com ângulos em $VO_{c_{max}}[1]$ até $VO_{c_{max}}[id_{c_{max}}]$
  $c_2 \leftarrow$ vértices com ângulos em $VO_{c_{max}}[id_{max}+1]$ até $VO_{c_{max}}[\eta_{c_k}]$
  Adicionar $c_1$ e $c_2$ ao conjunto de grupos C
**until** $\eta_{c_k} = 1$, $\forall c_k$;

## 3.5 Validação de agrupamento de dados

Devido ao fato de que os algoritmos de agrupamento sempre vão formar grupos, mesmo que os dados não tenham estruturas de grupos, é que se faz imprescindível ter algum mecanismo de avaliação da qualidade dos grupos encontrados por um algoritmo. Para isto geralmente é utilizado índices estatísticos, os quais permitem avaliar quantitativamente a relevância dos grupos obtidos.

Existem três tipos de critérios de validação: critérios internos, externos e relativos. Os critérios internos servem para avaliar a qualidade dos grupos de uma partição, eles avaliam quão bons são os grupos obtidos do conjunto de dados de entrada, quer dizer, medem quão

**Algorithm 2:** atualizarAngulo

**Require**: $\eta_{c_k} > 1$

**forall the** *vértice* $v_i \in c_k$ **do**
  Inicializar $\theta_i(t = 0)$ com um valor aleatório entre $[0, 2\pi)$
**end**
**repeat**
  **forall the** *vértice* $v_i \in c_k$ **do**
    Atualizar $\theta_i(t + 1)$ de acordo com a Equação (3.11)
  **end**
  $t \leftarrow t + 1$
**until** *atingir estabilidade*;
$VO \leftarrow$ o conjunto ordenado de vértices, em ordem descendente
**return** $VO$

separados estão os diferentes grupos e quão próximos estão os elementos de cada grupo. Os critérios de avaliação interna lidam também com o problema da estimativa do número de grupos. Quando é preciso avaliar a partição obtida em relação a uma partição externa, por exemplo quando os grupos são conhecidos ou são estimados por um especialista, nesse caso é utilizado os critérios de avaliação externa, eles geralmente são empregados para avaliar a eficácia de um algoritmo de agrupamento de dados, a partição obtida pelo algoritmo deve ser o mais similar possível com a partição real dos dados, esse grau de semelhança é medido pelos índices externos. Por último, os critérios de avaliação relativa permitem comparar duas partições obtidas a partir do mesmo ou de diferentes algoritmos de agrupamento, servem para identificar qual é a melhor partição dentre um conjunto de partições do mesmo conjunto de dados. Os índices relativos são amplamente utilizados para encontrar a melhor partição do conjunto de partições geradas pelos algoritmos de agrupamento de dados hierárquicos, obtendo assim o melhor corte no dendograma gerado pelo algoritmo.

A seguir são apresentados alguns índices de critérios de validação: o índice Jaccard, o índice Rand corrigido e o índice da Silhueta.

### 3.5.1 Índice jaccard

O índice Jacccard também chamado coeficiente de similaridade Jaccard é um critério de validação externa, assim, ele compara uma partição obtida por um algoritmo com uma partição real externa e devolve o grau de similaridade entre as duas partições.

Seja $P$ e $P'$ as duas partições, a similaridade $J$ entre elas no índice Jaccard é definida na Equação (3.16).

$$J = \frac{SS}{SS + SD + DS}, \tag{3.16}$$

onde cada termo é dado por:

- $SS$ = número de pares de objetos que pertencem ao mesmo grupo em ambas

partições.

- $SD$ = número de pares de objetos que pertencem ao mesmo grupo na partição $P$ mas não pertencem ao mesmo grupo na partição $P'$.
- $DS$ = número de pares de objetos que não pertencem ao mesmo grupo na partição $P$ mas pertencem ao mesmo grupo na partição $P'$.

Uma característica do índice Jaccard é que tende a favorecer partições com muitos grupos.

### 3.5.2 Índice rand corrigido

O índice Rand corrigido é uma versão melhorada do índice Rand que leva em conta o efeito do acaso, e por isso é mais rigoroso que o índice Rand na validação da qualidade das partições. O índice $CR$ (do inglês *Corrected Rand*) mede a similaridade entre duas partições $P$ e $P'$, esta medida aumenta com quantidade de pares de objetos que pertencem ao mesmo grupo nas duas partições, e diminui com a quantidade de pares de objetos que pertencem ao mesmo grupo em uma partição mas não na outra. Valores próximos de 0 para esta medida indicam partições aleatórias, enquanto valores próximos a 1 são obtidos por partições mais relevantes.

O índice Rand corrigido é definido na Equação (3.17), onde $k_P$ e $k_{P'}$ são o número de grupos de $P$ e $P'$, $n$ é o número de objetos do conjunto de dados, $n_i$ é o número de objetos do grupo $C_i \in P$, $n_j$ é o número de objetos do grupo $C_j \in P'$ e $n_{ij}$ é o número de objetos que pertencem ao grupo $C_i \in P$ e $C_j \in P'$.

$$CR = \frac{A - EI}{B - EI}, \tag{3.17}$$

onde cada termo é definido como:

$$\begin{array}{l} A = \sum_{i=1}^{k_P} \sum_{j=1}^{k_{P'}} \binom{2}{n_{ij}} \\ B = \left[\sum_{i=1}^{k_P} \binom{2}{n_i} \sum_{j=1}^{k_{P'}} \binom{2}{n_j}\right] / 2 \\ EI = \left[\sum_{i=1}^{k_P} \binom{2}{n_i} \sum_{j=1}^{k_{P'}} \binom{2}{n_j}\right] / \binom{2}{n} \end{array}, \tag{3.18}$$

onde $EI$ (do inglês *Expected Index*) é o termo de ajuste que serve para corrigir o efeito do acaso nas partições $P$ e $P'$.

### 3.5.3 Índice da silhueta

O índice da Silhueta é um critério de validação interna que leva em conta a coesão e separação dos grupos, ele avalia a qualidade dos grupos formados em uma partição. Esse índice pode ser utilizado também como um índice relativo, pois é possível comparar duas partições comparando os valores da Silhueta para cada partição, quanto maior o valor da Silhueta, melhor a partição.

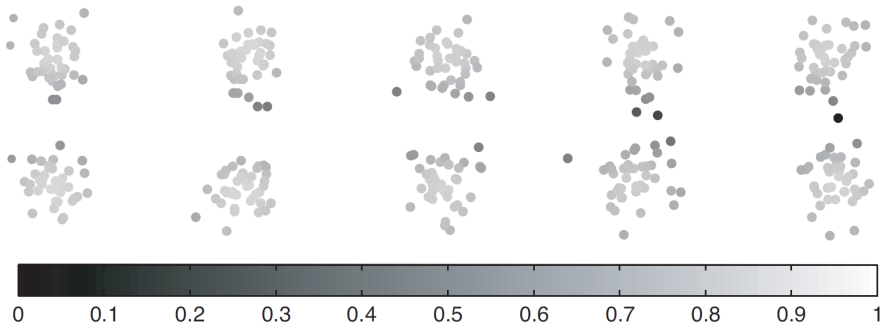



Figura 3.12: Índice da Silhueta para cada elemento de um conjunto de dados com 10 grupos. Os pontos mais escuros indicam menor valor da Silhueta. Figura adaptada de (Pang-Ning Tan, 2006)

O índice da Silhueta para cada elemento $i$ do conjunto de dados é dado pela Equação (3.19).

$$S_i = \frac{(b_i - a_i)}{max(a_i, b_i)}, \tag{3.19}$$

onde:

- $a_i$ é a média das distâncias entre o elemento $i$ e todos os elementos do seu grupo.

- $b_i$ é o valor mínimo das médias das distâncias do elemento $i$ para os grupos restantes.

O termo $b_i$ é considerado como a distância média do elemento $i$ e os elementos do grupo mais próximo de $i$, sendo que $i$ não pertence a este último. O valores da Silhueta podem variar entre $-1$ e 1, valores positivos altos da Silhueta são desejáveis, isso acontece quando $(a_i < b_i)$ e $a_i$ é próximo de 0.

O índice da Silhueta para cada grupo pode ser calculado como a média do índice da Silhueta para cada elemento do grupo, e o índice que indica a qualidade da partição pode ser calculado como a média do índice da Silhueta para todos os elementos do conjunto de dados. A Figura 3.12 mostra o índice da Silhueta para um conjunto de dados com 10 grupos, as cores mais escuras indicam valores menores da Silhueta para cada elemento.

Uma aplicação importante do índice da Silhueta é na estimativa do número de grupos de um conjunto de dados, assim depois de obter varias partições de diferente número de grupos com algum algoritmo de agrupamento, pode-se calcular o índice para cada partição, assim o número de grupos estimado é o número de grupos da partição com maior valor da Silhueta. A Figura 3.13 mostra o valor da Silhueta para diferentes partições com diferente número de grupos de um mesmo conjunto de dados. É importante ressaltar que o índice da Silhueta tende a favorecer partições com os grupos bem separados.

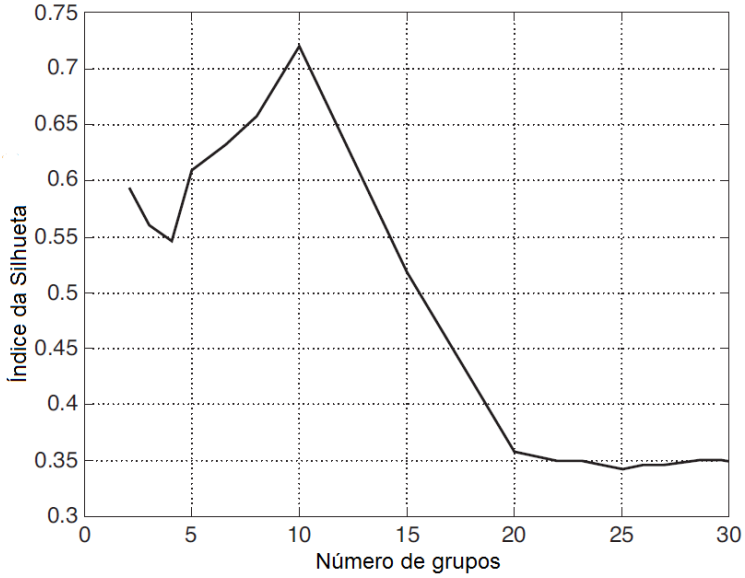



Figura 3.13: Índice da Silhueta médio para partições com diferente número de grupos do conjunto de dados da Figura 3.12. Figura adaptada de (Pang-Ning Tan, 2006)

## 3.6 Considerações finais

Neste capítulo foram apresentados alguns conceitos básicos de agrupamento de dados relevantes para o presente trabalho. Foram descritas algumas técnicas em maior detalhe, como o algoritmo $K$-médias, os algoritmos hierárquicos: Single Linkage, Complete Linkage e Average Linkage, assim como o algoritmo baseado em grafos CHAMELEON proposto em (Karypis et al., 1999), que consiste em duas etapas: uma etapa divisiva e outra etapa aglomerativa. É interessante ver que a formação de uma rede a partir dos dados de entrada, permite ao algoritmo CHAMELEON encontrar grupos com diferentes formas, isto devido a que, na formação da rede, com os dados de entrada, o algoritmo está levando em conta a informação topológica dos dados, em outras palavras, os relacionamentos entre os dados, e não somente a distância ou densidade dos dados. Um fator critico nos algoritmos de agrupamento de dados baseados em grafos é a construção da rede, a eficiência desses algoritmos é fortemente afetada pela rede construída, está precisa ser o mais esparsa possível, sem perder a estrutura de comunidades da rede, além disso, a rede precisa ser conectada.

Além disso, foram descritos dois trabalhos importantes de agrupamento de dados baseados em detecção de comunidades em redes complexas. O primeiro deles trabalha somente com dados unidimensionais, como imagens em tons cinza, sendo aplicado na segmentação de imagens. O segundo trabalho é baseado na movimentação dos vértices da rede em uma circunferência com convergência pontual para detectar as comunidades, uma vantagem da técnica é que converge em poucas iterações, mas acredita-se que essa técnica possa ter uma desvantagem, pois várias comunidades podem convergir em um

mesmo ponto na circunferência pelas ligações entre comunidades, dificultando assim a detecção dos grupos reais.

Finalmente, foram apresentados os tipos de critérios de avaliação de de algoritmos de agrupamento de dados e alguns índices de avaliação de agrupamento de dados tais como: o índice Jaccard, o índice Rand Corrigido e o índice da Silhueta.

Capítulo

# 4

# Agrupamento de dados e detecção de simplicidade utilizando a medida de tempo de consenso em redes complexas

As redes têm sido muito utilizadas no modelamento e análise de sistemas complexos com muitos agentes interagindo, pois elas podem representar bem as características e a natureza desse tipo de sistemas, revelando a estrutura dos dados. Redes complexas podem ser definidos como grafos de grande escala com uma distribuição do grau não trivial. Uma propriedade importante em redes complexas estudada nos últimos anos é a "estrutura de comunidade", onde cada comunidade é um grupo de vértices densamente conectados.

Análise de agrupamento de dados é uma tarefa muito importante para a mineração de dados. Existem diversos algoritmos para agrupamento de dados, mas no caso de grupos com diferentes formas, tamanhos e densidades, muitos algoritmos têm dificuldade em lidar com esses grupos (Karypis et al., 1999), um exemplo é o $K$-médias que identifica apenas grupos com forma esférica e de tamanhos similares. Já algoritmos baseados em grafos conseguem identificar grupos com essas características, onde primeiramente é formado uma rede a partir do conjunto de dados para depois ser aplicado um algoritmo de detecção de comunidades, no final cada comunidade representa um grupo no conjunto de dados original. Embora esses algoritmos consigam identificar grupos com as características descritas, eles dependem muito da rede construída a partir do conjunto de dados, assim essa etapa é muito importante e geralmente é utilizada a técnica de $K$-vizinhos mais próximos, que em alguns casos pode construir redes desconectadas ou redes muito densas com as comunidades misturadas, que dificulta o processo de detecção de comunidades.

