UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
Câmpus de São José do Rio Preto

Larissa Moura

Agrupamento espectral através de grafos Laplacianos e uma aplicação no cultivo da soja.

São José do Rio Preto
2018

Larissa Moura

Agrupamento espectral através de grafos Laplacianos e uma aplicação no cultivo da soja.

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Matemática, junto ao Programa de Pós-Graduação em Matemática, do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Câmpus de São José do Rio Preto.

Orientadora: Profa. Dra. Alice Kimie Miwa Libardi

São José do Rio Preto
2018

Moura, Larissa.
Agrupamento espectral através de grafos Laplacianos e uma aplicação no cultivo da soja / Larissa Moura. -- São José do Rio Preto, 2018
87 f. : il.

Orientador: Alice Kimie Miwa Libardi
Dissertação (mestrado) – Universidade Estadual Paulista “Júlio de Mesquita Filho”, Instituto de Biociências, Letras e Ciências Exatas

1. Matemática. 2. Topologia. 3. Teoria dos grafos. 4. Soja. I. Universidade Estadual Paulista "Júlio de Mesquita Filho". Instituto de Biociências, Letras e Ciências Exatas. III. Título.

CDU – 518.71

Ficha catalográfica elaborada pela Biblioteca do IBILCE
UNESP - Câmpus de São José do Rio Preto

Larissa Moura

Agrupamento espectral através de grafos Laplacianos e uma aplicação no cultivo da soja.

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Matemática, junto ao Programa de Pós-Graduação em Matemática, do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista "Júlio de Mesquita Filho", Câmpus de São José do Rio Preto.

Comissão Examinadora

Profa. Dra. Alice Kimie Miwa Libardi
Orientadora

Prof. Dr. Thiago de Melo
UNESP/Rio Claro - SP

Prof. Dr. Washington Mio
Florida State University - Tallahassee, USA

São José do Rio Preto
16 de fevereiro de 2018

RESUMO

O objetivo desta dissertação é apresentar uma versão detalhada do artigo: “A Tutorial on Spectral Clustering” de U. von Luxburg sobre agrupamentos através de grafos Laplacianos, suas propriedades e mostrar alguns resultados da teoria de agrupamentos. Além disso, serão apresentados três algoritmos de agrupamentos e ilustraremos um deles com uma aplicação no cultivo da soja em diferentes condições de cultivo.

Palavras-chave: Agrupamentos, Grafo Laplaciano, Análise Topológica de Dados, Algoritmos de Agrupamentos.

ABSTRACT

The main goal of this dissertation is to present a detailed version of the paper: " A Tutorial on Spectral Clustering" of U. von Luxburg on clusters, through Laplacian graphs, their properties and to show some results of the cluster theory. In addition, it will be presented three clustering algorithms and we will illustrate one of them with an application in the soybean cultivation, under different conditions.

Keywords: Clustering, Laplacian Graph, Topological Data Analysis, Clustering Algorithms.

Aos meus amados pais
e queridos irmãos.

Agradecimentos

Primeiramente agradeço a minha família por toda a paciência e motivação. Em especial a minha mãe, os meus irmãos e os meus padrinhos pelas palavras de apoio quando eu acreditava ser impossível, pelo carinho, amor e por acreditarem em mim.

Agradeço principalmente minha orientadora, Profa. Dra. Alice Libardi que nunca mediu esforços, tampouco poupou dedicação para ensinar, pesquisar e me guiar durante esse processo.

Também sou grata á minha orientadora da graduação Profa. Dra. Eliris Cristina Rizzioli pelos ensinamentos compartilhados e por me inspirar e incentivar a fazer esse trabalho.

Agradeço a Profa. Dra. Leandra Bordignon (UFAC) que gentilmente disponibilizou os dados que foram utilizados neste trabalho, e aos professores Thiago de Melo, Jamil Viana Pereira por fornecer o trabalho realizado em cima desses dados. Agradeço também ao Bruno Zumpano, Northon Cannevari Penteado e Sérgio Tsuyoshi Ura pelo apoio e por todas as horas gastas me ajudando a fazer a programação sem a qual a aplicação desenvolvida nessa dissertação não teria sido possível.

Aos grandes amigos que fiz em Rio Claro, pessoas que se tornaram especiais e que foram minha família durante todo esse período. Amigos que levarei para o resto da minha vida.

Agradeço imensamente a Coordenação de Aperfeiçoamento de Pessoal de Nivel Superior, CAPES, por possibilitar a existência desse trabalho por meio do auxílio financeiro. Por fim, agradeço a todos que, direta ou indiretamente, tiveram participação ou influência no desenvolvimento deste trabalho.

Alguns homens vêm as coisas como são, e dizem
Por quê?
Eu sonho com as coisas que nunca foram e digo
Por que não?
Geroge Bernard Shaw

Lista de Figuras

Sumário

1 Introdução

Agrupamento é uma ferramenta para explorar a estrutura de dados que não requer os pressupostos comuns na maioria dos métodos estatísticos. As técnicas de agrupamento desempenham um papel central em várias partes da análise de dados, com aplicações variando de engenharias a biologia, psicologia, medicina, etc. Elas podem dar indícios importantes á estrutura dos conjuntos de dados e, portanto, sugerir resultados e hipóteses nas ciências.

Existem muitos métodos interessantes de agrupamento disponíveis, que foram aplicados com relativo sucesso ao lidar com conjuntos com grande quantidade de dados e são considerados métodos importantes na análise de dados exploratórios. Organizar dados em agrupamentos (clusters) é um dos mais fundamentais modos de entender as propriedades dos dados. A análise de agrupamentos é um estudo formal de algoritmos e métodos para agrupamentos de objetos.

Essa dissertação está dividida em 9 capítulos. No capítulo dois introduzimos os conceitos básicos de grafos e agrupamentos em grafos e no capítulo três algumas propriedades do grafo Laplaciano. No capítulo quatro apresentamos três algoritmos do agrupamento espectral e nos três capítulos seguintes explicações do por que esses algoritmos funcionam, sendo que o capítulo cinco descreve uma abordagem a partir da teoria de partição do grafo, no capítulo seis a teoria de caminhos aleátorios no grafo e no sétimo é dada uma abordagem através da teoria da perturbação. No capítulo oito estudamos problemas que aparecem ao utilizar o agrupamento espectral e finalmente no capítulo nove mostramos uma aplicação de um dos algoritmos no cultivo de soja.

2 Grafos e Agrupamentos

Intuitivamente, um grafo $G = (V, E)$ é uma estrutura formada por um conjunto V de vértices e um conjunto E de arestas, onde uma aresta é um segmento que conecta um par de vértices.

Chamamos de agrupamento o processo de colocar em um mesmo grupo aqueles dados que possuem um determinado tipo de similaridade. Tais conjuntos são chamados agrupamentos (clusters).

Agrupamentos em grafos visam separar em conjuntos de vértices que tenham a mesma estrutura, de modo que estes conjuntos formem uma partição do conjunto de vértices do grafo.

Considere um conjunto de dados, formado por pontos, $x_1, x_2, \ldots, x_n$, e alguma noção de similaridade, $s_{ij} \geq 0$, entre os pares de pontos , x_i e x_j, o objetivo intuitivo do agrupamento é dividir os pontos em alguns grupos tais que os pontos em um mesmo grupo são similares e pontos em grupos diferentes não o são.

No caso dos dados serem grafos $G = (V, E)$, os vértices v_i representam os pontos x_i. Diremos que dois vértices são conectados se a similaridade, s_{ij}, entre os pontos correspondentes, x_i e x_j, é positiva e a aresta é ponderada por s_{ij}. Dessa forma, podemos abordar o problema de agrupamentos, usando similaridade de grafos. Grosseiramente falando, queremos achar uma partição do grafo tal que as arestas entre dois grupos distintos têm um peso muito baixo. Posteriormente, essas idéias serão formalizadas.

Assumiremos que o grafo G com conjuntos de vértices, $V = \{v_1, v_2, \ldots, v_n\}$, é não orientado e ponderado, ou seja, cada aresta entre dois vértices, v_i e v_j, leva consigo um peso positivo, w_{ij}.

Definição 2.1. *A matriz de adjacência ponderada é a matriz* $W = (w_{ij})_{i,j \in \{1,2,\ldots,n\}}$
O grau do vértice $v_i \in V$ *é definido como*

$$d_i = \sum_{j=1}^{n} w_{ij}.$$

Se não existe uma aresta entre v_i *e* v_j *dizemos que* $w_{ij} = 0$*, assim a soma é feita apenas sobre os vértices adjacentes a* v_i*.*

Definição 2.2. *Definimos a matriz grau* D *como sendo uma matriz diagonal com os graus* $d_1, d_2, \ldots, d_n$ *na diagonal.*

Dado um subconjunto de vértices $A \subset V$, denotamos o complemento, $V - A$, por A^c.