Outro tópico de análise de dados é a análise de simplicidade em redes complexas,

esse assunto é de grande importância, pois permite caracterizar a complexidade de uma rede, e melhor ainda permite identificar regiões homogêneas ou simples na rede, regiões que podem ser de grande importância para a rede e que podem revelar algum padrão interessante nos dados. Existem muito poucos trabalhos nesse tema, em (da F. Costa & Rodrigues, 2009) foi proposto um método muito interessante para detectar regiões simples em redes complexas, além de um índice de simplicidade. Esse método é baseado na escolha de algumas propriedades da rede, as quais geralmente são medidas locais que podem não caracterizar bem a rede, assim a escolha das medidas corretas é um fator crítico a ser considerado nesse método.

Esse Capítulo apresenta a medida de distância baseada no tempo de consenso em redes complexas, medida que pode ser utilizada em distintas tarefas de análise de dados. A medida de distância proposta, junto com um método de detecção de comunidades são apresentados na Seção 4.1. Um método de agrupamento de dados baseado no método de detecção de comunidades proposto, além de uma técnica de construção de redes esparsas e conectadas a partir de um conjunto de dados são descritos na Seção 4.2. Por último a Seção 4.3 apresenta um método de detecção de regiões simples baseado na medida de distância proposta. Cada seção mostra também os resultados obtidos com diferentes redes e conjuntos de dados, artificiais e reais.

## 4.1 Detecção de comunidades baseado no tempo de consenso em redes complexas

Nesta seção é apresentado uma medida de distância para redes complexas proposta, baseada no tempo de consenso da rede. Além disso é apresentado um método proposto para a detecção de comunidades baseado nessa medida de distância. A medida de distância proposta consiste em considerar cada vértice como um agente (com um estado inicial) em um sistema dinâmico de agentes acoplado e medir o tempo (em número de iterações) que um vértice leva para chegar no consenso (atingir certo estado desejado) com ajuda de um vértice como líder.

### 4.1.1 Medida de distância de tempo de consenso

Será proposto uma medida de distância baseado nos conceitos de consenso e controle focalizado apresentados na Seção 2.5. Considerando a Equação (2.25) no caso em que $f(\cdot) = 0$, o controle focalizado é reduzido ao problema de consenso na presença de líderes com uma trajetória de referência $s(t) = \bar{x}$, na qual $\bar{x}$ é o estado estacionário desejado.

O problema de consenso na presença de um líder em tempo discreto, com a função de acoplamento interno $h(x) = x$ e o estado estacionário desejado $s(t) = \bar{x}$ é definido na Equação (4.1).

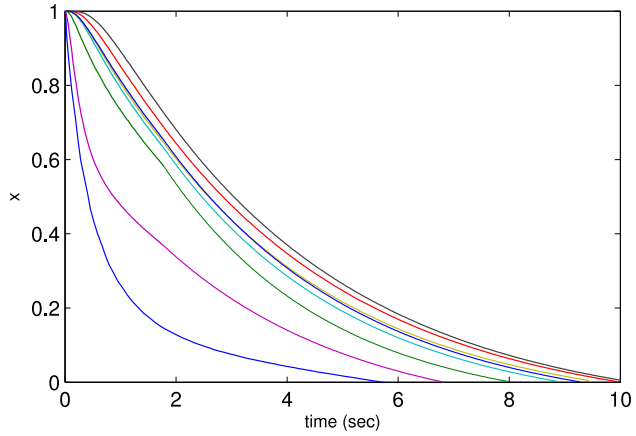



Figura 4.1: Evolução dos estados em uma rede aleatória com 8 vértices atingindo um consenso na presença de um líder (primeiro vértice), $\bar{x} = 0$, $\beta = 1$.

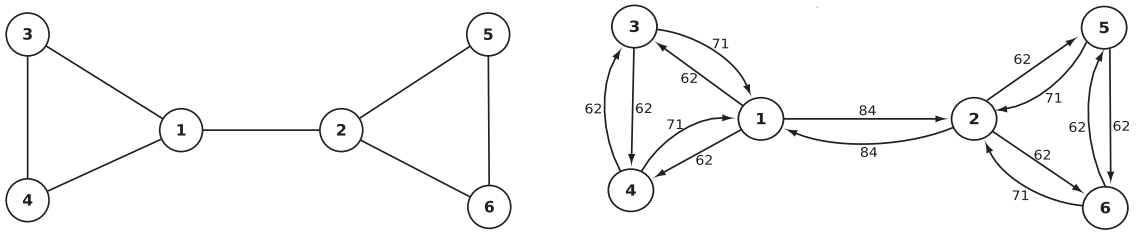
$$x_i(t+1) = x_i(t) - \epsilon \sum_{j=1}^{N} l_{ij} x_j(t) + \epsilon u_i(t), \tag{4.1}$$

onde $\epsilon > 0$ é o parâmetro de passo para cada iteração em tempo discreto, $u_i(t) = k_i(\bar{x} - x_i(t))$ e no caso do vértice controlado (o líder) $k_i = z$ sendo $z \neq 0$, e para o resto dos vértices da rede o parâmetro de controle é $k_i = 0$.

Assim pode ser definido uma medida de distância $d_{ij} = t_j$, como o número de passos de tempo (iterações) que um vértice $j$ leva para atingir o estado estacionário $\bar{x}$, isso acontece quando $x_j(t) = \bar{x}$ no passo de tempo $t = t_j$ em um problema de consenso na presença do vértice $i$ sendo um líder. Cada vértice na rede é estabilizado para $\bar{x}$ em tempos diferentes por um só controlador, o vértice $i$. Como a medida proposta é o tempo que cada vértice leva para atingir $\bar{x}$, precisamos que todos os vértices possuam o mesmo estado inicial $x_1(0), x_2(0), ..., x_N(0) = \beta$ com $\beta \neq \bar{x}$. Na Figura 4.1 é mostrado a evolução dos estados no tempo contínuo em uma rede aleatória com 8 vértices em um problema de consenso com o primeiro vértice sendo controlado. Note-se que cada vértice atinge $\bar{x} = 0$ em tempos diferentes.

Pode-se calcular $d_{ij}$ para qualquer vértice $j$ fazendo o vértice $i$ um líder e medindo o número de iterações que $j$ leva para atingir $\bar{x} = 0$, fazendo o mesmo procedimento para todos os vértices $i = 1, 2, ..., N$ na rede obtemos a matriz de distâncias assimétricas $D = [d_{ij}]$. Por exemplo a matriz de distâncias assimétricas para a rede na Figura 4.2(a) é:

$$D = \begin{bmatrix} 0 & 84 & 62 & 62 & 95 & 95 \\ 84 & 0 & 95 & 95 & 62 & 62 \\ 71 & 101 & 0 & 62 & 115 & 115 \\ 71 & 101 & 62 & 0 & 115 & 115 \\ 101 & 71 & 115 & 115 & 0 & 62 \\ 101 & 71 & 115 & 115 & 62 & 0 \end{bmatrix} \tag{4.2}$$




(a) (b)

Figura 4.2: (a) Uma rede simples com 6 vértices e (b) as distâncias assimétricas entre vizinhos.

A Figura. 4.2(b) mostra as distâncias assimétricas entre vizinhos para uma rede simples com 6 vértices, $\bar{x} = 0$, $\beta = 1$, $\epsilon = 0.015$, $z = 0.15$.

### 4.1.2 Método de detecção de comunidades

É possível encontrar comunidades em uma rede complexa aplicando algoritmos de agrupamento de dados, sobre a matriz de distâncias assimétricas obtidas $D$, algoritmos tais como Single Linkage, Complete Linkage e Average Linkage (Jain et al., 1999). Aplicando algum desses algoritmos sobre $D$ obtemos uma estrutura hierárquica de partições que pode ser representada em um dendograma. Para poder utilizar a matriz de distâncias assimétricas $D$ com algum desses algoritmos precisamos modificar a matriz $D$ para uma matriz de distâncias simétrica, isto devido ao fato de que os algoritmos de agrupamento de dados hierárquicos precisam de uma matriz de distâncias simétrica como entrada. Uma abordagem simples é pegar a média das distâncias entre dois vértices $d_{ij}$ e $d_{ji}$. Assim a matriz de distâncias simétricas $D_s$ da rede é definida como:

$$D_s = (D + D^T)/2. \tag{4.3}$$

Assim os passos a seguir para a detecção de comunidades utilizando a medida de distância proposta são:

- Calcular a matriz de distâncias assimétricas $D$ como descrito na Subseção 4.1.1.
- Modificar a matriz $D$ para $D_s$ segundo a Equação (4.3).
- Aplicar algum dos algoritmos hierárquicos descritos.
- Escolher alguma partição do dendograma obtido, segundo o número de grupos desejados.

A Figura 4.3 mostra o dendograma da aplicação do Average Linkage sobre a matriz de distâncias simétricas $D_s$ obtida a partir da matriz (4.2). Note-se as duas comunidades claramente identificadas da rede simples da Figura 4.2.

É importante ressaltar que os valores para os diferentes parâmetros do método proposto para o cálculo das distâncias ($\bar{x}$, $\beta$, $\epsilon$ e $z$) não afetam o resultado do agrupamento,

desde que os parâmetros somente afetem a velocidade da convergência de todos os vértices, assim, se são escolhidos outros valores para os parâmetros, todas as distâncias obtidas serão afetadas na mesma proporção e o resultado do algoritmo de agrupamento será o mesmo.

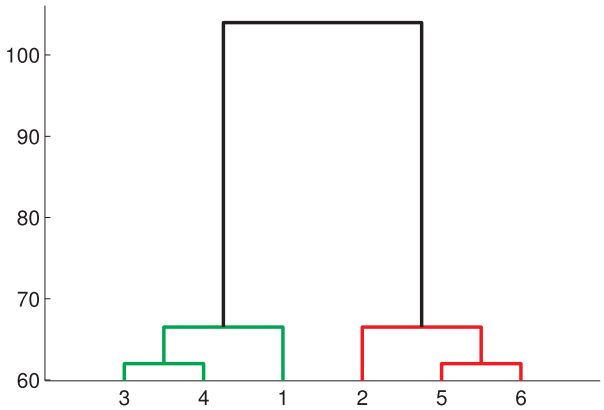



Figura 4.3: Dendograma mostrando as duas comunidades encontradas pelo Average Linkage sobre a matriz $D_s$ obtida a partir da matriz de distâncias 4.2.

### 4.1.3 Resultados obtidos em redes artificiais

O método de detecção de comunidades proposto foi aplicado em algumas redes artificiais que apresentam estrutura de comunidade para verificar a sua eficácia. Para todas as redes testadas foi calculada a medida de distância proposta com $\bar{x} = 0$, $\beta = 1$, $\epsilon = 0.015$, $z = 0.15$, e foi aplicado algum dos algoritmos de agrupamento de dados hierárquicos sobre as distâncias calculadas, com o objetivo de encontrar as comunidades como descrito na Subseção 4.1.2.

A primeira rede utilizada é uma rede em forma de árvore com 15 vértices, embora a topologia desta rede seja bem simples, a rede possui estrutura de comunidades. Calculando as distâncias com o método proposto e aplicando o Average Linkage sobre elas obtemos 5 comunidades. A Figura 4.4 mostra as 5 comunidades encontradas na rede e o correspondente dendograma obtido pode ser apreciado na Figura 4.5. Note-se na Figura 4.5 que a partição com 5 comunidades tem o maior valor de modularidade no dendograma.

A segunda rede testada é uma rede apresentada em (Fortunato & Barthlemy, 2007), essa rede tem a característica de ser a rede mais modular possível com 25 arestas (Figura 4.6(a)) de acordo com as restrições mencionadas em (Fortunato & Barthlemy, 2007). Aplicando o método proposto com o Average Linkage obtemos o maior valor de modularidade $Q = 0.6$, isto quando a rede é dividida em 5 comunidades que indica que existe uma forte estrutura de comunidades. O dendograma obtido é mostrado na Figura 4.6(b).

Por último foi gerado uma seqüência de redes incrementando o grau de mistura das comunidades. Cada rede tem $N = 128$ vértices, grau médio da rede $\langle k \rangle = 16$ e 4 comunidades com 32 vértices cada. Cada rede é construída como segue:

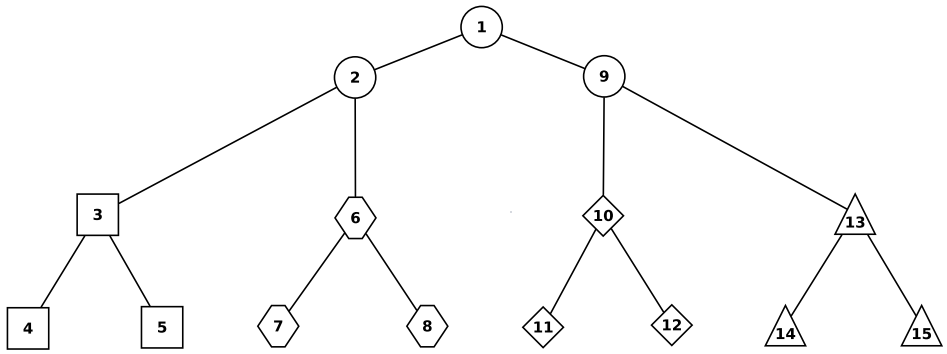



Figura 4.4: As 5 comunidades encontradas na rede em forma de árvore. Cada comunidade esta representada com vértices de diferente forma.

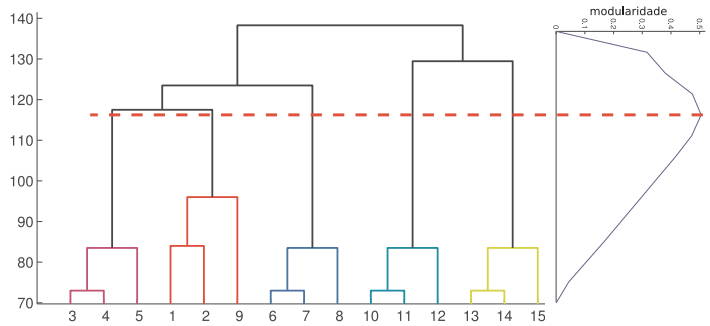
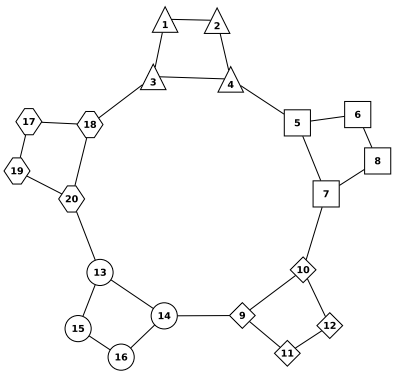
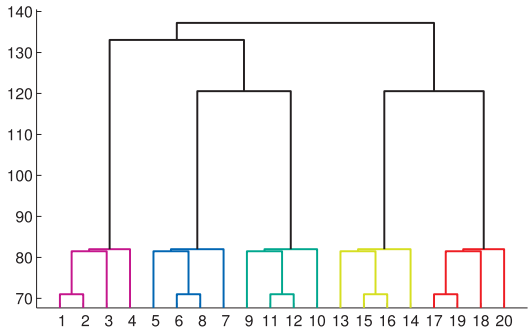



Figura 4.5: Dendograma e modularidade para a rede em forma de árvore. O pico na modularidade (linha pontilhada) corresponde às 5 comunidades encontradas pelo algoritmo proposto.



(a)




(b)

Figura 4.6: (a) A rede mais modular com 25 arestas. Existem 5 comunidades, cada comunidade é representada com uma forma diferente de vértice. (b) As 5 comunidades são claramente identificadas no dendograma obtido pelo método proposto.

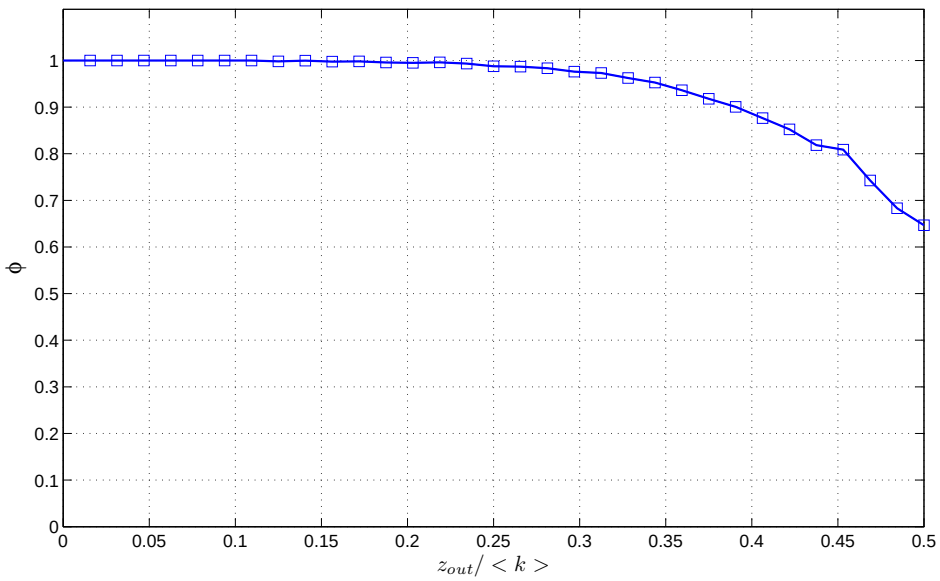



Figura 4.7: Fração de vértices corretamente classificados utilizando o método proposto em redes com 4 comunidades, $N = 128$ e $\langle k \rangle = 16$. O eixo $x$ representa a fração de arestas entre comunidades. Cada ponto da linha na figura representa a média de 50 execuções do algoritmo com uma rede gerada aleatoriamente em cada iteração.

Um par de vértices selecionados aleatoriamente na mesma comunidade, são conectados com probabilidade $p_{in}$, entretanto um par de vértices selecionados aleatoriamente de diferentes comunidades, são conectados com probabilidade $p_{out}$. Os valores de $p_{in}$ e $p_{out}$ são escolhidos de modo a controlar o número de arestas esperado de um vértice para vértices da mesma comunidade $z_{in}$ e o número de arestas esperado de um vértice para vértices de outras comunidades $z_{out}$. Nesse sentido podemos definir a fração de arestas para a mesma comunidade como $z_{in}/\langle k \rangle$ e a fração de arestas para outras comunidades como $z_{out}/\langle k \rangle$. Este modelo de construção de redes modulares para testar algoritmos de detecção de comunidades foi proposto em Newman & Girvan (2004).

A Figura 4.7 mostra a fração de vértices classificados corretamente $\phi$ nas quatro comunidades. Devido ao fato de que as comunidades são geradas de forma aleatória, cada ponto na figura representa a média de 50 realizações. Nota-se que o método proposto de detecção de comunidades obtém bons resultados, mesmo quando as comunidades estão bem misturadas, isto acontece quando $z_{out}/\langle k \rangle = 0.5$, ou seja, em média a metade das arestas de um vértice conectam os vértices da mesma comunidade e a outra metade conectam-se com vértices de outras três comunidades.

### 4.1.4 Resultados obtidos em redes reais

O método proposto de detecção de comunidades também foi aplicado para algumas redes reais. A primeira é a rede do clube de karatê registrada por Zachary (1977). Esta rede é amplamente utilizada para verificar algoritmos de detecção de comunidades, a qual representa a amizade entre 34 membros de um clube de karatê em uma universidade dos

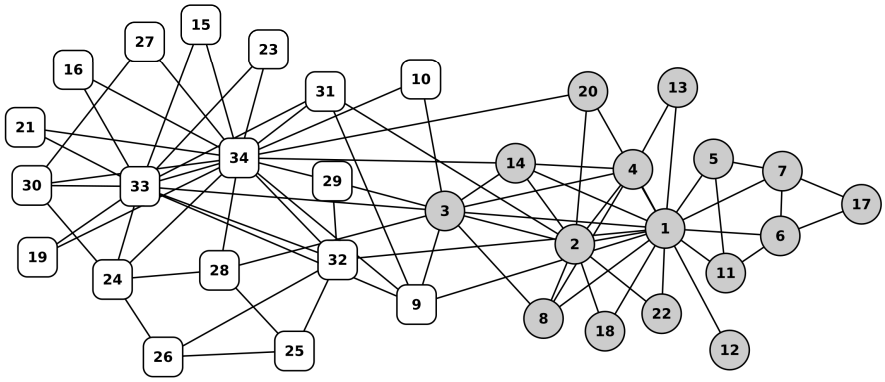



Figura 4.8: A rede do clube de karatê de Zachary mostrando as duas comunidades originais.

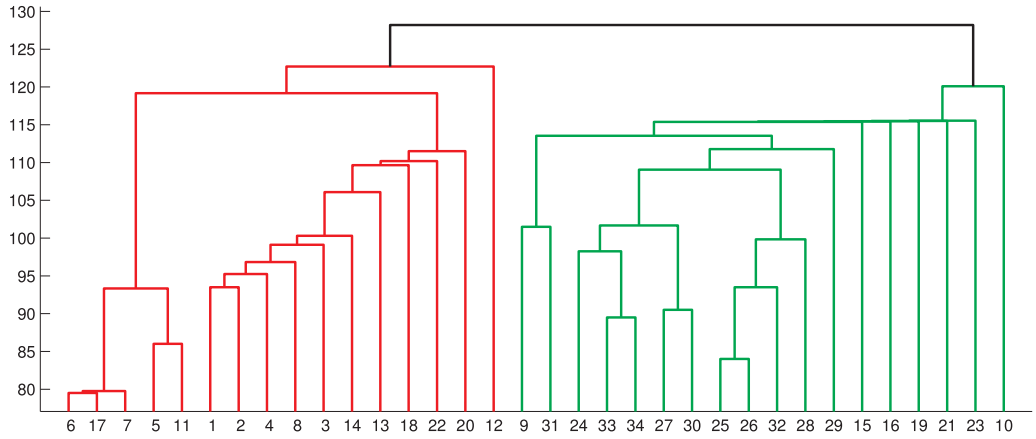



Figura 4.9: Dendograma mostrando as duas comunidades encontradas pela técnica proposta na rede do clube de karatê.

EUA. Por causa de uma disputa, o instrutor do clube (vértice 34) formou um novo clube, e o clube original perdeu alguns membros que se juntaram ao clube do instrutor. Assim alguns membros ficaram com o instrutor do clube e outros ficaram com o administrador do clube (vértice 1). A Figura 4.8 mostra as duas comunidades originais da rede, o dendograma obtido com o Average Linkage, sobre as distâncias obtidas pelo método proposto é mostrado na Figura 4.9, nesta figura pode-se notar que as duas comunidades foram claramente identificadas. É importante mencionar que os vértices 3 e 10 são difíceis de ser classificados corretamente nas suas respectivas comunidades porque esses vértice estão bem no meio das duas comunidades. Por exemplo o vértice 10 possui somente duas arestas, uma aresta para cada comunidade, o vértice 3 é fortemente conectado com 5 arestas para cada comunidade. É comum que muitos algoritmos de detecção de comunidades não sejam capazes de classificar corretamente esses dois vértices.

A segunda rede real é uma rede de interações sociais entre golfinhos registrado por Lusseau (2003), que foi construída mediante a observação de 62 golfinhos na qual as

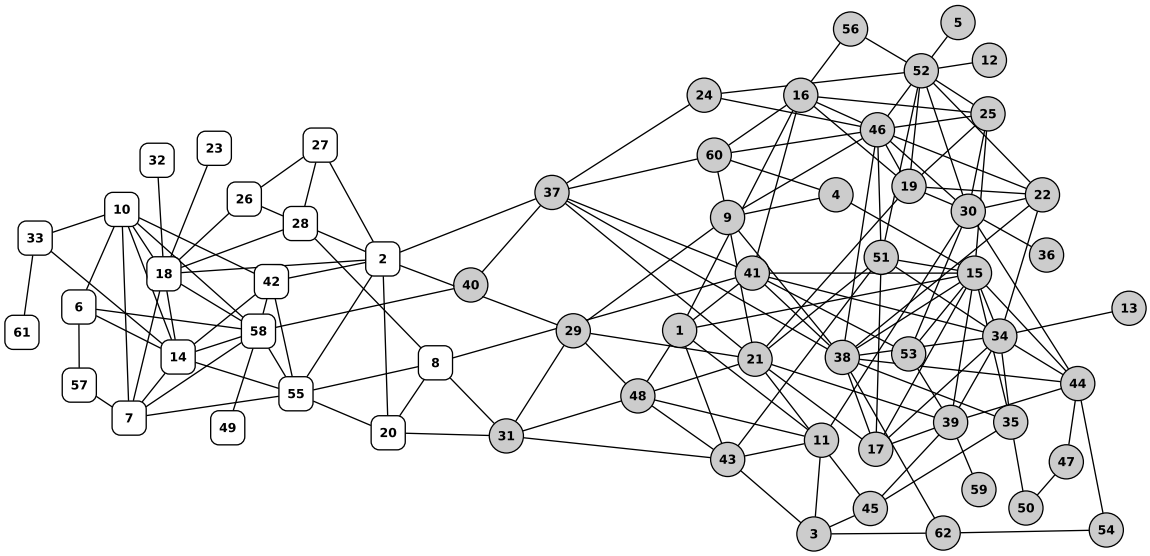



Figura 4.10: A rede de interações sociais entre golfinhos mostrando as duas comunidades originais.

arestas representam associações entre eles, permanecem no mesmo grupo mais freqüentemente que o esperado pelo acaso. Devido à separação de um individuo do estudo, a rede de golfinhos ficou dividida em duas comunidades, este individuo (vértice 37) jogou um papel importante na integração das duas comunidades. A Figura 4.10 mostra as duas comunidades na rede. Para a detecção de comunidades nesta rede foi utilizado o algoritmo Average Linkage com pesos descrito em (Jain et al., 1999) sobre as distâncias calculadas pelo método proposto. O dendograma obtido é mostrado na Figura 4.11. Como pode ser visto no dendograma foram encontradas duas comunidades. A divisão obtida pelo algoritmo proposto difere somente em dois vértices (8 e 20) referente ao caso real observado na Figura 4.10.

## 4.2 Agrupamento de dados via detecção de comunidades

Nesta seção é apresentado o método proposto para agrupamento de dados, que está baseado no método proposto de detecção de comunidades descrito na Seção 4.1 e uma técnica proposta para a construção de uma rede a partir de um conjunto de dados descrito a seguir na Subseção 4.2.1. O método de agrupamento de dados proposto divide-se em duas etapas, a primeira consiste em criar uma rede a partir do conjunto de dados, esta rede precisa ser conectada e é desejável que seja esparsa, o método proposto para a construção da rede é baseado na idéia do algoritmo de agrupamento Single Linkage. A segunda etapa é aplicar o método de detecção de comunidades proposto com uma pequena modificação, isto para evitar que grupos que não estejam conectados na rede sejam agrupados em iterações iniciais, caso que pode acontecer em conjuntos de dados com um número considerável de elementos. A Figura 4.12 mostra o processo de agrupamento de dados via detecção de comunidades em redes complexas.

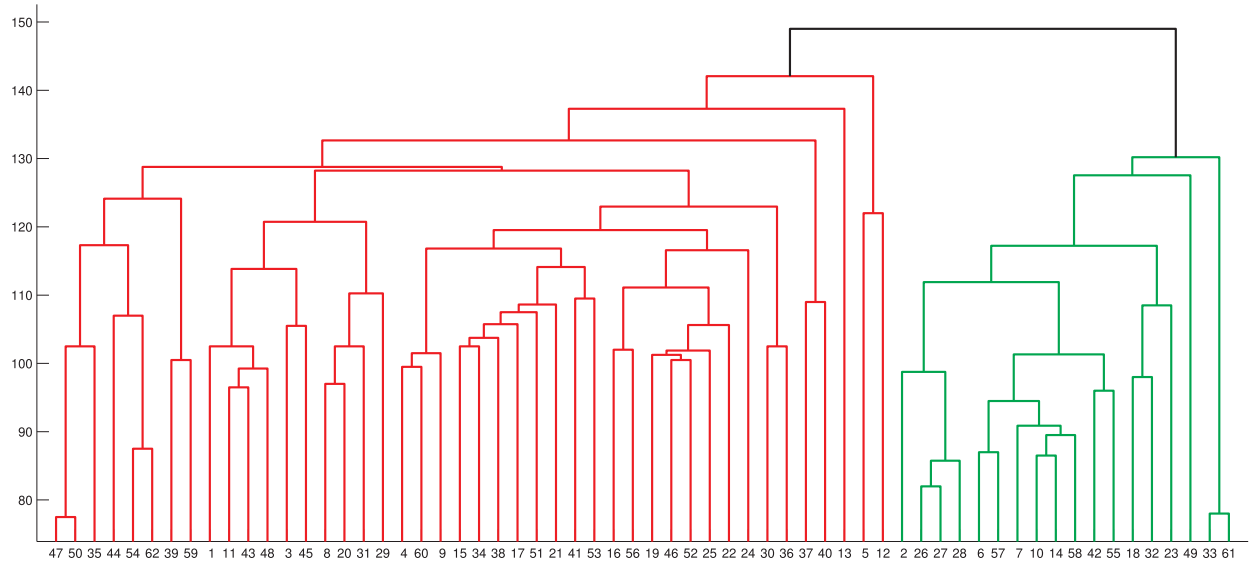



Figura 4.11: Dendograma obtido pelo método proposto na rede de interação social dos golfinhos.