Definição 2.3. *Definimos o vetor "indicador"* $\mathbf{1}_A = (f_1, f_2, \ldots, f_n)^t \in \mathbb{R}^n$ *como o vetor cujas entradas satisfazem* $f_i = 1$ *se* $v_i \in A$ *e* $f_i = 0$ *caso* $v_i \in A^c$*.*

Por conveniência, usaremos a notação $i \in A$ para o conjunto de indices $\{i|v_i \in A\}$.

Consideraremos duas maneiras diferentes para medir o "tamanho" de um conjunto $A \subset V$:

- $|A|$ = o número de vértices em A;
- $vol(A) = \sum_{i \in A} d_i$.

Intuitivamente, $|A|$ mede o tamanho de A pelo seu número de vértices, enquanto $vol(A)$ mede o tamanho de A pelos pesos de suas arestas.

Definição 2.4. *Um subconjunto $A \subset V$ de um grafo é conexo, se quaisquer dois vértices em A podem ser unidos por um caminho, de modo que todos os pontos intermediários também estão em A. Além disso, A é uma componente conexa, se A é conexo e se não existem arestas entres os vértices de A e A^c. Os conjuntos $A_1, A_2, \ldots, A_k$ formam uma partição do grafo se $A_i \cap A_j = \emptyset$ e $V = A_1 \cup A_2 \cup \cdots \cup A_k$.*

2.1 Diferentes Similaridades nos Grafos

Dado um conjunto de dados, observe que, decidir se dois elementos são similares ou não é uma questão aberta. Por exemplo, para decidir se duas cores são similares é um processo completamente diferente do que decidir se dois parágrafos de texto são similares. Claramente, então, antes que possamos decidir se são similares, precisamos encontrar uma maneira de comparar os objetos.

Qualquer conjunto de dados contém uma estrutura diferente entre eles, devido a heterogeneidade dos dados, uma vez que 8nossos dados podem ser de muitos tipos diferentes, por exemplo, pode ser um número, uma cor, uma localização geográfica, uma resposta verdadeira/falsa a uma pergunta, o que exigiria diferentes maneiras de medir a similaridade, precisamos, então, primeiro processar os dados no banco de dados de forma a garantir que podemos compará-los. Uma maneira comum de fazer isso é tentar converter todas as nossas características em um valor numérico, como converter cores para valores RGB, convertendo locais para latitude e longitude.

Uma vez que temos tudo como números, podemos imaginar um espaço no qual cada uma de nossas características é representado por uma dimensão diferente, e o valor de cada número para cada característica é a sua coordenada nessa dimensão. Então, nossos dados tornam-se pontos no espaço e podemos interpretar a distância entre eles como sendo sua similaridade (usando alguma métrica apropriada).

No capítulo 9 nosso conjunto de dados inicial consiste de fotos coloridas de folhas de soja, vários processos foram feitos para tornar essas fotos em imagens preto e branco e por fim foram calculadas as áreas de cada folha e esses números usados como coordenadas.

Existem várias construções para obter um grafo a partir de um determinado conjunto de dados, $x_1, x_2, \ldots, x_n$, com similaridades s_{ij} entre os pares ou com distância d_{ij} entre os pares. O objetivo ao construir grafos com determinada similaridade é modelar a relação numa vizinhança dos pontos. Além disso, a maioria das contruções abaixo leva a uma representação com menos dados, o que tem vantagens computacionais.

Enumeramos abaixo as diferentes similaridades que usaremos no nosso trabalho, baseados em [22], e para ilustra-las consideremos um conjunto aleatório com doze pontos.

- ε-vizinhança (ε-neighborhood): Os pontos cujas distâncias entre os pares são menores do que ε são conectados.

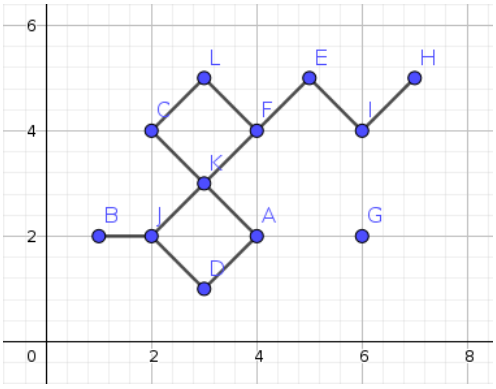

Figura 2.1: ε-vizinhança, $\varepsilon = 1,5$

Como as distâncias entre cada par de pontos são no maximo ε, ponderar as arestas não irá incorporar mais informações sobre os dados. Assim o grafo ε-vizinhança é geralmente considerado como um grafo não ponderado.

- k-vizinhos mais próximos (k-nearest neighbor): Aqui o objetivo é conectar vértices, v_i com vértices v_j se v_j está entre os k-vizinhos mais próximos de v_i. Esta definição leva a um grafo orientado, uma vez que a relação não é simétrica.

  Existem duas maneiras de fazer desse grafo, um grafo não orientado. A primeira maneira é simplesmente ignorar as direções das arestas, isto é, conectando v_i e v_j com uma aresta sem direção se v_i está entre os k-vizinhos mais próximos de v_j ou se v_j está entre os k-vizinhos mais próximos de v_i. O grafo resultante é o que chamamos de *k-vizinhos mais próximos.*

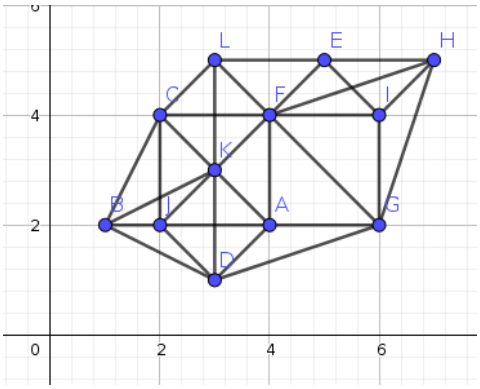

Figura 2.2: k-vizinhos mais próximos, $k = 4$

A segunda opção é ligar vértices v_i e v_j se, v_i está entre os k-vizinhos mais próximos de v_j e v_j está entre os k-vizinhos mais próximos de v_i. O grafo resultante é chamado *k-vizinhos mutualmente mais próximos (mutual k-nearest neighbor).* Em ambos os casos, depois de conectar os vértices apropriados, ponderamos as arestas pelas similaridades dos pontos adjacentes.

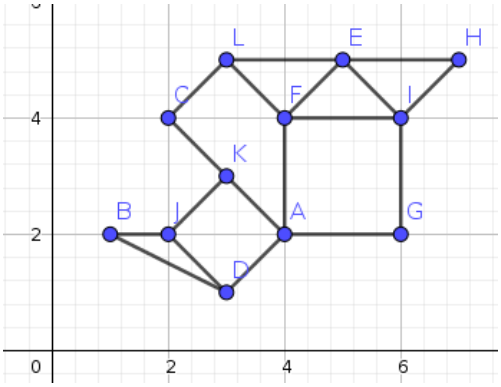

Figura 2.3: k-vizinhos mutualmente mais próximos, $k = 4$

- O grafo totalmente conexo (the fully connected graph): Nesta caso, simplesmente conectamos todos os pontos com similaridade positiva e ponderamos todas as arestas por s_{ij}.

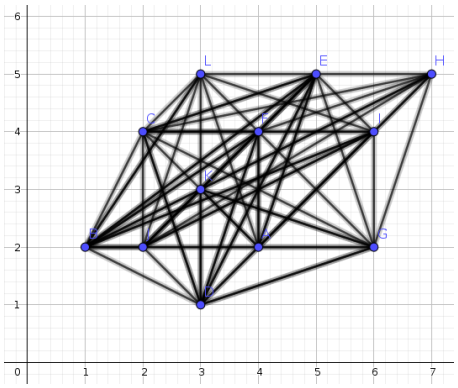

Figura 2.4: Grafo totalmente conexo

Como o grafo deveria modelar as relações de vizinhança local, esta construção é usualmente escolhida apenas se a própria função similaridade já codifica as principais vizinhanças locais.

Um exemplo de função similaridade, onde isto ocorre, é a função *Gaussiana de similaridade* $s(x_i, x_j) = e^{\left(\frac{-\|x_i - x_j\|^2}{2\sigma^2}\right)}$. Aqui o parâmetro σ controla a largura das vizinhanças, de modo similar ao parâmetro ε no primeiro caso.

Note que os grafos gerados são diferentes, no caso do ε-vizinhança o ponto G fica completamente isolado do resto dos vértices do grafo, o que não ocorre nos demais grafos. A questão que se coloca aqui é qual grafo deve ser usado. Há estudos teóricos que afirmam que mesmo em circunstâncias particulares, não há como decidir.

2.2 Os k-vizinhos mais próximos

Como em nossa aplicação usaremos a similaridade do k-vizinhos mais próximos vamos fazer algumas observações sobre o mesmo.

Dado um conjunto de dados finito, $X = \{x_1, \ldots, x_n\}$, em um espaço métrico (M, d) qualquer, o algoritmo para construir o grafo k-vizinhos mais próximos segue os seguintes passos:

1) Dado $x \in X$ e $k \geq 1$ definimos o vetor:

$$(x, X) = (d(x, x_1), \ldots, d(x, x_n))$$

2) Tomamos as k entradas de (x, X) que possuem os menores valores, desconsiderando a entrada nula. Denotaremos por $V_k(x)$ o conjunto de tais pontos.

3) Construímos o conjunto

$$A_k(x) = \{xx_i, \forall x_i \in V_k(x)\},$$

onde xx_i denota o segmento que conecta o ponto x ao ponto x_i.

4) Por fim, construímos o grafo k-vizinhos mais próximos, $G = (V, A)$, onde $V = X$ e $A = \cup A_k$.

Observe que no passo (2), se tivermos duas entradas iguais, ou seja, $d(x, x_i) = d(x, x_j)$, com $i < j$ mas só pudermos tomar uma, então consideramos apenas o x_i.

Podemos, também, construir o grafo k-vizinhos mutualmente mais próximos, que como comentamos anteriormente é o resultado quando queremos transformar o grafo k-vizinhos mais próximos em um grafo não orientado. Para isso trocamos o passo (4) para:

4') $G = (V, A)$ é um grafo k-vizinhos mutualmente mais próximos, onde $V = X$, e xy será uma aresta desse grafo se, e só se, $x \in V_k(y)$ e $y \in V_k(x)$, ou equivalentemente, $xy \in A_k(x) \cap A_k(y)$.

Exemplo 2.1. Considere o conjunto $X = \{A, B, C, D, E, F, G, H, I, J, K, L\}$, como na seção 2.1, em $\mathbb{R}^2$ com a métrica usual. No passo (1) definimos o vetor, (x, X) para todo $x \in X$, como o vetor a seguir:

$$(A, X) = (d(A, A), \ldots, d(A, L)) = (0, 3, 2.83, 1.41, 3.16, 2, 2, 4.24, 2.83, 2, 1.41, 3.16)$$

No passo (2) tomamos os $k = 4$ menores valores do vetor, no caso do (A, X) são as entradas que correspondem ao pontos, D, F, G, K note que $d(A, F) = d(A, G) = d(A, J) = 2$, contudo só podemos tomar dois deles, uma vez que já temos dois com valores $1, 41$, e portanto desconsideramos a entrada correspondente ao ponto J. Neste caso, consideramos a ordem alfabética como ordem dos pontos, e os pontos foram gerados de forma aleatória. E obtemos o grafo da figura 2.2.

3 Propriedades do Grafo Laplaciano

As matrizes do grafo Laplaciano fornecem os dados para obtermos os agrupamentos. Neste trabalho apresentaremos diferentes grafos Laplacianos e suas propriedades mais importantes. Note que na literatura, não existe uma convenção de qual matriz, exatamente, é a matriz chamada de matriz do grafo Laplaciano.

No que se segue sempre assumiremos que G é um grafo não orientado, ponderado com a matriz peso W, com pesos não negativos, $w_{ij} = w_{ji}$. Quando falarmos dos autovetores da matriz, não necessariamente assumiremos que eles estão normalizados. Por exemplo, o vetor constante $\mathbf{1}$ e um múltiplo $a\mathbf{1}$ para algum $a \neq 0$ são considerados como o mesmo autovetor. Os autovalores serão sempre ordenados em ordem crescente, respeitando as multiplicidades. Por *os primeiros k autovetores* estaremos nos referindo aos autovetores correspondentes aos k menores autovalores.

3.1 O Grafo Laplaciano não Normalizado

Definição 3.1. *A matriz do grafo Laplaciano não normalizado é a matriz*

$$L = D - W,$$

onde as matrizes D e W são as matrizes grau e adjacência ponderada, respectivamente, definidas anteriormente.

Proposição 3.2. *Laços no grafo não mudam a matriz do grafo Laplaciano correspondente.*

Demonstração. Por definição temos que:

$$W = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \vdots & \dots & \ddots & \vdots \\ w_{n1} & \dots & w_{nn-1} & w_{nn} \end{bmatrix},$$

$$D = \begin{bmatrix} d_1 & 0 & \dots & 0 \\ 0 & d_2 & \dots & 0 \\ \vdots & \dots & \ddots & \vdots \\ 0 & \dots & 0 & d_n \end{bmatrix}.$$

Como, $d_i = \sum_{j=1}^{n} w_{ij}$, e $L = D - W$, podemos escrever

$$L = \begin{bmatrix} \sum_{j\neq 1} w_{1j} & -w_{12} & \dots & -w_{1n} \\ -w_{21} & \sum_{j\neq 2} w_{2j} & \dots & -w_{2n} \\ \vdots & \dots & \ddots & \vdots \\ -w_{n1} & \dots & -w_{nn-1} & \sum_{j\neq n} w_{nj} \end{bmatrix}.$$

Ou seja, os elementos da diagonal de L, são da forma:

$$l_{ii} = \sum_{j\neq i} w_{ij}.$$

Portanto, os elementos da diagonal de L não dependem dos elementos da diagonal de W. Dessa forma, toda matriz U que coincide com W em todas as posições fora da diagonal representa o mesmo grafo Laplaciano não normalizado L. □

Proposição 3.3. *A matriz L satisfaz as seguintes propriedades:*

1) Para todo vetor $f \in \mathbb{R}^n$ tem-se

$$f^t L f = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2.$$

2) L é simétrica e positiva semi-definida, ou seja, $f^t L f \geq 0$, para todo $f \in \mathbb{R}^n$.

3) O menor autovalor de L é zero, e um autovetor correspondente é o vetor constante 1.

4) L tem n autovalores reais não negativos $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$

Demonstração. 1) Pela definição de L, temos que

$$f^t L f = f^t(D - W)f = f^t D f - f^t W f,$$

sendo D a matriz grau e $W = (w_{ij})$ a matriz de adjacência ponderada. Logo,

$$f^t L f = \sum_{i=1}^{n} d_i f_i^2 - \sum_{i,j=1}^{n} f_i f_j w_{ij},$$

$$\implies f^t L f = \frac{1}{2}\Big(\sum_{i=1}^{n} d_i f_i^2 - 2\sum_{i,j=1}^{n} f_i f_j w_{ij} + \sum_{j=1}^{n} d_j f_j^2 \Big).$$

Por definição, temos que $d_i = \sum_{j=1}^{n} w_{ij}$. Assim,

$$f^t L f = \frac{1}{2}\Big(\sum_{i=1}^{n} f_i^2 \Big(\sum_{j=1}^{n} w_{ij} \Big) - 2\sum_{i,j=1}^{n} f_i f_j w_{ij} + \sum_{j=1}^{n} f_j^2 \Big(\sum_{i=1}^{n} w_{ij} \Big) \Big).$$

Agrupando, obtemos a seguinte igualdade:

$$f^t L f = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij}(f_i^2 - 2 f_i f_j + f_j^2).$$

Portanto,

$$f^t L f = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2.$$

2) Como G é não orientado devemos ter $w_{ij} = w_{ji}$, logo a matriz W é simétrica. Além disso, D é uma matriz diagonal, logo, simétrica. Portanto L é simétrica. Pelo item (1), temos que para todo vetor $f \in \mathbb{R}^n$,

$$f^t L f = \frac{1}{2} \sum_{j=1}^{n} w_{ij}(f_i - f_j)^2 \geq 0$$

uma vez que, por definição, $w_{ij} \geq 0$.

3) Basta observar que: "L é positiva semi-definida se, e só se, todos os autovalores de L são positivos e existe um autovalor nulo". (ver [19]) Além disso, observe que para o autovalor nulo λ de L, v é um autovetor correspondente se $Lv = \lambda v = 0$. Resolvendo tal equação,obtemos o vetor constante 1.

4) Segue do fato de L ser positiva semi-definida que L possui n autovalores não negativos e o número de autovalores λ_i para os quais $\lambda_i > 0$ é igual ao posto de L, ver [19].

□

Proposição 3.4. *Seja G um grafo não orientado com pesos não negativos. Então a multiplicidade k do autovalor 0 de L é igual ao número de componentes conexas $A_1, \ldots, A_k$ no grafo. O autoespaço do autovalor 0 é abrangido pelo vetor indicador $\mathbf{1}_{A_1}, \ldots, \mathbf{1}_{A_k}$ dessas componentes.*

Demonstração. O caso $k = 1$, onde o grafo é conexo sai como consequência imedita da proposição 3.3 itens (3) e (4).

Agora considere o caso de k componentes conexas. Sem perda de generalidade assumiremos que os vértices são ordenados de acordo com as componentes conexas a que eles pertencem. Desse modo, a matriz adjacência ponderada W tem forma de bloco diagonal, e o mesmo é verdadeiro para a matriz L. Ou seja, podemos escrever:

$$L = \begin{bmatrix} L_1 & 0 & \ldots & 0 \\ 0 & L_2 & \ldots & 0 \\ \vdots & \ldots & \ddots & \vdots \\ 0 & \ldots & 0 & L_k \end{bmatrix}.$$

Note que cada bloco L_i é um grafo Laplaciano, a saber, o Laplaciano correspondente ao subgrafo da i-ésima componente conexa. Sabemos que o espectro de L é dado pela união dos espectros de L_i, preenchendo com 0 as posições restantes. Como cada L_i é um grafo Laplaciano de um grafo conexo, sabemos que todo L_i tem autovalor 0 com multiplicidade 1, e um correspondente autovetor é o vetor constante na i-ésima componente conexa.