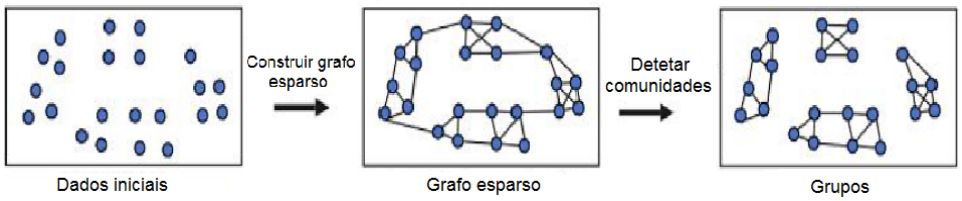



Figura 4.12: Processo de agrupamento de dados via detecção de comunidades.

### 4.2.1 Formação da rede

Tendo-se um conjunto de dados é preciso construir uma rede com esses para aplicar o algoritmo de detecção de comunidades e assim poder obter os grupos. Essa representação em rede do conjunto de dados, tem a vantagem de revelar a estrutura topológica dos dados. Duas características são desejáveis nesta rede, que seja conexa e esparsa, mas com estrutura de comunidade definida. Uma abordagem possível é utilizar a construção da rede mediante os $K$-vizinhos mais próximos como no algoritmo de agrupamento de dados CHAMELEON, onde cada elemento do conjunto de dados representa um vértice e cada vértice é conectado com os seus $K$ vizinhos mais próximos. Um problema com essa abordagem é que $K$ tem de ser suficientemente grande para a rede não ficar desconectada e em alguns casos ao utilizar um $K$ maior para evitar esse efeito a rede fica muito densa, com muitas arestas entre os vértices, dificultando assim o processo de detecção de comunidades. A Figura 4.13 mostra uma conjunto de dados de 300 elementos formando 3 grupos, e o resultado de aplicar o K-vizinhos mais próximos para a construção da rede com diferentes valores de $K$. Note-se que mesmo $K$ sendo muito grande a rede fica muito densa e desconectada, somente quando $K = 33$ a rede fica conectada.

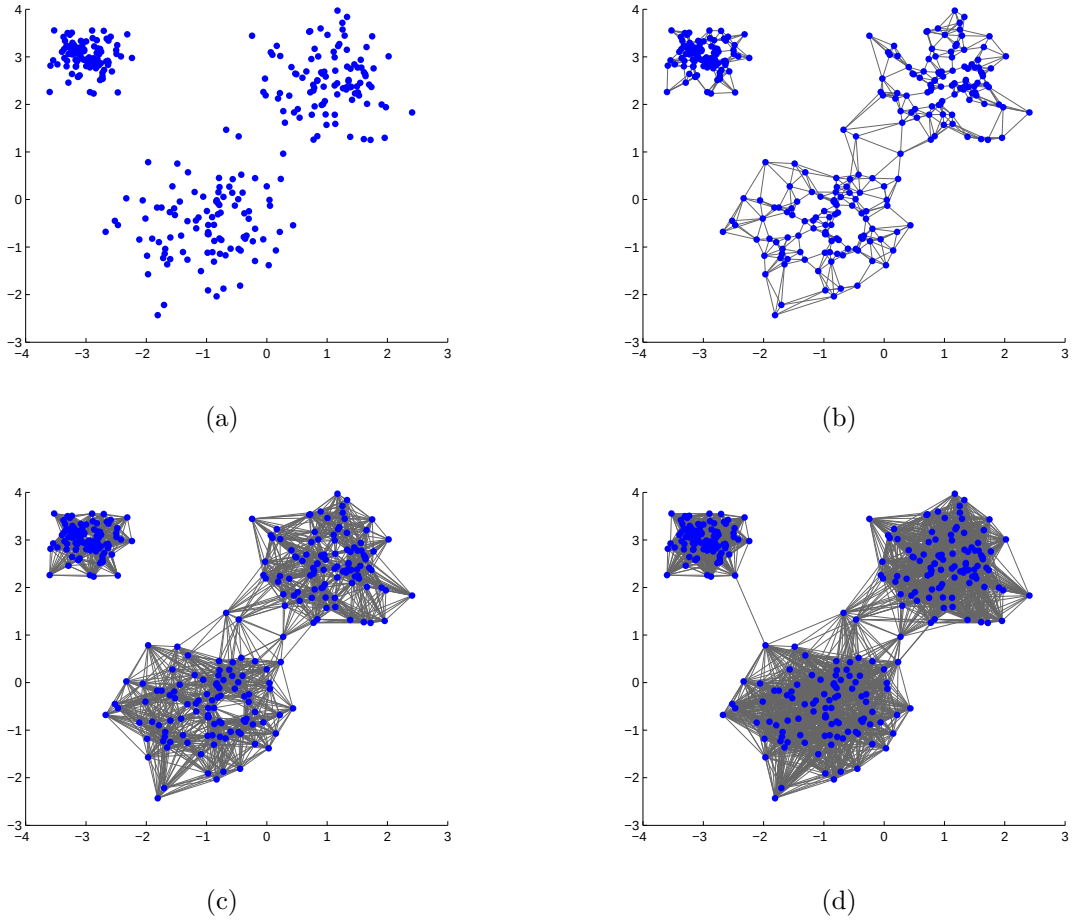



Figura 4.13: Resultado da aplicação do k-vizinhos em um conjunto de dados com 3 grupos. (a) Rede original com 300 vértices, (b) com $k = 5$, (c) com $k = 20$, (d) com $k = 33$.

A seguir propomos uma técnica de construção da rede a partir de um conjunto de dados

baseado na idéia do Single Linkage, a idéia básica é que em cada iteração do algoritmo Single Linkage é criado certo número de ligações entre vértices dos grupos sendo juntados.

Os passos do algoritmo para a construção da rede são:

1. Colocar cada elemento no seu próprio grupo. Construir a matriz de distâncias entre os elementos do conjunto de dados e construir a matriz de adjacência da rede com cada elemento sendo um vértice e nenhuma ligação entre os vértices.

2. Juntar os dois grupos com a menor distância entre eles, denotá-los por grupo $C_1$ e grupo $C_2$.

3. Calcular a média das distâncias entre cada par de vértices já conectados para cada grupo $C_1$ e $C_2$, denotar eles por $dm_1$ e $dm_2$ respectivamente.

4. Criar até $K$ novas arestas entre vértices $i$ e $j$ com as menores distâncias entre eles, onde $i \in C_1$ e $j \in C_2$, ou seja, conectar os dois grupos criando até $K$ arestas entre os vértices mais próximos entres os dois grupos, sempre e quando a distância entre os vértices de cada nova aresta criada seja menor que um valor $d_{lim}$ definido como:

$$d_{lim} = \alpha \max(dm_1, dm_2), \tag{4.4}$$

   onde $\alpha$ é um parâmetro que permite ajustar o limite máximo da distância das novas arestas criadas em função da média das distâncias das arestas já criadas anteriormente para cada grupo, esse limite baseia-se na hipótese de que vértices de um mesmo grupo estão dispostos com uma densidade geralmente uniforme. Se nenhum par de vértices $i, j$ tem menor distância que $d_{lim}$ (provavelmente estão sendo unidos dois grupos originais diferentes), então somente é criada uma aresta entre o par de vértices $i, j$ mais próximos para garantir a conectividade da rede resultante.

5. Recalcular a distâncias entre o novo grupo formado no passo 2 e todos os grupos restantes, onde essa distâncias serão as menores distâncias entre elementos do grupo formado e elementos de cada grupo restante.

6. Se ainda todos os elementos não estão em um grupo só voltar ao passo 2.

A Figura 4.14 mostra as redes geradas pela técnica proposta com $\alpha = 3$ e diferentes valores de $K$ para o conjunto de dados da Figura 4.13(a). A Figura 4.15 mostra as redes geradas pela técnica proposta com $K = 5$ e diferentes valores de $\alpha$ para o mesmo conjunto de dados com 3 grupos e 300 elementos. Note-se que em ambas figuras a rede resultante sempre fica conectada e que cada grupo tem quase mesma densidade de arestas (efeito do $K$), mesmo que a densidade dos grupos no conjunto de dados original não seja semelhante, além disso, o número de arestas que conectam diferentes grupos é baixo comparado com o número de arestas entre vértices do mesmo grupo (efeito do $d_{lim}$), características desejáveis em uma rede com estrutura de comunidade para facilitar a detecção das mesmas.

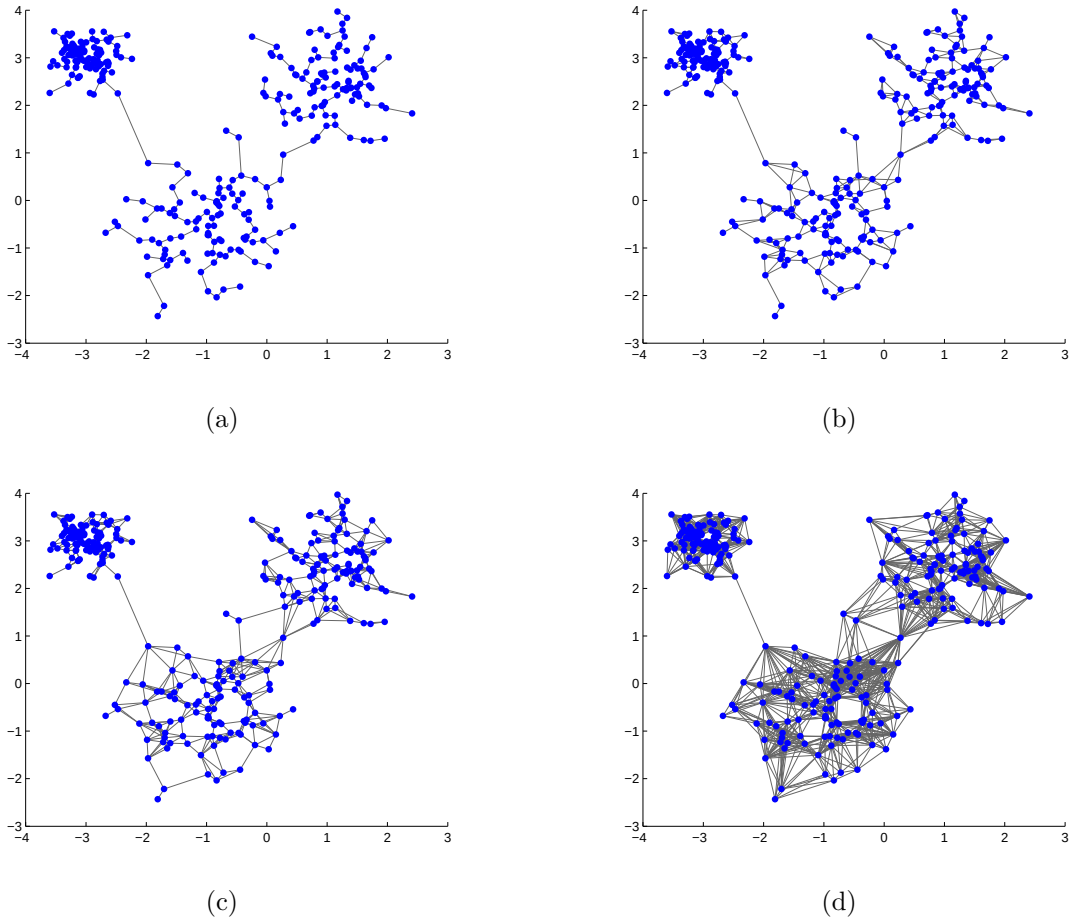



Figura 4.14: Resultado da aplicação da técnica de construção da rede proposta no conjunto de dados da Figura 4.13(a) com $\alpha = 3$. (a) com $K = 1$, (b) $K = 3$, (c) $K = 5$ e (d) $K = 20$.

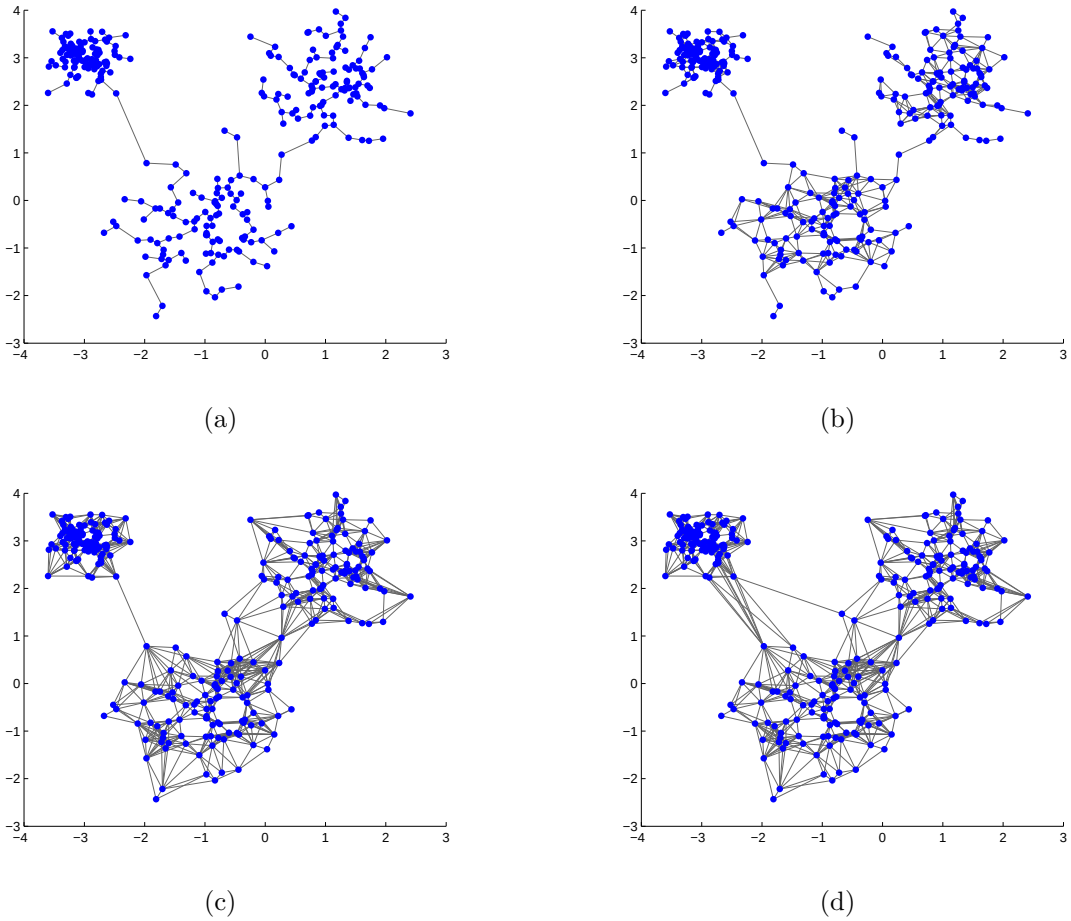



Figura 4.15: Resultado da aplicação da técnica de construção da rede proposta no conjunto de dados da Figura 4.13(a) com $K = 5$. (a) com $\alpha = 1$, (b) $\alpha = 2$, (c) $\alpha = 4$ e (d) $\alpha = 8$.

### 4.2.2 Método de agrupamento

O método de agrupamento de dados proposto consiste em duas etapas:

1. Construir a rede a partir do conjunto de dados original como descrito na Subseção 4.2.1.

2. Sobre a rede obtida aplicar o método de detecção de comunidades descrito na Subseção 4.1.2 utilizando neste caso o Average Linkage sobre as distâncias calculadas segundo o método proposto. Para evitar que grupos que não estejam conectados na rede sejam agrupados em iterações iniciais do Average Linkage é feita uma pequena modificação da matriz de distâncias simétricas $D_s = \{d_{ij}\}$. Essa modificação consiste em atribuir um valor suficientemente grande de distância para cada par de vértices da rede que não estejam conectados. Assim cada elemento $d'_{ij}$ da matriz modificada de distâncias $D'_s$ é definido como:

$$d'_{ij} = \begin{cases} d_{ij} & \text{se existe uma aresta entre os vertices i e j;} \\ \omega & \text{caso contrario,} \end{cases} \tag{4.5}$$

   onde $\omega \gg \max(D_s)$ é um valor bem maior do que a máxima distância da matriz $D_s$ original.

Finalmente podemos escolher a melhor partição do dendograma gerado pela detecção de comunidades, isto pode ser feito com algum índice de validação de agrupamento, por exemplo o índice da Silhueta descrito na Subseção 3.5.3.

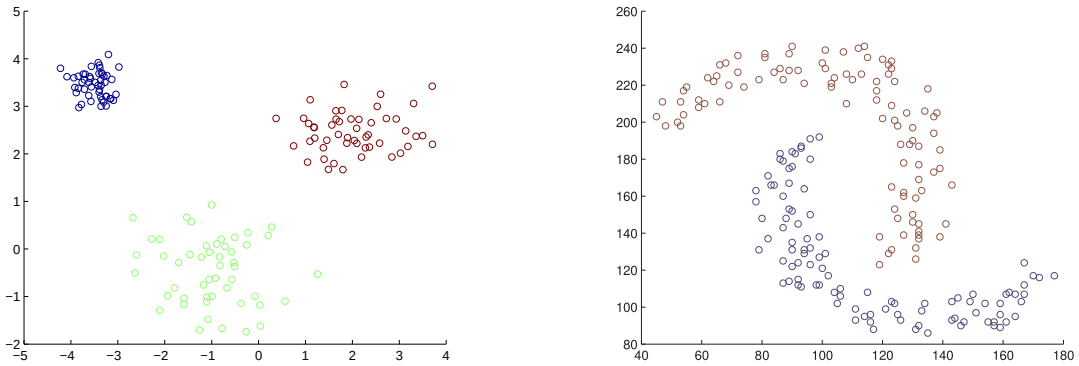
### 4.2.3 Resultados obtidos em redes artificiais

Para avaliar o comportamento do método de agrupamento de dados proposto foram utilizados alguns conjuntos de dados com estrutura de grupos conhecida, os conjuntos de dados são bidimensionais para facilitar a visualização dos grupos.

Para todos os experimentos a rede foi construída como descrito na Subseção 4.2.1 com $K = 5$ e $\alpha = 3$. O agrupamento de dados sobre a rede construída foi feita como descrito na Subseção 4.2.2 com $\omega = 10^3$, para o calculo das distâncias entre os vértices segundo a medida baseada no tempo de consenso foi utilizado $\bar{x} = 0$, $\beta = 1$, $\epsilon = 0.015$, $z = 0.15$.

O primeiro conjunto de dados é formado por 300 elementos e possui 3 grupos com uma distribuição gaussiana bem definidos, os grupos desse conjunto de dados tem diferentes densidades e tamanhos. A Figura 4.16(a) mostra o resultado do método proposto identificando os 3 grupos. O segundo conjunto de dados tem 200 elementos formando dois grupos com formas não globulares, a Figura 4.16(b) mostra o resultado obtido com o método proposto. Em ambos conjuntos de dados o valor do índice de validação Rand corrigido pelo algoritmo proposto foi de 1, quer dizer, a partição é bem definida.

O último conjunto de dados possui 500 elementos várias estruturas de grupos hierárquicos, além disso, alguns elementos entre alguns grupos ficam bem próximos, o método proposto




(a) (b)

Figura 4.16: Resultado da aplicação da técnica de agrupamento de dados proposta em dois conjuntos de dados artificiais, (a) conjunto com 300 elementos e 3 grupos de tamanhos e densidades diferentes, (b) conjunto com 200 e 2 grupos de formas não globulares.

consegue identificar a estrutura hierárquica dos grupos do conjunto de dados como mostrado na Figura 4.17.

### 4.2.4 Resultados obtidos em redes reais

O método proposto de agrupamento de dados foi aplicado em dois conjuntos de dados reais, com os grupos originais conhecidos a priori, e onde os atributos de cada conjunto de dados foram primeiramente normalizados. Nos dois conjuntos de dados, foram utilizados os mesmos parâmetros dos experimentos feitos com os conjuntos de dados artificiais da Subseção 4.2.3. Os resultados forma comparados com os resultados obtidos dos algoritmos clássicos de agrupamentos de dados: Single Linkage (**SL**), Complete Linkage (**CL**), Average Linkage (**AL**) e $K$-médias (**KM**). Para avaliar a eficácia de cada algoritmo em cada conjunto de dados foi utilizado o índice de validação externa Rand Corrigido (**CR**), para os algoritmos que geram uma hierarquia de partições foi avaliada cada partição e foi escolhida aquela com o melhor valor de **CR**.

O primeiro conjunto de dados reais é chamado *iris* introducido por Fisher (1936), este conjunto de dados é regularmente utilizado para testar algoritmos de aprendizado de máquina. Esse conjunto de dados tem 150 elementos divididos em 3 grupos de 50 elementos cada, que representam três espécies da planta íris: Íris Setosa, Íris Verginica e Íris Versicolor. Os 4 atributos do conjunto de dados representam o comprimento e largura das pétalas e sépalos.

Os resultados obtidos pelo método proposto e os resultados dos algoritmos clássicos são mostrados na Tabela 4.1. O número de grupos encontrados por cada algoritmo é definido pela melhor partição obtida (maior valor de $CR$) em cada algoritmo testado. O algoritmo proposto obtém um valor máximo de $CR = 0.8857$ para a melhor partição, valor superior aos obtidos pelos algoritmos tradicionais, além disso, a melhor partição é formada por 3 grupos, definindo corretamente o número de grupos do conjunto de dados.

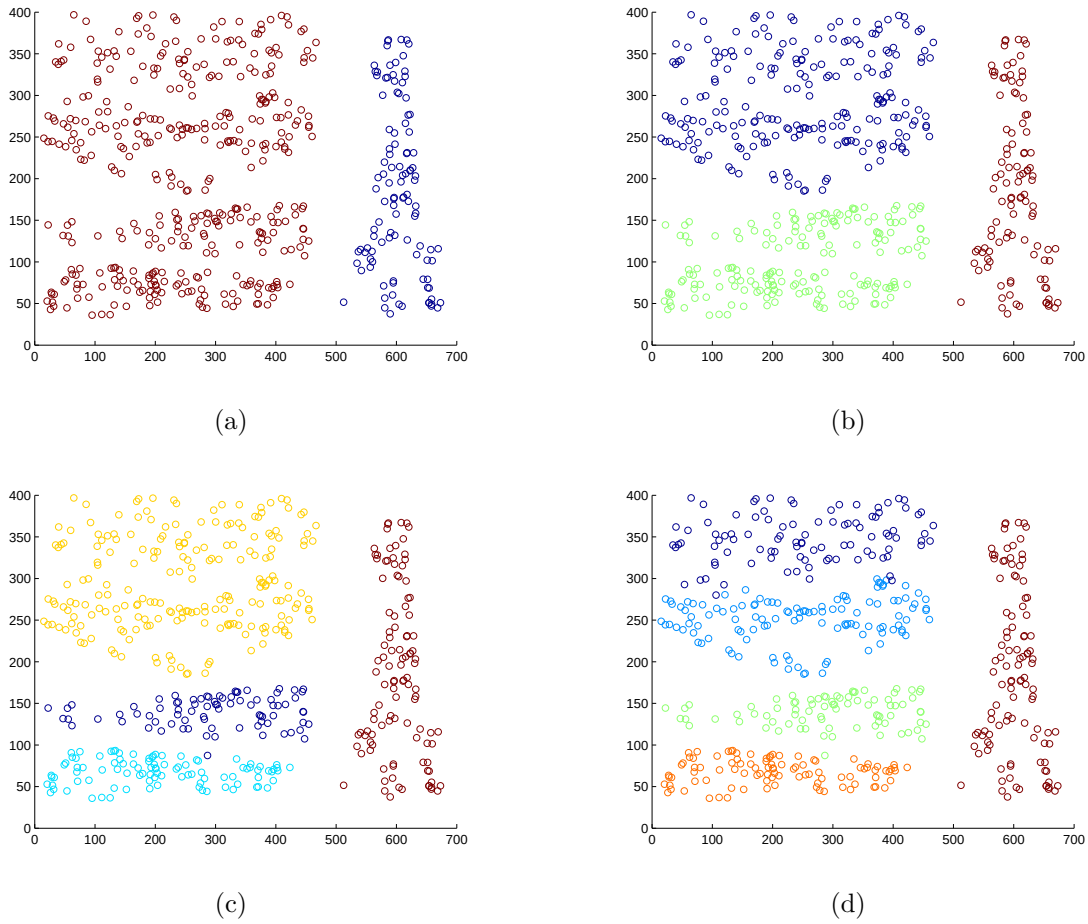



Figura 4.17: Resultado da aplicação da técnica de agrupamento de dados proposta em um conjunto de dados com 500 elementos e estrutura hierárquica de grupos. Do dendograma gerado foram obtidas partições com: (a) 2 grupos, (b) 3 grupos, (c) 4 grupos e (d) 5 grupos.

Tabela 4.1: Resultados obtidos para o conjunto de dados *Íris*. O conjunto de dados possui 3 grupos reais.

| **Algoritmo** | **CR** | **Grupos** |
|---|---|---|
| **SL** | 0.5681 | 2 |
| **CL** | 0.7060 | 3 |
| **AL** | 0.7196 | 3 |
| **KM** | 0.7163 | 3 |
| **Proposto** | 0.8857 | 3 |

Tabela 4.2: Resultados obtidos para a primeira divisão do conjunto de dados *golub* em 2 grupos reais.

| **Algoritmo** | **CR** | **Grupos** |
|---|---|---|
| **SL** | 0.1018 | 9 |
| **CL** | 0.7844 | 2 |
| **AL** | 0.8760 | 6 |
| **KM** | 0.4184 | 3 |
| **Proposto** | 0.7963 | 2 |

Tabela 4.3: Resultados obtidos para a segunda divisão do conjunto de dados *golub* em 3 grupos reais.

| **Algoritmo** | **CR** | **Grupos** |
|---|---|---|
| **SL** | 0.0124 | 9 |
| **CL** | 0.5337 | 2 |
| **AL** | 0.7884 | 9 |
| **KM** | 0.4802 | 3 |
| **Proposto** | 0.7397 | 3 |

O segundo conjunto de dados é chamado de *golub*, que reúne informações de pacientes com leucemia, foi proposto em (Golub et al., 1999) e consiste em um conjunto de dados de expressões de genes de 72 pacientes com 3571 atributos. No conjunto de dados existe duas divisões, a primeira divisão em dois grupos refere-se a dois tipos de leucemia, um grupo de 47 pacientes com $ALL$ (Acute Lymphoblastic Leukemia), e outro grupo de 25 pacientes com $AML$ (Acute Myeloid Leukemia). Uma segunda divisão em três grupos correspondentes ao grupo $AML$ e à divisão do grupo $ALL$ em dois subgrupos: $T$-$ALL$ e $B$-$ALL$, $p$ pacientes com $T$-$ALL$ e 38 pacientes com $B$-$ALL$.

Os resultados obtidos com o algoritmo proposto e os algoritmos clássicos para a primeira divisão em 2 grupos reais são mostrados na Tabela 4.2, os resultados para a segunda divisão em três grupos são mostrados na Tabela 4.3. O algoritmo proposto alcança um valor de $CR = 0.7963$ com 2 grupos encontrados para a primeira divisão e um valor de $CR = 0.7397$ com 3 grupos encontrados para a segunda divisão, embora o algoritmo Average Linkage alcance maior valor de $CR$ nas duas divisões, o número de grupos estimados por esse algoritmo (9 em cada divisão), não coincide com o numero de grupos reais do conjunto de dados, no entanto, o método proposto de agrupamento de dados consegue um bom valor de $CR$ e divide corretamente o conjunto de dados na quantidade real de grupos.