Então a matriz L tem tantos autovalores nulos quanto o número de componentes conexas, e os respectivos autovetores são os vetores indicadores das componentes. □

3.2 O Grafo Laplaciano Normalizado

Na literatura existem duas matrizes que são chamadas de grafos Laplacianos normalizados. Ambas as matrizes são próximas uma da outra, no sentido da Prop. 3.6.

Nesta seção iremos considerar que todos os grafos possuem laços, dessa maneira temos que $d_i > w_{ii} > 0$ podemos fazer isso por causa do resultado visto na Prop. 3.2 e assim evitamos problemas na definição 3.5.

Definição 3.5. *A primeira matriz é denotada por, L_{sym}, por ser uma matriz simétrica e é definida por:*

$$L_{sym} = D^{\frac{-1}{2}} L D^{\frac{-1}{2}} = I - D^{\frac{-1}{2}} W D^{\frac{-1}{2}},$$

onde,

$$D^{\frac{-1}{2}} = \begin{bmatrix} \frac{1}{\sqrt{d_1}} & 0 & \dots & 0 \\ 0 & \frac{1}{\sqrt{d_2}} & \dots & 0 \\ \vdots & \dots & \ddots & \vdots \\ 0 & \dots & 0 & \frac{1}{\sqrt{d_n}} \end{bmatrix}.$$

A segunda matriz é denotada por, L_{rw}, por se tratar de um grafo conectado a um caminho aleátorio, e é definida como se segue.

$$L_{rw} = D^{-1} L = I - D^{-1} W,$$

onde D^{-1} denota a matriz inversa de D. Como D é diagonal, temos que:

$$D^{-1} = \begin{bmatrix} \frac{1}{d_1} & 0 & \dots & 0 \\ 0 & \frac{1}{d_2} & \dots & 0 \\ \vdots & \dots & \ddots & \vdots \\ 0 & \dots & 0 & \frac{1}{d_n} \end{bmatrix}.$$

Proposição 3.6. *1) Para todo $f \in \mathbb{R}^n$ temos*

$$f^t L_{sym} f = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} \left(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}} \right)^2.$$

2) λ é um autovalor de L_{rw} com autovetor v se, e só se, λ é autovalor de L_{sym} com autovetor $w = D^{\frac{1}{2}} v$.

3) λ é um autovalor de L_{rw} com autovetor v se, e só se, λ e v são soluções para

$$Lv = \lambda D v.$$

4) 0 é um autovalor de L_{rw} com o vetor constante $\mathsf{1}$ como autovetor, e 0 é um autovalor de L_{sym} com o vetor constante $D^{\frac{1}{2}}\mathsf{1}$ como autovetor.

5) L_{sym} e L_{rw} são positivas semi-definidas e possuem autovalores reais não negativos.

Demonstração. 1) Por definição, dado $f \in \mathbb{R}^n$, temos que

$$f^t L_{sym} f = f^t I f - f^t D^{\frac{-1}{2}} W D^{\frac{-1}{2}} f.$$

Logo,

$$f^t L_{sym} f = \sum_{j=1}^{n} f_j^2 - \sum_{i,j=1}^{n} w_{ij} \frac{f_i}{\sqrt{d_i}} \frac{f_j}{\sqrt{d_j}}.$$

Ou seja,

$$f^t L_{sym} f = \frac{1}{2}\Big(\sum_{i=1}^{n} f_i^2 \frac{d_i}{d_i} - 2 \sum_{i,j=1}^{n} w_{ij} \frac{f_i}{\sqrt{d_i}} \frac{f_j}{\sqrt{d_j}} + \sum_{j=1}^{n} f_j^2 \frac{d_j}{d_j} \Big).$$

Temos, por definição que, $d_j = \sum_{i=1}^{n} w_{ij}$, assim obtemos a seguinte igualdade:

$$f^t L_{sym} f = \frac{1}{2}\Big(\sum_{i,j=1}^{n} w_{ij} \frac{f_i^2}{d_i} - 2 \sum_{i,j=1}^{n} w_{ij} \frac{f_i}{\sqrt{d_i}} \frac{f_j}{\sqrt{d_j}} + \sum_{i,j=1}^{n} w_{ij} \frac{f_j^2}{d_j} \Big).$$

Portanto,

$$f^t L_{sym} f = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} \left(\frac{f_i^2}{d_i} - 2 \frac{f_i}{\sqrt{d_i}} \frac{f_j}{\sqrt{d_j}} + \frac{f_j^2}{d_j} \right).$$

Consequentemente,

$$f^t L_{sym} f = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} \left(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}} \right)^2.$$

2) Observe que:

$$L_{sym} w = \lambda w$$

$$\iff$$

$$(I - D^{\frac{-1}{2}} W D^{\frac{-1}{2}}) w = \lambda w$$

$$\iff$$

$$D^{\frac{-1}{2}} (I - D^{\frac{-1}{2}} W D^{\frac{-1}{2}}) w = D^{\frac{-1}{2}} \lambda w$$

Substituindo, $v = D^{\frac{-1}{2}} w$, temos:

$$v - D^{-1} W v = \lambda v$$

$$\iff$$

$$(I - D^{-1} W) v = \lambda v$$

$$\iff$$

$$L_{rw} v = \lambda v$$

Ou seja, λ é autovalor de L_{rw} com autovetor v se, e só se, λ é autovalor de L_{sym} com autovetor $w = D^{\frac{1}{2}} v$.

3) Note que:

$$L_{rw}v = \lambda v$$

$$\iff$$

$$(I - D^{-1}W)v = \lambda v$$

$$\iff$$

$$D(I - D^{-1}W)v = D\lambda v$$

$$\iff$$

$$(D - W)v = \lambda Dv$$

$$\iff$$

$$Lv = \lambda Dv$$

Ou seja, λ é autovalor de L_{rw} com autovetor v se, e só se, λ e v são soluções para

$$Lv = \lambda Dv.$$

4) Pela *proposição* 3.4 temos que 0 é autovalor de L com autovetor $\mathsf{1}$, ou seja,

$$L\mathsf{1} = 0.$$

Observe que para $v = \mathsf{1}$ e $\lambda = 0$, temos que:

$$Lv = 0 = \lambda Dv.$$

Ou seja, $\lambda = 0$ e $v = \mathsf{1}$ são soluções para

$$Lv = \lambda Dv.$$

e consequentemente 0 é autovalor de L_{rw} com autovetor $\mathsf{1}$. De onde concluímos que 0 é autovalor de L_{sym} com autovetor $D^{\frac{1}{2}}\mathsf{1}$.

5) Do *item* (1) temos que:

$$f^t L_{sym} f = \frac{1}{2}\sum_{i,j=1}^{n} w_{ij}\left(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}}\right)^2,$$

de onde segue que L_{sym} é positiva semi-definida. E portanto possui autovalores reais não negativos. Do item (2) temos que λ_i, também é autovalor de L_{rw}, de onde concluímos que L_{rw} é positiva semi-definida.

□

Proposição 3.7. *Seja G um grafo não orientado com pesos não negativos. Então a multiplicidade k do autovalor 0 de L_{rw} e L_{sym} é igual ao número de componentes conexas $A_1, \ldots, A_k$ no grafo. Para L_{rw} o autoespaço de 0 é gerado pelo vetor indicador 1_{A_i} dessas componentes conexas. Para L_{sym} o autoespaço de 0 é gerado pelos vetores $D^{\frac{1}{2}}1_{A_i}$.*

Demonstração. Inicialmente provaremos o caso $k = 1$, isto é, consideraremos que o grafo é conexo. Assuma que f é um autovetor com autovalor 0. Então

$$0 = f^t L_{rw} f = \sum_{i,j=1}^{n} \frac{w_{ij}}{d_i}(f_i - f_j)^2.$$

Por definição os pesos, w_{ij}, são não negativos, assim, esta soma pode zerar apenas se todos os seus termos zerarem, ou seja, $\frac{w_{ij}}{d_i}(f_i - f_j)^2 = 0$ para todo i, j. Então, como G é conexo, temos que $\frac{w_{ij}}{d_i} > 0, \ \forall i, j$ e consequentemente devemos ter $f_i - f_j = 0$, logo $f_i = f_j$. Observe que $f_i \neq 0, \ \forall i$, pois se $f_1 = 0$, e considerando o termo da soma, para $i = 2 \dots n$ e $j = 1$ temos:

$$0 = \frac{w_{i1}}{d_i}(f_i - f_1)^2 = \frac{w_{i1}}{d_i} f_i^2$$

De onde concluiríamos que $f_i = 0, \ \forall i$, ou seja f seria o vetor nulo, contradizendo o fato de f ser um autovetor. Pelo item (4) da proposição 3.3 temos que 0 é autovalor com vetor constante 1 como um autovetor, que é claramente o vetor indicador da componente conexa. Pela proposição 3.6 item (2) temos que 0 é autovalor de L_{sym} com autovetor $D^{\frac{1}{2}}1$.