## 4.3 Simplicidade em redes complexas

Redes complexas por definição são redes que não apresentam uma estrutura trivial de conexões, porém em alguns casos as redes podem apresentar regiões ou subredes com estrutura de conexões simples. Uma rede regular é entendida como uma rede simples em

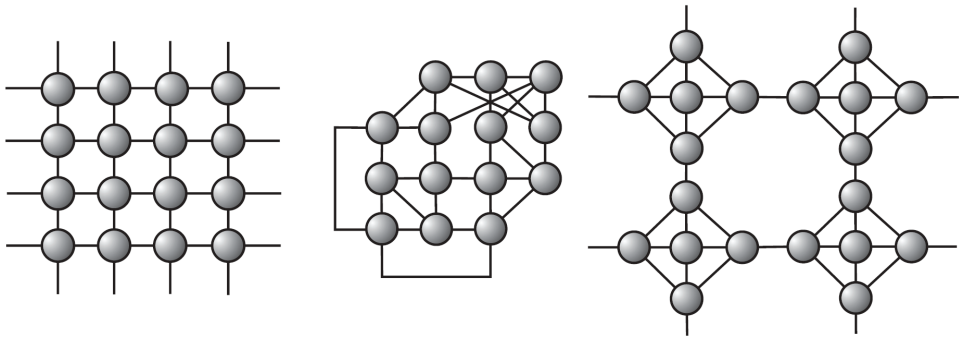


Figura 4.18: Três redes regulares com todos os vértices com grau 4 e configurações diferentes. Figura obtida de (da F. Costa & Rodrigues, 2009).

termos de estrutura de conexões entre vértices, pois cada vértice possui o mesmo grau, mas mesmo em redes regulares com o mesmo grau pode existir diferentes configurações, umas mais simples que outras, isto pode ser observado na Figura 4.18.

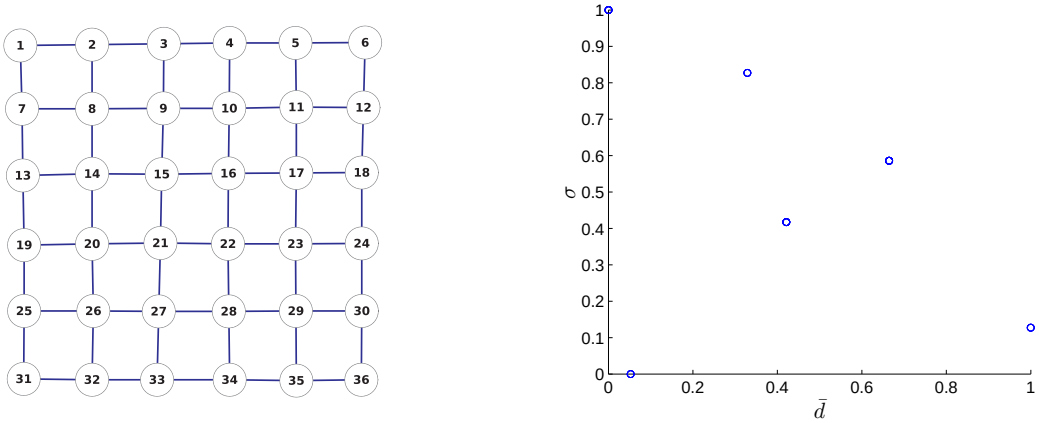
Uma tarefa interessante em redes complexas é caracterizar a simplicidade da rede. Em (da F. Costa & Rodrigues, 2009) os autores propuseram um método de detecção de regiões simples em redes complexas. Esse método é baseado na extração de $m$ medidas da rede, tais como: a distribuição de grau, coeficiente de agrupamento, entre outras. Com essas medidas próprias de cada vértice são representados pontos no espaço $m$-dimensional, cada ponto representa um vértice. O espaço é projetado em um espaço bidimensional utilizando a análise de componentes principais (PCA), para depois detectar regiões de pontos com alta densidade, essas regiões representam regiões simples na rede. Essa abordagem tem certa dificuldade na escolha das medidas da rede, pois a escolha de algumas medidas locais (a grande maioria de medidas estudadas em redes complexas) podem não caracterizar bem a simplicidade da rede.

Nesta seção é apresentada uma nova técnica para encontrar as regiões mais simples da rede, técnica baseada na medida de distâncias descrita na Subseção 4.1.1.

### 4.3.1 Método de detecção de regiões simples em redes complexas

A medida de distância proposta, indica o número de iterações que um vértice $j$ levou para atingir o consenso para um estado desejado com ajuda de um vértice líder $i$. Assim se olharmos as distâncias obtidas para todos os vértices $j$ que são vizinhos de $i$, podemos ver o grau de homogeneidade da vizinhança de $i$. Se todos os vizinhos de $i$ atingiram o consenso em tempos similares, faz sentido pensar que as redondezas de $i$ têm certa homogeneidade, e assim, é possível que o vértice $i$ seja parte de uma região simples ou homogênea. Nesse sentido foi proposto um método para encontrar as regiões mais simples, em termos de estrutura de conexões, baseado na medida de distância proposta.

Considere a matriz de distância assimétricas $D$, de uma rede com $n$ vértices, calculadas como descrito na Subseção 4.1.1 com elementos $d_{ij}$, cada vértice da rede irá representar um ponto $p_i$ em um espaço bidimensional com coordenadas $p_i = (\bar{d}_i, \sigma_i)$, onde $\bar{d}_i$ e $\sigma_i$ são




(a) (b)

Figura 4.19: Detecção de regiões simples. (a) Rede com 36 vértices e (b) Pontos obtidos no espaço bidimensional.

a média e o desvio padrão das distâncias $d_{i,k}$, sendo $k$ o índice de um vértice vizinho de $i$. Assim essas coordenadas são normalizadas no intervalo $[0, 1]$ formando a matriz $P_{n\times 2}$ que representa esses pontos, se plotamos esses pontos no espaço bidimensional, os vértices com desvio padrão e média similares formarão grupos no espaço, esses grupos podem representar regiões simples da rede se os vértices desse grupo estiverem conectados. A Figura 4.19 mostra uma rede simples e os pontos obtidos no espaço bidimensional.

Para obter as regiões mais simples da rede é aplicado um algoritmo de agrupamento de dados, neste caso é escolhido o Single Linkage, pois permite criar uma estrutura hierárquica de regiões simples e permite juntar duas regiões simples sempre e quando elas sejam conectadas. Assim, para poder aplicar o Single Linkage é preciso calcular a matriz de distâncias euclidianas entre vizinhos $E$ dos pontos na matriz $P$, $E$ com elementos $e_{ij}$ definidos como:

$$e_{ij} = \begin{cases} \sqrt{(\bar{d}_i - \bar{d}_j)^2 + (\sigma_i - \sigma_j)^2} + \bar{\sigma} & \text{se há uma aresta entre os vertices i e j;} \\ \infty & \text{caso contrário,} \end{cases} , \quad (4.6)$$

onde $\bar{\sigma} = (\sigma_i + \sigma_j)/2$ é a média dos desvios padrões de $i$ e $j$ e serve para dar maior importância ao desvio padrão no cálculo da distância. Se $i$ e $j$ não estiverem conectados na rede, então $e_{ij} = \infty$, isto para assegurar que o algoritmo Single Linkage não agrupe dois conjuntos de vértices que não estejam conectados, pois para um grupo ser uma região simples tem de ser conectado.

Assim para obter as regiões mais simples da rede basta aplicar o algoritmo de agrupamento Single Linkage sobre a matriz de distâncias entre vizinhos $E$, o resultado será um conjunto de partições hierárquicas que podem ser representadas em um dendograma, os grupos que forem unidos em iterações iniciais do Single Linkage serão as regiões mais simples da rede. Assim se olhamos o dendograma de baixo para acima, as regiões ou grupos

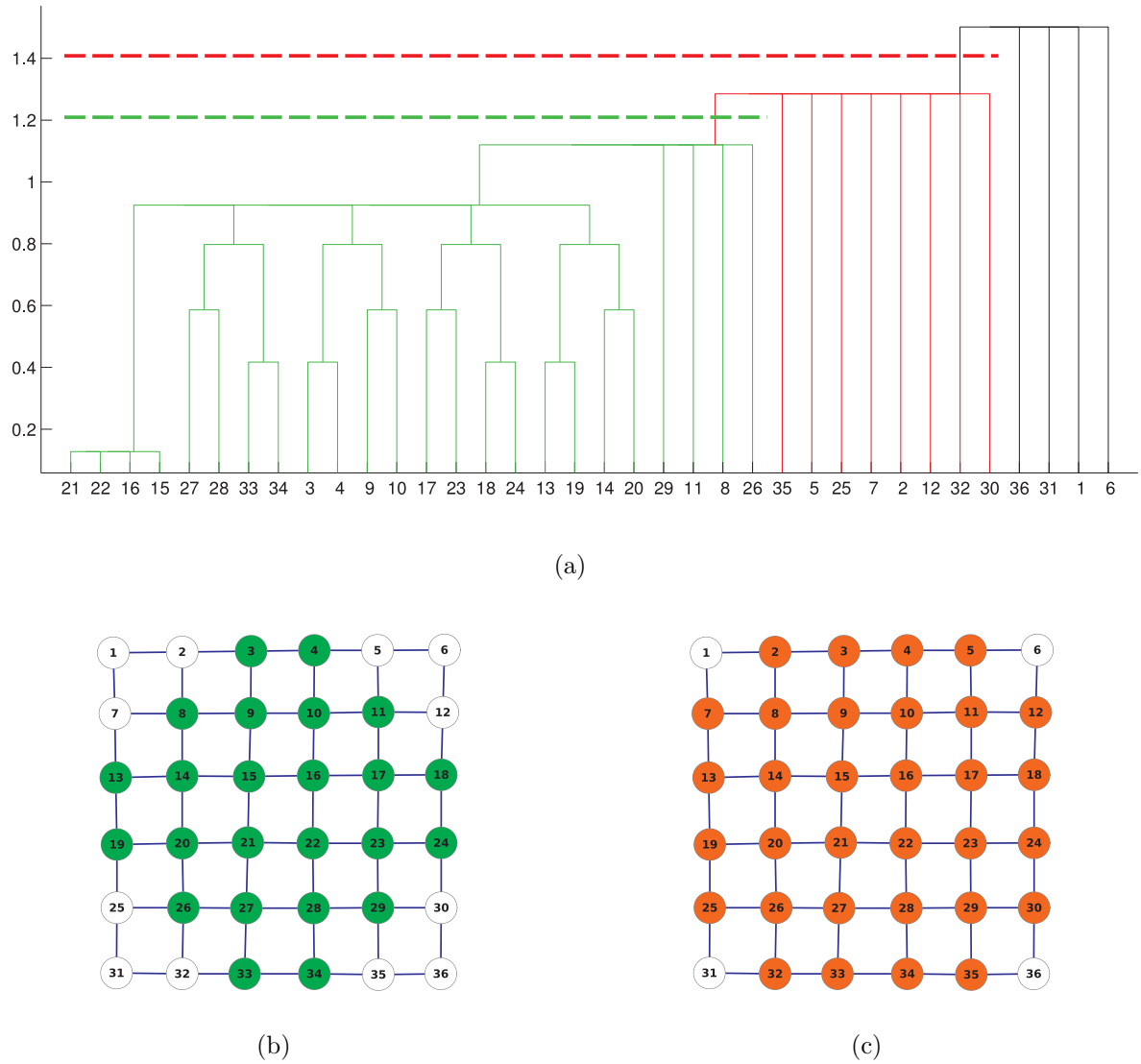



Figura 4.20: Regiões simples encontradas na rede da Figura 4.19(a). (a) Dendograma obtido, dois cortes (linhas pontuadas) foram feitos no dendrograma. (b) Região simples encontrada para o primeiro corte no dendograma (linha pontuada verde). (c) Região simples encontrada para o segundo corte no dendograma (linha pontuada vermelha).

de vértices estarão classificados hierarquicamente em ordem decrescente de simplicidade. A Figura 4.20(a) mostra o dendograma obtido para a rede simples da Figura 4.19(a), e a Figura 4.20(b) mostra a região simples obtida para o primeiro corte no dendograma e a Figura4.20(c) mostra a região simples encontrada para o segundo corte no dendograma. Nota-se que os vértices nas proximidades das esquinas são os vértices mais diferenciados dos demais vértices da rede.

### 4.3.2 Simulações

Com intenção de avaliar o comportamento do método de detecção de regiões simples em redes complexas proposto, foram utilizados duas redes com regiões homogêneas ou simples.

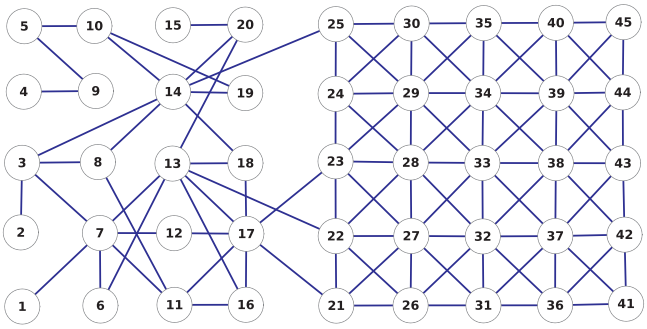



Figura 4.21: Rede com uma região quase regular e outra quase aleatória.

Para todos os experimentos, primeiramente foram calculados as distâncias entre os vértices segundo a medida baseada no tempo de consenso, descrita na Subseção 4.1.1 com os mesmos parâmetros utilizados nos experimentos de detecção de comunidades e agrupamento de dados: $\bar{x} = 0$, $\beta = 1$, $\epsilon = 0.015$ e $z = 0.15$.

A primeira rede testada tem 45 vértices e está dividida em duas partes, uma parte com estrutura quase regular e outra parte com uma estrutura quase aleatória, esta rede é mostrada na Figura 4.21. Aplicado o método de detecção de regiões simples proposto, obtemos um dendograma, a Figura 4.22, mostra as regiões simples encontras em distintos cortes do dendrograma.

No primeiro corte, na Figura 4.22(a), é encontrada uma região simples que não contem os vértices das esquinas (41 e 45), possivelmente por ter estruturas diferentes dos demais vértices próximos a eles, e também não contem os vértices (22 e 25), isto pode ser, devido ao fato de que existem 4 vértices que ligam a região simples completa e a região quase aleatória (vértices 21, 22, 23 e 25), vértices que se diferenciam do resto, mas note-se que os vértices (21 e 23) estão considerados na região simples obtida pelo algoritmo, acredita-se que esses dois vértices foram considerados porque eles estão conectados com o vértice 17 que de alguma maneira segue o padrão de conexões da região simples obtida, inclusive as arestas entre os vértices 21 e 23 com o vértice 17, seguem o padrão de conexões da região simples, não sendo o mesmo caso com os vértices 22 e 25 que não foram considerados na região simples obtida. Isto pode ser melhor observado na Figura 4.23, onde foram retiradas duas arestas entre os vértices (13,22) e (13,26) para melhor visualização.

No terceiro corte do dendrograma foram encontradas duas regiões simples mostradas na Figura 4.22(c), uma bem simples e grande (região verde) e a outra menos simples e menor (região cinza) aparentemente quase aleatória, mas olhando somente as arestas dos vértices da região cinza, é revelada certa estrutura que segue também o padrão da região simples verde como mostrado na Figura 4.24.

A segunda rede tem 44 vértices, esta rede foi criada a partir de uma rede quase regular de 36 vértices, na qual foram adicionados 6 vértices com poucas conexões aleatórias a alguns dos 36 vértices iniciais. A idéia desta rede é representar uma rede quase regular com ruído, a Figura 4.25 mostra esta rede com ruído.

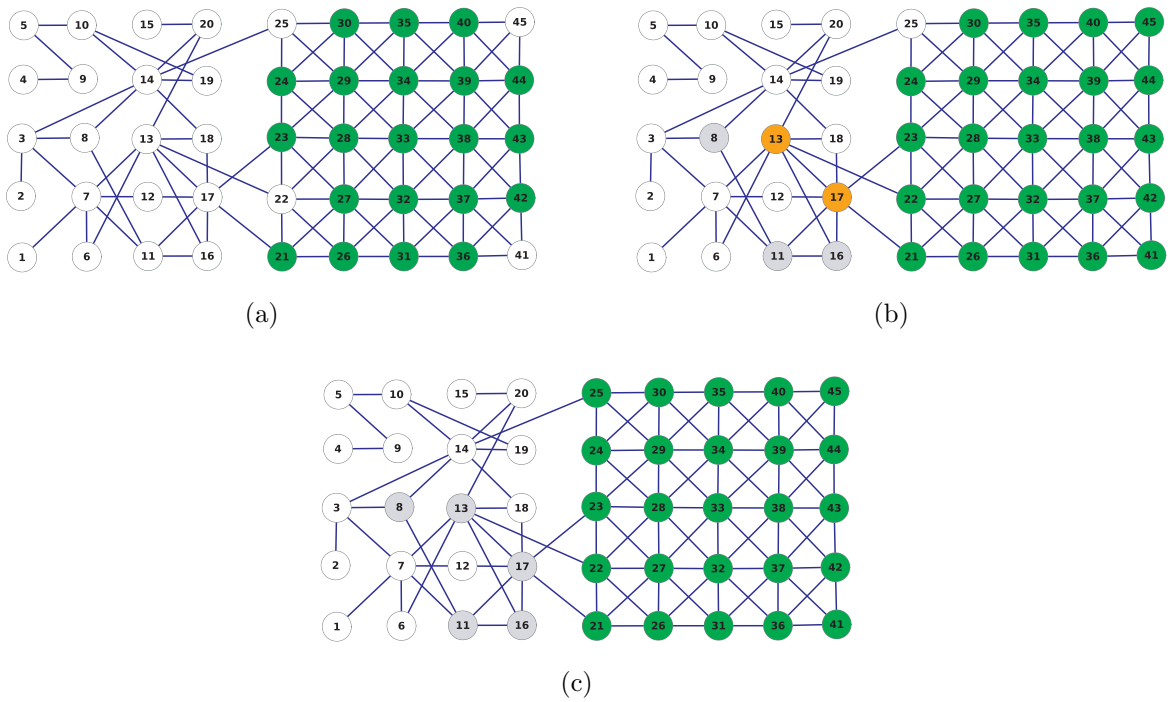



Figura 4.22: Regiões simples encontradas na rede da Figura 4.21 para 3 cortes diferentes no dendograma. Cada região é representada por uma cor diferente. (a) primeiro corte com uma região simples, (b) segundo corte com três regiões simples e (c) terceiro corte com duas regiões simples.

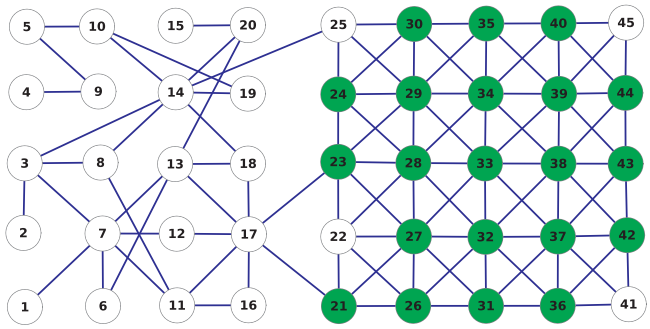


Figura 4.23: Primeira região simples obtida da rede da Figura 4.21, na qual duas arestas entre os vértices (13,22) e (13,26) foram removidas para melhor visualização do padrão de conexões na redondeza do vértice 17.

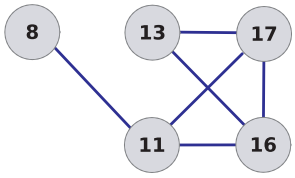



Figura 4.24: Segunda região simples obtida da rede da Figura 4.21 para o terceiro corte no dendrograma (Figura 4.22(c)), onde somente foram consideradas as arestas que ligam vértices dessa região.

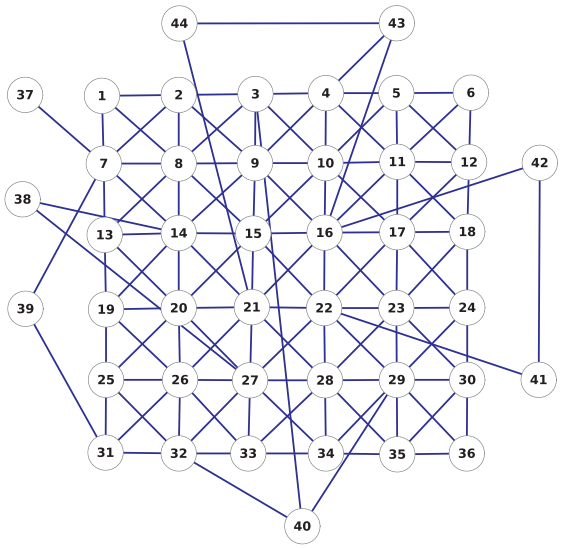



Figura 4.25: Rede quase regular com alguns vértices adicionados como ruído.

Foi aplicado a técnica proposta de detecção de regiões simples com o propósito de tentar encontrar somente os vértices que formam a região quase regular, descartando os vértices que representam ruído. A Figura 4.26 mostra o resultado obtido, quatro cortes foram feitos no dendograma, no último corte ficou claramente identificada a região quase regular.

É interessante observar que o vértice 7 na região do terceiro corte (Figura 4.26(c)) ainda não é considerado dentro da região simples, isto pode ser devido a que esse vértice é o único vértice que tem duas arestas que o ligam com vértices representando ruído (vértices 37 e 39), e que é um vértice da borda da região quase regular obtida. O vértice 16 também possui duas ligações para vértices ruído (vértices 42 e 43) mas ele não pertence à borda da região simples, sendo considerado dentro da região quase regular em iterações inicias da técnica.

## 4.4 Considerações finais

Neste capítulo, foi apresentada uma nova medida de distância para redes complexas baseado no tempo que um vértice leva para atingir um estado desejado no problema de consenso na presença de um líder na rede. Esta medida de distância junto com algoritmos de agrupamento de dados hierárquicos foi aplicada para detecção de comunidades em redes complexas. Também foi proposto uma técnica para a criação de uma rede conectada e esparsa a partir de um conjunto de dados, isto para ser utilizado junto com a técnica de detecção de comunidades proposta e formar um método de agrupamento de dados completo. Além disso, foi proposto um método para detectar regiões simples em redes complexas, baseado na medida de distância do tempo de consenso.

Os métodos de detecção de comunidades, agrupamento de dados e de detecção de regiões simples foram testados com diferentes redes (ou conjuntos de dados no caso do

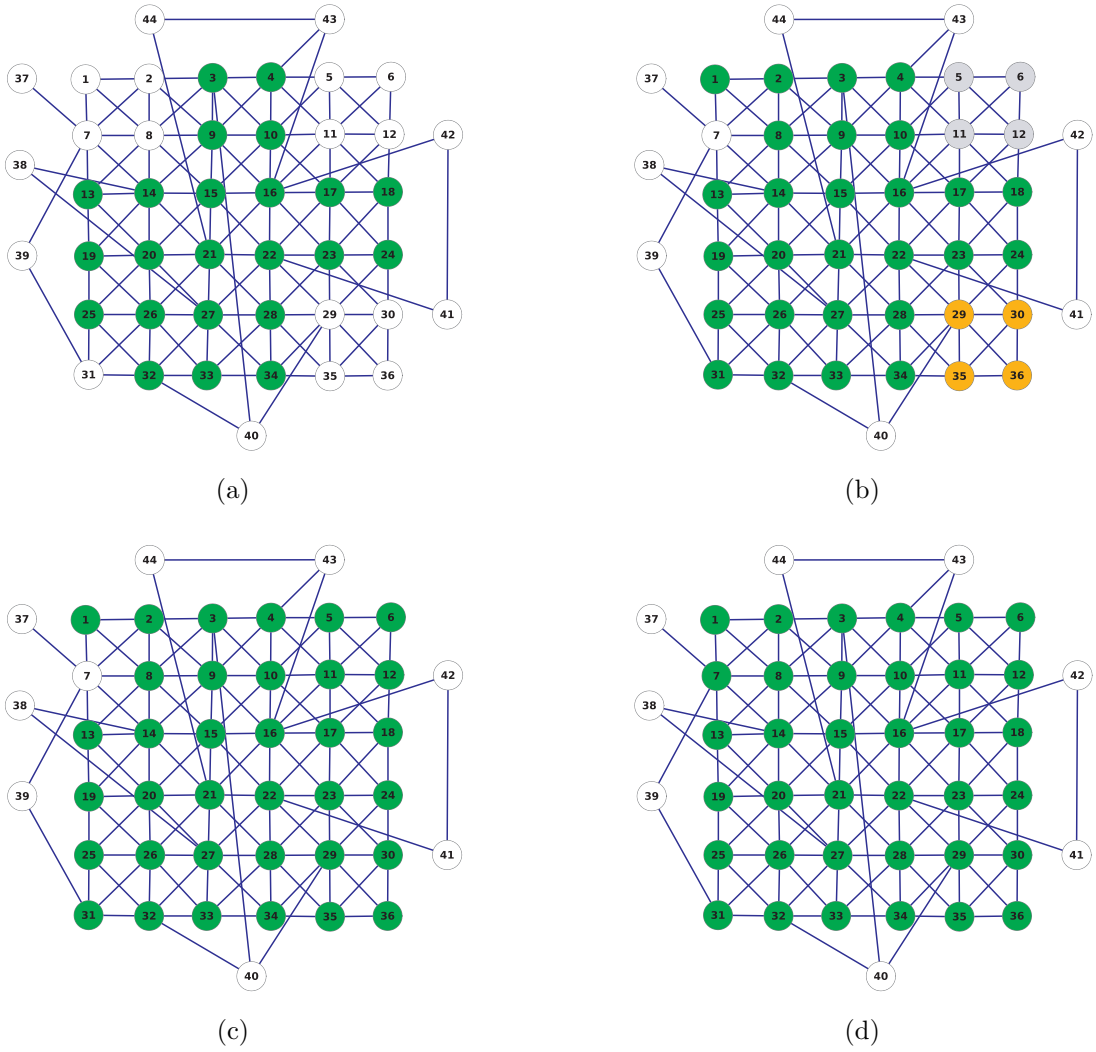



Figura 4.26: Regiões simples encontradas na rede da Figura 4.25 para 4 cortes diferentes no dendograma. Cada região é representada por uma cor diferente. (a) primeiro corte com uma região simples, (b) segundo corte com três regiões simples, (c) terceiro corte com uma região simples e (d) quarto corte com a região simples completamente identificada.

agrupamento de dados), artificiais e reais. No caso da detecção de comunidades também foram apresentados os resultados obtidos com várias redes geradas com estrutura de comunidade, com diferentes graus de separabilidade entre os grupos. A técnica de agrupamento de dados foi comparada com os algoritmos clássicos de agrupamento em conjuntos de dados bastante utilizados na validação de algoritmos de agrupamento, nos resultados obtidos o método proposto mostrou-se superior aos demais. No caso da técnica de detecção de regiões simples na rede, foram apresentados os resultados obtidos com duas redes, uma contendo uma região quase homogênea e uma região quase aleatória, e uma rede quase regular, na qual foram adicionados alguns vértices conectados à região simples da rede para representar ruído na região homogênea.

Os resultados alcançados mostram que a medida de distância proposta tem algumas características interessantes que permitem obter bons resultados na detecção de comunidades ou agrupamento de dados e na detecção de regiões simples em redes complexas. Acredita-se que existam outras aplicações onde pode-se obter resultados também relevantes. O próximo capítulo apresentará a relevância dos resultados obtidos, conclusões e trabalhos futuros.