Como feito na demonstração da proposição 3.4 podemos escrever L_{rw} com blocos na diagonal e obter o desejado.

□

4 Algoritmos de Agrupamento Espectral

Vamos agora introduzir os algoritmos mais comuns de agrupamento espectral. Assumiremos que é dado um conjunto de pontos $x_1, \ldots, x_n$ que podem ser objetos arbitrários, e a sua similaridade, $s_{ij} = s(x_i, x_j)$, medida de acordo com alguma função similaridade simétrica e não negativa. Denotamos a matriz similaridade correspondente por $S = (s_{ij})_{i,j=1,\ldots,n}$.

Uma das maneiras de se obter o agrupamento espectral não normalizado é através do roteiro abaixo, de acordo com [23].

1) Construir o grafo $G = (V, E)$ e a sua matriz Laplaciana, L.

2) Calcular os k primeiros autovetores, $v_1, \ldots, v_k$, de L.

3) Considerar V a matriz formada pelos k autovetores nas colunas.

4) Tomar y_i o vetor correspondente a i-ésima linha de V.

5) Aplicar o algoritmo k-means nos vetores y_i.

Dependendo do grafo Laplaciano considerado, temos diferentes versões do agrupamento espectral normalizado.

O agrupamento de acordo com Shi e Malik(2000), segue os seguintes passos:

1) Dado um conjunto de dados, construir o grafo $G = (V, E)$.

2) Construir a matriz Laplaciana e resolver $Lv = \lambda Dv$, para os k primeiros autovetores, $v_1, \ldots, v_k$.

3) Usar o autovetor correspondente ao segundo autovalor para bipartição do grafo.

4) Decidir se tal partição deve ser subdividida e fazer uma partição recursiva se necessário.

Ao invés de usar tal processo de partição com dois cortes, Shi e Malik também recomendam o algoritmo k-means, que pode usar os autovetores simultaneamente, onde substituímos os passos 3 e 4 pelos seguintes passos:

3') Se V é a matriz contendo os autovetores $v_1, \ldots, v_k$ por coluna, considerar y_i o vetor correspondente a i-ésima linha de V.

4') Usar tais vetores no algoritmo k-means para obter os agrupamentos.

Note que este algoritmo usa autovetores generalizados de L, ou seja, os vetores que satisfazem $Lv = \lambda Dv$, o que de acordo com a proposição 3.6 corresponde aos autovetores da matriz L_{rw}.

Então o algoritmo funciona com autovetores da Laplaciana normalizada L_{rw}, e portanto é chamado de agrupamento espectral normalizado. O próximos algoritmo também usa a matriz Laplaciana normalizada, mas dessa vez usamos a matriz L_{sym} ao invés da L_{rw}. Como iremos ver este algoritmo precisa de um passo de normalização da linha que não é necessário no outro algoritmo. A razão irá se tornar clara em 7. Para exemplos ver [21]. O agrupamento de acordo com Ng. Jordan e Weiss (2002), segue os seguintes passos:

1) Dado um conjunto de dados, $X = \{x_1, \ldots, x_n\}$, construa o grafo $G = (V, E)$.

2) Construa a matriz Laplaciana, L_{sym}.

3) Calcule os k primeiros autovetores de L_{sym} e denote por V a matriz contendo os autovetores, $v_1, \ldots, v_k$, por coluna.

4) Forme a matriz Y a partir de V, normalizando as linhas de V, ou seja, $y_{ij} = \frac{v_{ij}}{(\sum_j v_{ij}^2)^{\frac{1}{2}}}$.

5) Trate cada linha de Y como sendo um ponto e os use como entradas, para o algoritmo k-means.

6) Finalmente, dizemos que o ponto original x_i pertence ao agrupamento j se, e só se, a linha i da matriz Y foi atribuída ao agrupamento j.

Ng. Jordan e Weiss [18] consideram a matriz Laplaciana, L_{sym}, como sendo $D^{-1/2}WD^{1/2}$.

Nesse caso há apenas uma mudança na ordem dos autovalores, então ao invés de tomar os k primeiros autovalores, são tomados os últimos k autovalores.

Todos os três algoritmos acima indicados são bastante parecidos, além do fato de que eles usam os grafos de três Laplacianas diferentes. Em todos os três algoritmos, o truque principal é mudar a representação dos pontos de dados abstratos x_i para $y_i \in \mathbb{R}^k$. Devido as propriedades do grafo Laplaciano essa mudança de representação é útil. Veremos que esta mudança de representação aumenta as propriedades do agrupamento nos dados, de modo que eles podem ser detectados trivialmente na nova representação. Em particular, o algoritmo k-means não tem dificuldade para detectar os agrupamentos nesta nova representação.

Exemplo 4.1. Para ilustrar os algoritmos citados acima, vamos apresentar um exemplo que pode ser encontrado em [23] e será usado várias vezes. Este conjunto de dados contém 200 pontos $x_1, \ldots, x_{200} \in \mathbb{R}$ desenhados de acordo com uma mistura de quatro Gaussianas. A primeira linha da imagem 4.1 mostra o histograma do conjunto de dado. Como função similaridade foi escolhida a função similaridade Gaussiana

$$s(x_i, x_j) = \exp(\frac{-|x_i - x_j|^2}{2\sigma^2}),$$

com $\sigma = 1$. Como grafo similaridade, foi considerado grafo totalmente conexo e o grafo k-vizinhos mais próximos com $k = 10$. Na figura 4.1 a primeira coluna mostra os dez primeiro autovalores da matriz Laplaciana não normalizada, L, e da normalizada, L_{rw},

onde foi plotado $i \times \lambda_i$, no momento as cores e formatos dos autovalores não importam, eles serão importantes quando estudarmos esse exemplo novamente no capítulo 8. Nas figuras do autovetores foi plotado $x_i \times v_i$ para um autovetor $v = (v_1, \ldots, v_{200})^t$.

As primeiras duas linhas mostram o resultado baseado no grafo do k-vizinhos mais próximos. Podemos ver que os quatro primeiros autovalores estão próximos de zero e os autovetores correspondentes são os vetores indicadores. A razão é que os agrupamentos formam partes desconexas no grafo.

As próximas duas linhas mostram o resultado do grafo totalmente conexo. Como a função Gaussiana é sempre positiva, este grafo consiste de uma única componente conexa. Portanto, o autovalor 0 tem multiplicidade 1, e o autovetor correspondente é o vetor constante. Os autovetores carregam informações sobre os agrupamentos, se limitarmos o segundo autovetor por 0, então a parte abaixo de 0 corresponde aos agrupamentos 1 e 2 e a parte acima de 0 para os agrupamentos 3 e 4. Similarmente, se limitarmos o terceiro autovetor separamos os agrupamentos 1 e 4 dos agrupamentos 2 e 3, e limitando o quarto separamos os agrupamentos 1 e 3 dos agrupamentos 2 e 4. No total os quatro primeiros autovetores carregam todas as informações dos quatro agrupamentos. Em todos os casos ilustrados nesta figura em [22], o agrupamento espectral usando k-means nos primeiros quatro autovetores detectam facilmente os quatros agrupamentos corretamente.