Capítulo

5

# Conclusão

O presente trabalho trata-se de análise de dados utilizando redes complexas, especificamente o agrupamento de dados e a detecção de regiões simples em redes. Para isso, foi apresentado uma revisão sobre redes complexas no Capítulo 2. Nesse capítulo, foram apresentados os principais modelos de redes, medidas de interesse na rede, métodos representativos de detecção de comunidades e dois conceitos importantes em sistemas dinâmicos de redes acopladas: consenso e o controle focalizado (*pinning control*), os quais serviram para a definição da medida de distância em redes complexas proposta. Além disso, foi apresentado uma revisão do tema de agrupamento de dados, métodos particionais, hierárquicos e métodos baseados em grafos (Capítulo 3).

No Capítulo 4, foi apresentada a nova medida de distância para redes complexas baseada no tempo que um vértice leva para atingir o consenso na presença de um líder de grupo. A medida de distância, juntamente com os algoritmos hierárquicos de agrupamento de dados, foi utilizada para a detecção de comunidades em redes complexas. Os resultados das simulações aplicadas em redes artificiais e reais indicam que a medida de distância em redes complexas proposta é promissora para detecção de comunidades. Acreditamos que isto seja devido ao fato de que vértices densamente conectados entre eles (formando uma comunidade) movem-se quase na mesma velocidade. Portanto, desde o ponto de vista de um vértice líder, vértices na mesma comunidade serão guiados para o consenso mais rapidamente que vértices fora da própria comunidade, devido ao denso acoplamento em cada comunidade.

No mesmo capítulo, foi apresentado um método de agrupamento de dados baseado na técnica de detecção de comunidades proposta. O método consiste em duas etapas: construção de uma rede esparsa a partir do conjunto de dados e a detecção de comunidades nessa rede. Para a primeira etapa, foi proposta uma técnica baseada na idéia do *Single Linkage*, que forma algumas conexões entre dois grupos unidos pelo algoritmo

em cada iteração, assim a rede formada é esparsa e conectada. Além disso, grupos de dados com distintas densidades no conjunto original de entrada, ficam com semelhante densidade de conexões na rede produzida, característica importante para que o algoritmo de detecção de comunidades identifique corretamente as comunidades. O método de agrupamento proposto foi testado com conjuntos de dados artificiais com grupos de diversas densidades, tamanhos e formas para avaliar o comportamento deste, também foram feitas comparações com algoritmos de agrupamento de dados clássicos em conjuntos de dados reais, conjuntos bastante utilizados para avaliar algoritmos de agrupamento de dados, obtendo bons resultados em relação aos algoritmos tradicionais.

Também foi desenvolvido um método para encontrar as regiões mais simples de uma rede baseado na medida de distância proposta. O método retorna uma hierarquia de regiões simples que pode ser representado em um dendograma. O método proposto foi testado com algumas redes artificiais com regiões simples, e uma rede que representa uma região quase regular com alguns vértices atuando como ruído, o objetivo era descartar os vértices que representavam ruído e encontrar a região regular. Foram obtidos resultados muito interessantes, pois acredita-se que o algoritmo é capaz de encontrar regiões de vértices com algum padrão de conexões, tolerando alguns vértices e conexões, tarefa que seria muito difícil de ser realizada por uma pessoa, mesmo em redes pequenas. Acreditamos que os bons resultados obtidos neste método devam-se à natureza global da medida de distância proposta, pois pelo fato de ser calculada em um sistema dinâmico acoplado (no qual todos os elementos interagem entre eles), a medida entre dois vértices leva também informações da topologia de toda a rede, porque a velocidade com que um vértice acoplado atinge o consenso é afetada em maior grau pela velocidade dos seus vizinhos, e os seus vizinhos são afetados por seus vizinhos respectivos e assim por diante. É importante ressaltar que no tópico de detecção de regiões simples em redes complexas existem ainda muito poucos trabalhos, e tem muito espaço para o desenvolvimento de novos métodos, medidas e índices para medir a simplicidade de uma rede.

Assim podemos concluir que redes complexas têm a vantagem de revelar informações importantes para a análise de dados, e que o estudo de sistemas de agentes acoplados pode contribuir muito no desenvolvimento de novas técnicas de análise de dados em geral.

## 5.1 Contribuições

Entre as principais contribuições que foram propostas no presente trabalho destacam-se:

- Uma medida de distância para redes complexas baseada no tempo de consenso da rede.
- Um método de detecção de comunidades em redes complexas baseado na medida de distância.

- Uma técnica de construção de redes esparsas e conectadas a partir de um conjunto de dados.

- Um método de agrupamento de dados hierárquico baseado no método de detecção de comunidades e a técnica de construção da rede, capaz de detectar grupos com formas, tamanhos e densidades arbitrários.

- Uma técnica para encontrar as regiões mais simples de uma rede baseada na medida de distância de tempo de consenso em redes complexas.

Os resultados obtidos no presente trabalho, resultaram na elaboração de um artigo aceito no *International Joint Conference on Neural Networks (IJCNN)*. Além disso, durante o estudo do tópico de redes complexas, foi elaborado em conjunto, um artigo sobre detecção de *outliers* em redes complexas (Berton et al., 2010), trabalho muito ligado com os temas aqui tratados, o qual foi aceito no *IEEE World Congress on Computational Intelligence (WCCI)*.

## 5.2 Trabalhos futuros

No presente trabalho foi tratado a detecção de comunidades, agrupamento de dados e análise de simplicidade em redes complexas, tarefas baseadas na medida de distância proposta. No caso da detecção de comunidades foi utilizado um algoritmo de agrupamento de dados hierárquico (os quais precisam de uma matriz de distâncias simétrica como entrada) para encontrar as comunidades a partir das distâncias, podem ser utilizados e testados outros algoritmos de agrupamento de dados ou pode-se desenvolver um algoritmo de detecção de comunidades que leve em conta a natureza assimétrica da distância proposta.

Uma parte muito importante dos métodos de agrupamento de dados baseados em grafos é a construção da rede a partir do conjunto de dados. A técnica de construção de redes proposta pode ser testada com outros algoritmos de agrupamento de dados e assim avaliar o novo comportamento de cada algoritmo.

No método de detecção de regiões simples proposto pode ser definido um índice de simplicidade para cada região encontrada e para a rede, esse índice permitiria comparar regiões simples e mais interessante ainda é comparar a simplicidade de duas redes ou a evolução de uma rede dinâmica, reconhecer se ela vai se tornando mais simples ou mais complexa com o tempo. Pode ser também proposto um *benchmark* para avaliar métodos de detecção de simplicidade em redes complexa, o qual pode consistir em ir adicionando vértices e conexões a uma rede regular (aumentar gradualmente o número de vértices adicionados), os vértices adicionados irão representar ruído na rede regular que o algoritmo sendo testado deve ser capaz de tolerar e encontrar os vértices que formam parte da região regular.

Além disso, acredita-se que a medida de distância baseada no tempo de consenso pode ser utilizada para detecção de vértices singulares (*outliers*), pois pela forma de calcular a medida de distância, um vértice é selecionado como líder para guiar o consenso de toda a rede em cada iteração, faz sentido pensar que vértices singulares guiem toda a rede para o consenso de uma maneira diferente da maioria dos vértices na rede.

# Bibliografia


Albert, R., Albert, I., & Nakarado, G. L. (2004). Structural vulnerability of the north american power grid. *Phys. Rev. E*, 69(2), 025103.

Albert, R., Jeong, H., & Barabasi, A.-L. (1999). The diameter of the world wide web. *Nature*, 401, 130–131.

Berton, L., Huertas, J., Araújo, B., & Zhao, L. (2010). Identifying abnormal nodes in complex networks by using random walk measure. *2010 IEEE Congress on Evolutionary Computation (CEC)*, (pp. 1 –6).

Chen, F., Chen, Z., Xiang, L., Liu, Z., & Yuan, Z. (2009). Reaching a consensus via pinning control. *Automatica*, 45(5), 1215 – 1220.

Chen, T., Liu, X., & Lu, W. (2007). Pinning complex networks by a single controller. *Circuits and Systems I: Regular Papers, IEEE Transactions on*, 54(6), 1317 –1326.

Clauset, A., Newman, M. E. J., & Moore, C. (2004). Finding community structure in very large networks. *Physical Review E*, 70(6), 066111+.

da F. Costa, L. & Rodrigues, F. A. (2009). Seeking for simplicity in complex networks. *EPL (Europhysics Letters)*, 85(4), 48001.

Daxin Jiang, Chun Tang, A. Z. (2004). Cluster analysis for gene expression data: a survey. *IEEE Transactions on Knowledge and Data Engineering*, 16(11), 1370–1386.

de Oliveira, T., Zhao, L., Faceli, K., & de Carvalho, A. (2008). Data clustering based on complex network community detection. In *Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on* (pp. 2121–2126).

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern Classification*. Wiley, John and Sons, Incorporated.

Erdös, P. & Rényi, A. (1959). On random graphs, i. *Publicationes Mathematicae (Debrecen)*, 6, 290–297.

Ester, M., Kriegel, H., S, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. (pp. 226–231).

Faloutsos M., Faloutsos, P. F. (1999). On power-law relationship of the internet topology. *ACM SIGCOMM 99*, (29), 251–260.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7, 179–188.

Flake, G. W., Lawrence, S., Giles, C. L., & Coetzee, F. M. (2002). Self-organization and identification of web communities. *Computer*, 35(3), 66–70.

Fortunato, S. & Barthlemy, M. (2007). Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1), 36–41.

G. Karypis, V. K. (1998). Hmetis 1.5: A hypergraph partitioning package.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., Bloomfield, C. D., & Lander, E. S. (1999). Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. *Science*, 286(5439), 531–537.

Gu, L., Zhang, X.-D., & Zhou, Q. (2010). Consensus and synchronization problems on small-world networks. *Journal of Mathematical Physics*, 51(8), 082701.

Guha, S., Rastogi, R., & Shim, K. (1998). Cure: an efficient clustering algorithm for large databases. In *SIGMOD '98: Proceedings of the 1998 ACM SIGMOD international conference on Management of data* (pp. 73–84). New York, NY, USA: ACM.

Guha, S., Rastogi, R., & Shim, K. (2000). Rock: A robust clustering algorithm for categorical attributes. In *In Proc.ofthe15thInt.Conf.onDataEngineering*.

Jain, A. K. & Dubes, R. C. (1988). *Algorithms for Clustering Data*. Prentice Hall.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: a review. *ACM Comput. Surv.*, 31(3), 264–323.

Karypis, G., Han, E.-H., & Kumar, V. (1999). Chameleon: hierarchical clustering using dynamic modeling. *Computer*, 32(8), 68–75.

Kogan, J., Nicholas, C., & Teboulle, M. (2006). *Grouping Multidimensional Data: Recent Advances in Clustering*. Springer.

Kryszkiewicz, M. & Skonieczny, L. (2005). Faster clustering with dbscan. *Intelligent Information Processing and Web Mining*, (pp. 605–614).

Li, K., Small, M., & Fu, X. (2008). Generation of clusters in complex dynamical networks via pinning control. *Journal of Physics A: Mathematical and Theoretical*, 41(50), 505101.

Li, Z., Duan, Z., Chen, G., & Huang, L. (2010). Consensus of multiagent systems and synchronization of complex networks: A unified viewpoint. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 57(1), 213–224.

Luciano, Rodrigues, F. A., Travieso, G., & Boas, V. P. R. (2006). Characterization of complex networks: A survey of measurements. *Advances in Physics*, 56(1), 167–242.

Lusseau, D. (2003). The emergent properties of a dolphin social network. *Proc Biol Sci*, 270.

Nagy, G. (1968). State of the art in pattern recognition. *Proceedings of the IEEE*, 56(5), 836–863.

Newman, M. E. & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 69.

Newman, M. E. J. (2003). The structure and function of complex networks. *SIAM Review*, 45(2), 167–256.

Newman, M. E. J. (2004a). Detecting community structure in networks. *The European Physical Journal B*, 38(2), 321–330.

Newman, M. E. J. (2004b). Fast algorithm for detecting community structure in networks. *Physical Review E*, 69, 066133.

Ng, R. T. & Han, J. (2002). Clarans: A method for clustering objects for spatial data mining. *IEEE Transactions on Knowledge and Data Engineering*, 14(5), 1003–1016.

Olfati-saber, R., Fax, J. A., & Murray, R. M. (2007). Consensus and cooperation in networked multi-agent systems. In *Proceedings of the IEEE* (pp. 2007).

Olfati-Saber, R. & Murray, R. (2004). Consensus problems in networks of agents with switching topology and time-delays. *Automatic Control, IEEE Transactions on*, 49(9), 1520 – 1533.

Pang-Ning Tan, Michael Steinbach, V. K. (2006). *Introduction to DataMining*. Addison-Wesley.

Porfiri, M. & Fiorilli, F. (2009). Node-to-node pinning control of complex networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 19(1), 013122.

Quiles, M. G., Zhao, L., Alonso, R. L., & Romero, R. A. F. (2008). Particle competition for complex network community detection. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 18(3), 033107.

Rosvall, M. & Bergstrom, C. T. (2007). An information-theoretic framework for resolving community structure in complex networks. *physics/0612035*.

Silva, T. C. & Zhao, L. (2007). Pixel clustering by using complex network community detection technique. *Intelligent Systems Design and Applications, International Conference on*, (pp. 925–932).

Sponrs, O. (2002). Networks analysis, complexity, and brain function. *Complexity*, 8, 56–60.

Strogatz, S. H. (2001). Exploring complex networks. *Nature*, 410(6825), 268–276.

Tan, P.-N., Steinbach, M., & Kumar, V. (2006). *Introduction to Data Mining*. Addison Wesley.

Wang, X. F. & Chen, G. (2002). Pinning control of scale-free dynamical networks. *Physica A: Statistical Mechanics and its Applications*, 310(3-4), 521 – 531.

Watts, D. J. & Strogatz, S. H. (1998). Collective dynamics of small-world networks. *Nature*, 393(6684), 440–442.

Xiang, L., Liu, Z., Chen, Z., Chen, F., & Yuan, Z. (2007). Pinning control of complex dynamical networks with general topology. *Physica A: Statistical Mechanics and its Applications*, 379(1), 298 – 306.

Zachary, W. W. (1977). An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33, 452.

Zhou, H. (2003a). Distance, dissimilarity index, and network community structure. *Physical Review E*, 67, 061901.

Zhou, H. (2003b). Network landscape from a brownian particle's perspective. *Physical Review E*, 67, 041908.