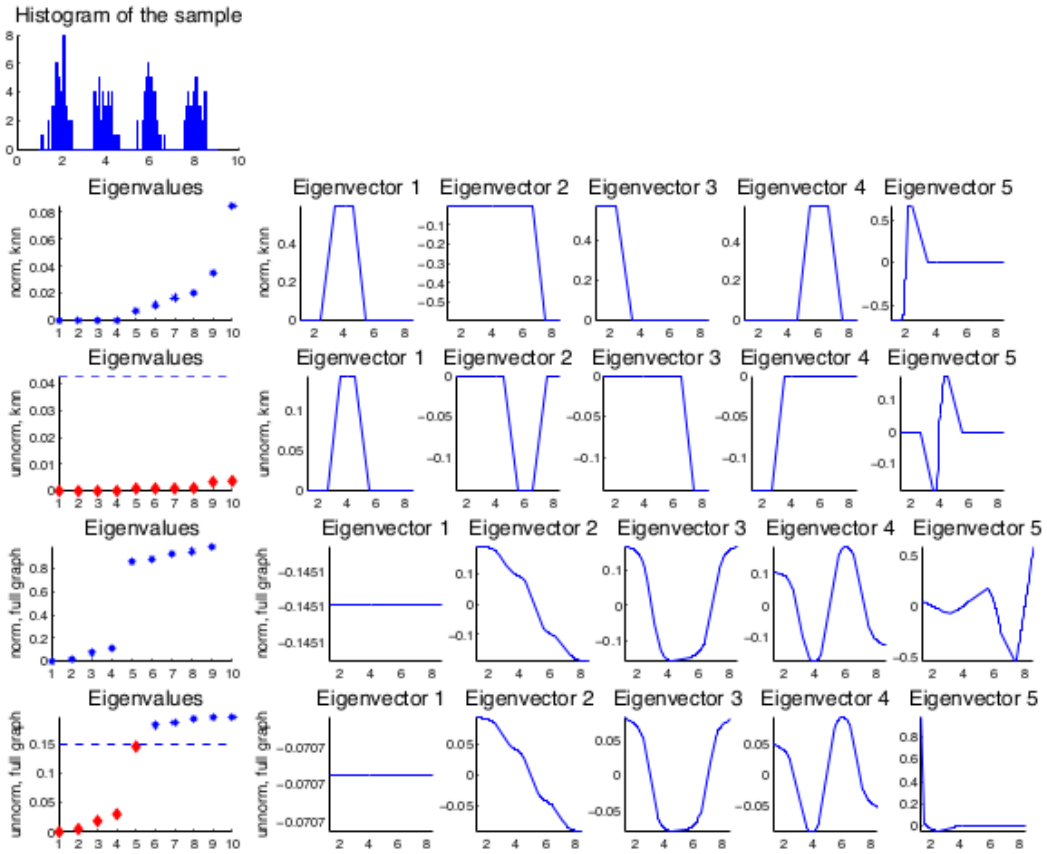

Figura 4.1: Exemplos de agrupamentos de Von Ulrike, [23].

A primeira vista o algoritmo não parece fazer muito sentido. Uma vez que rodamos

o algorítimo k-means apenas no passo 5 e por que não aplicar diretamente no conjunto de dados? O agrupamento natural em $\mathbb{R}^2$ não corresponde a regiões convexas, e aplicando o algoritmo k-means diretamente teríamos um agrupamento insatisfatório na figura 4.2(i). Mas uma vez que aplicamos os pontos para $\mathbb{R}^k$ (linhas de Y), formam-se agrupamentos pequenos como 4.2(h) do qual o algoritmo de [18] obtém agrupamentos, como na figura 4.2(e).

Exemplo 4.2. Para testar o algoritmo Ng. Jordan e Weiss aplica-se o algoritmo em sete problemas de agrupamentos, e o resultado são mostrados na figura 4.2(a-g). Dando ao algoritmo apenas as coordenadas dos pontos e o valor de k, os agrupamentos diferentes são mostrados nas figuras com simbolos e cores diferentes. O resultado é realmente bom, mesmo os agrupamentos que não formam regiões convexas ou que não são claramente separados (como na figura 4.2(g)).

Eles também apresentam um algoritmo baseado em k vetores na figura 4.2(l). Além disso Ng. Jordan e Weiss também comparam seu algoritmo com o de Meila e Shi [15], na figura 4.2(k).

4.1 k-means

O k-means (que optamos por não traduzir) é um método de encontrar agrupamentos e os seus centros a partir de um conjunto de dados. Dado um conjunto de dados, são escolhidos centros iniciais aleatoriamente, após isso o algoritmo k-means se alterna em dois passos:

- Para cada centro identificamos o subconjunto de pontos que está mais perto desse centro do que qualquer outro.
- A média total entre os pontos do subconjunto acima são calculadas, e este vetor média se torna o novo centro do agrupamento.

Esses dois passos são iterados até convergirem. Normalmente os centros iniciais são escolhidos aleatoriamente a partir de um conjunto inicial, chamado de conjunto de treinamento.

Exemplo 4.3. A figura 4.3 mostra um exemplo simulado com $k = 5$, com três classes diferentes e mostra as regiões de classificação e as linhas tracejadas mostra o limite da decisão dessas regiões. Para mais detalhes ver [9].

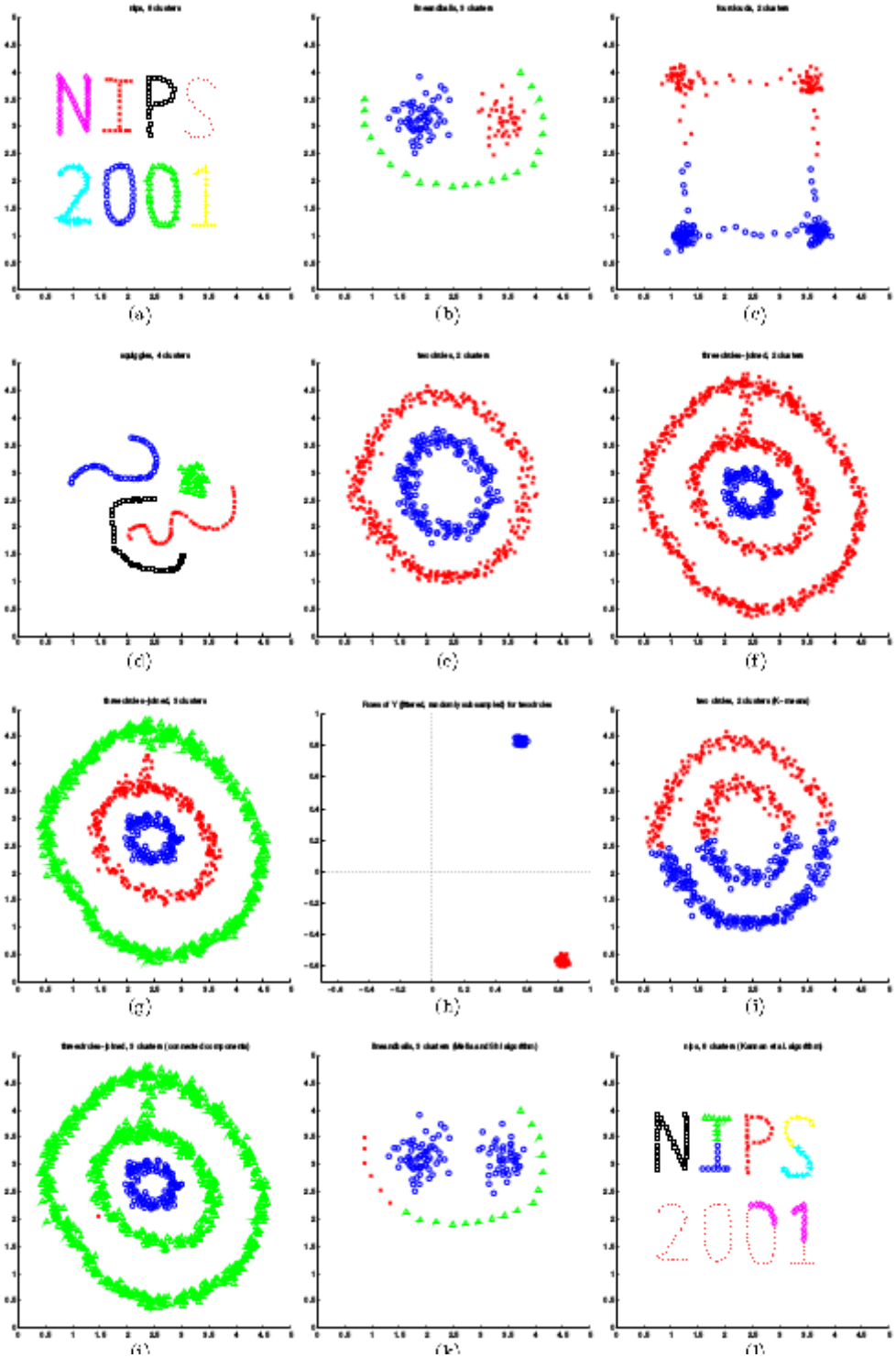

Figura 4.2: Exemplos de agrupamentos de Ng, Jordan e Weiss, [18]

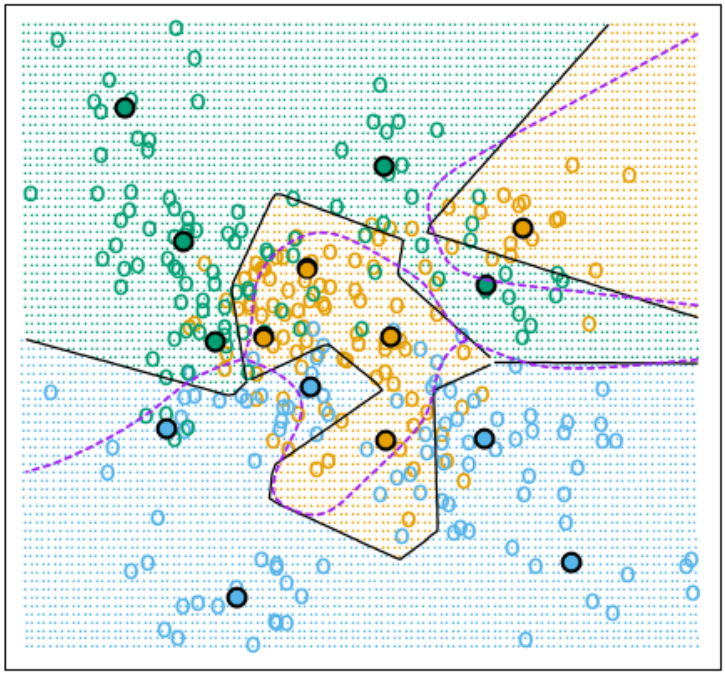

Figura 4.3: k-means para $k = 5$ de Hastie, Tibshirani e Friedman, [9].

5 Partições dos Grafos

Como já vimos anteriormente, podemos reformular o problema do agrupamento, usando a similaridade de grafos, achando uma partição do grafo tal que as arestas entre dois grupos distintos tenham um peso muito baixo.

Agora veremos como o agrupamento espectral pode ser obtido como uma aproximação a tais problemas de particionamento de grafo.

Definição 5.1. *Dados dois subconjuntos disjuntos, $A, B \subset V$, definimos, o corte entre A e B:*

$$cut(A, B) = \sum_{i \in A, j \in B} w_{ij}.$$

Dado um grafo, G, com matriz de adjacência ponderada, W, o jeito mais simples e mais direto de se construir uma partição é resolver o problema do "corte minimo", o qual consiste na escolha da partição, $A_1, \ldots, A_k$, o que minimiza:

$$cut(A_1, \ldots, A_k) = \sum_{i=1}^{n} cut(A_i, A_i^c).$$

Em particular, para $k = 2$, o corte mínimo é um problema relativamente simples, mas que em geral não nos dá partições satisfatórias, pois em muitos casos, a solução do corte mínimo consiste em separar um único vértice do resto do grafo. Claramente, não é isso que queremos no agrupamento, uma vez que queremos agrupamentos, razoavelmente grandes. Um jeito de contornar esse problema, é exigir que os conjuntos $A_1, \ldots, A_k$ sejam razoavelmente grandes. Existem duas funções que resolvem isso, a primeira, o corte proporcional denotada por $RCut$ onde o tamanho de um subconjunto A do grafo é medido pelo número de seus vértices, $|A|$, enquanto que no segundo, $NCut$, o tamanho é medido pelo peso de suas arestas, $vol(A)$.

Definição 5.2. *Seja $A_1, \ldots, A_k$ uma partição de G. Definimos as seguintes funções:*

$$RCut(A_1, \ldots, A_k) = \sum_{i=1}^{k} \frac{cut(A_i, A_i^c)}{|A_i|}.$$

$$NCut(A_1, \ldots, A_k) = \sum_{i=1}^{n} \frac{cut(A_i, A_i^c)}{vol(A_i)}.$$

Note que ambas as funções tomam valores pequenos nos agrupamentos A_i que não são tão pequenos. Em particular, o mínimo da função $\sum_{i=1}^{k} \frac{1}{|A_i|}$ é alcançado se todos $|A_i|$ coincidem, e o minimo de $\sum_{i=1}^{k} \frac{1}{vol(A_i)}$ é alcançado se todos os $vol(A_i)$ coincidem.

Então, ambas as funções tentam alcançar o equilibrio entre os agrupamentos, medidos pelos números de vertices, ou pelo peso nas arestas, respectivamente. Veremos que o relaxamento do $NCut$ leva ao agrupamento espectral normalizado, enquanto que do $RCut$ leva ao agrupamento espectral não normalizado. Para tanto, precisamos ter em mente o teorema de Rayleight-Ritz:

Teorema 5.3. *Seja A uma matriz real simétrica de ordem m com autovalores $\lambda_1 \leq \cdots \leq \lambda_m$ e autovetores correspondentes $v_1, \ldots, v_m$. Então:*
$\min\{tr(X^tAX) : X$ *é uma matriz real* $m \times n, X^tX = I_n\} = \lambda_1 + \cdots + \lambda_n$.
A matriz minimizada é a matriz X contendo os n primeiros autovetores de A nas colunas.
$\max\{tr(X^tAX) : X$ *é uma matriz real* $m \times n, X^tX = I_n\} = \lambda_{n+1} + \cdots + \lambda_m$.
A matriz maximizada é a matriz X contendo os $m - n + 1$ últimos autovetores de A nas colunas.

5.1 Aproximação do Corte Proporcional

Nesta seção considere $G = (V, E)$ um grafo com $V = \{v_1, \ldots, v_n\}$ e $A \subset V$. Nosso objetivo é resolver o problema de otimização:

$$\min_{A \subset V} RCut(A, A^c). \tag{5.1}$$

Teorema 5.4 (Aproximação do Corte Proporcional para para $k = 2$)**.** *Para o vetor $f \in \mathbb{R}^n$. O problema em* (5.1) *pode ser reescrito da seguinte forma:*

$$\min_{f \in \mathbb{R}^n} f^tLf \text{ tal que } f \perp 1, \|f\| = \sqrt{n}.$$

Demonstração. Defina $f = (f_1, \ldots, f_n)^t$, com as entradas:

$$f_i = \begin{cases} \sqrt{\frac{|A^c|}{|A|}}, & se \quad i \in A \\ -\sqrt{\frac{|A|}{|A^c|}}, & se \quad i \in A^c. \end{cases} \tag{5.2}$$

Da proposição 3.3 temos que:

$$f^tLf = \frac{1}{2}\sum_{i=1}^{n} w_{ij}(f_i - f_j)^2$$

De (5.2), podemos escrever a igualdade acima da seguinte forma:

$$f^tLf = \frac{1}{2}\Big(\sum_{i\in A, j\in A^c} w_{ij}\Big(\sqrt{\frac{|A^c|}{|A|}} + \sqrt{\frac{|A|}{|A^c|}}\Big)^2 + \sum_{i\in A^c, j\in A} w_{ij}\Big(-\sqrt{\frac{|A^c|}{|A|}} - \sqrt{\frac{|A|}{|A^c|}}\Big)^2\Big).$$

Assim,

$$f^tLf = \frac{1}{2}\Big(\sum_{i\in A, j\in A^c} w_{ij}\Big(\frac{|A^c|}{|A|} + \frac{|A|}{|A^c|} + 2\sqrt{\frac{|A^c||A|}{|A||A^c|}}\Big) + \sum_{i\in A^c, j\in A} w_{ij}\Big(\frac{|A^c|}{|A|} + \frac{|A|}{|A^c|} + 2\sqrt{\frac{|A^c||A|}{|A||A^c|}}\Big)\Big).$$

Da definição de Corte, temos que

$$f^t Lf = cut(A, A^c)\left(\frac{|A^c|}{|A|} + \frac{|A|}{|A^c|} + 2\right).$$

Podemos escrever a igualdade acima da seguinte forma:

$$f^t Lf = cut(A, A^c)\left(\frac{|A^c|}{|A|} + \frac{|A|}{|A^c|} + \frac{|A^c|}{|A^c|} + \frac{|A|}{|A|}\right).$$

Assim,

$$f^t Lf = cut(A, A^c)\left(\frac{|A^c| + |A|}{|A|} + \frac{|A| + |A^c|}{|A^c|}\right).$$

Por definição, temos que $|V| = |A| + |A^c|$, logo,

$$f^t Lf = |V|\left(\frac{cut(A, A^c)}{|A|} + \frac{cut(A, A^c)}{|A^c|}\right).$$

Portanto,

$$f^t Lf = |V| RCut(A, A^c).$$

Além disso, note que

$$\sum_{i=1}^{n} f_i = \sum_{i\in A} \sqrt{\frac{|A^c|}{|A|}} - \sum_{i\in A^c} \sqrt{\frac{|A|}{|A^c|}} = |A|\sqrt{\frac{|A^c|}{|A|}} - |A^c|\sqrt{\frac{|A|}{|A^c|}} = 0.$$

Em outras palavras, o vetor f definido em (5.2) é ortogonal ao vetor constante 1. Além disso,

$$\|f\|^2 = \sum_{i=1}^{n} f_i^2 = \sum_{i\in A} \frac{|A^c|}{|A|} + \sum_{i\in A^c} \frac{|A|}{|A^c|} = |A|\frac{|A^c|}{|A|} + |A^c|\frac{|A|}{|A^c|} = |A| + |A^c| = |V| = n.$$

Consequentemente,

$$\|f\| = \sqrt{n}.$$

Como as entradas do vetor solução, f, só permitem dois valores, o relaxamento óbvio é descartar a condição de valores discretos em f_i e permitir $f_i \in \mathbb{R}$. O que nos leva a um problema de otimização:

$$\min_{f\in\mathbb{R}^n} f^t Lf \text{ tal que } f \perp 1, \|f\| = \sqrt{n}.$$

□

Teorema 5.5 (Aproximação do Corte Proporcional para $k > 2$)**.** *Para $H \in \mathbb{R}^{n\times k}$. O problema em* (5.1) *pode ser reescrito da seguinte forma:*

$$\min_{H\in\mathbb{R}^{n\times k}} Tr(H^t LH), \text{ onde } H^t H = I.$$

Demonstração. O relaxamento do *RCut* do problema de minimização para o caso geral segue de forma similar. Dada uma partição $A_1, \ldots, A_k$ de V, definimos k vetores $h_i = (h_{i,1}, \ldots, h_{i,n})^t$ onde

$$h_{i,j} = \begin{cases} \frac{1}{\sqrt{|A_i|}}, & se \quad j \in A_i \\ 0, & c.c. \end{cases} \tag{5.3}$$

Então tomamos a matriz $H \in \mathbb{R}^{n\times k}$ que contém esses k vetores indicadores na coluna.

Observe que:

- As colunas de H são ortonormais entre si, ou seja, $H^tH = I$.

  De fato, como $A_1, \ldots, A_k$ de V é uma partição de V, temos que para $i \neq j$, $A_i \cap A_j = \emptyset$. Assim para $i \neq j$:

$$(H^tH)_{ij} = \sum_{k=1}^{n} h_{i,k}h_{j,k} = \sum_{k\in A_i\cap A_j} \frac{1}{\sqrt{|A_i||A_j|}} = 0.$$

  Ainda,

$$(H^tH)_{ii} = \sum_{k=1}^{n} (h_{i,j})^2.$$

  Ou seja,

$$(H^tH)_{ii} = \sum_{k\in A_i} (\frac{1}{\sqrt{|A_i|}})^2.$$

  Portanto,

$$(H^tH)_{ii} = |A_i|\frac{1}{|A_i|} = 1.$$

- $h_i^tLh_i = \frac{cut(A_i,A_i^c)}{|A_i|}$.

  Já vimos que,

$$f^tLf = \frac{1}{2}\sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2, \forall f \in \mathbb{R}^n.$$

  Em particular, temos:

$$h_i^tLh_i = \frac{1}{2}\sum_{k,j=1}^{n} w_{kj}(h_{i,k} - h_{i,j})^2.$$

  Logo,

$$h_i^tLh_i = \frac{1}{2}\Bigg(\sum_{k\in A_i, j\in A_i^c} w_{ij}\frac{1}{|A_i|} + \sum_{k\in A_i^c, j\in A_i} w_{kj}\frac{1}{|A_i|}\Bigg).$$

  Ou seja,

$$h_i^tLh_i = \frac{1}{2}\frac{1}{|A_i|}\Bigg(\sum_{k\in A_i, j\in A_i^c} w_{kj} + \sum_{k\in A_i^c, j\in A_i} w_{kj}\Bigg).$$

  Assim, da definição de $cut(A_i, A_i^c)$ temos

$$h_i^tLh_i = \frac{1}{2}\frac{1}{|A_i|}\Bigg(cut(A_i, A_i^c) + cut(A_i^c, A_i)\Bigg).$$

  Consequentemente,

$$h_i^tLh_i = \frac{cut(A_i, A_i^c)}{|A_i|}.$$

- $h_i^t L h_i = (H^t L H)_{ii}$.

  Uma vez que:

$$(H^t L H)_{ii} = \sum_{j=1}^{n} d_j h_{i,j}^2 - \sum_{j,k=1}^{n} w_{jk} h_{i,j} h_{i,k} = h_i^t L h_i.$$

Juntando todas as informações obtidas nos items acima temos:

$$RCut(A_1, \ldots, A_k) = \sum_{i=1}^{n} \frac{cut(A_i, A_i^c)}{|A_i|} = \sum_{i=1}^{n} h_i^t L h_i = \sum_{i=1}^{n} (H^t L H)_{ii} = Tr(H^t L H),$$

onde TrH^tLH denota o traço da matriz H^tLH. Então podemos escrever o problema de minimização do $RCut$ como:

$$\min Tr(H^t L H), \text{ onde } H^t H = I, H \text{ definido em (5.3).}$$

Similarmente ao visto anteriormente, podemos "relaxar" o problema, permitindo que as entradas em H assumam valores reais arbitrários. Então o problema fica:

$$\min_{H \in \mathbb{R}^{n \times k}} Tr(H^t L H), \text{ onde } H^t H = I. \tag{5.4}$$

□

Esta é a forma padrão do problema de minimizar o traço, e uma versão do teorema de $Rayleigh - Ritz$, Teorema (5.3), nos diz que a solução é dada tomando H a matriz que contém os primeiros k autovetores de L nas colunas. Podemos ver que a matriz H é de fato a matriz V usada no algoritmo do agrupamento espectral não normalizado visto anteriormente.

Novamente precisamos reconfigurar os valores reais da matriz solução para uma partição discreta. Como antes, o caminho padrão é usar o algoritmo k-means nas linhas de V. O que nos leva a generalização do algoritmo do agrupamento espectral não normalizado.

5.2 Aproximação NCut

Nesta seção considere $G = (V, E)$ um grafo com $V = \{v_1, \ldots, v_n\}$ e $A \subset V$. Nosso objetivo é resolver o problema de otimização:

$$\min_{A \subset V} NCut(A, A^c). \tag{5.5}$$

Técnicas bem similares às usadas no $RCut$ podem ser usadas para o caso do $NCut$.

Teorema 5.6 (Aproximação NCut para $k = 2$)**.** *Para o vetor $f \in \mathbb{R}^n$, o problema em* (5.5) *pode ser reescrito da seguinte forma:*

$$\min_{f \in \mathbb{R}^n} f^t L f, Df \perp 1, f^t D f = vol(V).$$

Demonstração. Defina o vetor f por:

$$f_i = \begin{cases} \sqrt{\frac{vol(A^c)}{vol(A)}}, & se \quad i \in A \\ -\sqrt{\frac{vol(A)}{vol(A^c)}}, & se \quad i \in A^c. \end{cases} \tag{5.6}$$

Então com cálculos similares podemos ver que:

- $(Df)^t 1 = 0$.

  De fato,

$$(Df)^t 1 = \sum_{i=1}^{n} d_i f_i = \sum_{i \in A} d_i \sqrt{\frac{vol(A^c)}{vol(A)}} - \sum_{i \in A^c} d_i \sqrt{\frac{vol(A)}{vol(A^c)}} =$$

$$vol(A)\sqrt{\frac{vol(A^c)}{vol(A)}} - vol(A^c)\sqrt{\frac{vol(A)}{vol(A^c)}}.$$

  Portanto,

$$(Df)^t 1 = \sqrt{vol(A)vol(A^c) - vol(A^c)vol(A)} = 0.$$

- $f^t Df = vol(V)$.

  Uma vez que,

$$f^t Df = \sum_{i=1}^{n} d_i f_i^2 = \sum_{i \in A} d_i \frac{vol(A^c)}{vol(A)} + \sum_{i \in A^c} d_i \frac{vol(A)}{vol(A^c)}.$$

  Pela definição de volume, visto anteriormente, temos que:

$$f^t Df = vol(A)\frac{vol(A^c)}{vol(A)} + vol(A^c)\frac{vol(A)}{vol(A^c)} = vol(A) + vol(A^c) = vol(V).$$

- $f^t Lf = vol(V)NCut(A, A^c)$

  Já vimos que:

$$f^t Lf = \frac{1}{2}\sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2.$$

  Assim,

$$f^t Lf = \frac{1}{2}\Bigg(\sum_{i \in A, j \in A^c} w_{ij}(\frac{vol(A^c)}{vol(A)} + 2 + \frac{vol(A)}{vol(A^c)}) + \sum_{i \in A, j \in A^c} w_{ij}(\frac{vol(A)}{vol(A^c)} + 2 + \frac{vol(A^c)}{vol(A)}) \Bigg).$$

  Logo,

$$f^t Lf = \frac{1}{2}(\frac{vol(A)}{vol(A^c)} + 2 + \frac{vol(A^c)}{vol(A)})(\sum_{i \in A, j \in A^c} w_{ij} + \sum_{i \in A, j \in A^c} w_{ij}).$$

  Da definição de $cut(A, A^c)$ temos que:

$$f^t Lf = cut(A, A^c)(\frac{vol(A)}{vol(A^c)} + \frac{vol(A^c)}{vol(A^C)} + \frac{vol(A)}{vol(A)} + \frac{vol(A^c)}{vol(A)}).$$

Portanto,

$$f^tLf = cut(A, A^c)(\frac{vol(V)}{vol(A^c)} + \frac{vol(V)}{vol(A)}) = vol(V)(\frac{cut(A, A^c)}{vol(A^c)} + \frac{cut(A, A^c)}{vol(A)}).$$

Da definição de $NCut(A, A^c)$ temos o desejado.

Então podemos reescrever o problema de minimização de $NCut$ para um problema equivalente:

$$\min_A f^tLf \text{ sendo } f \text{ como em (5.6)}, Df \perp 1, f^tDf = vol(V).$$

Novamente, "relaxamos" o problema permitindo f assumir valores reais:

$$\min_{f\in\mathbb{R}^n} f^tLf, Df \perp 1, f^tDf = vol(V).$$

□

Agora substituindo $g = D^{\frac{1}{2}}f$, temos:

- $f^tLf = g^tD^{-\frac{1}{2}}LD^{-\frac{1}{2}}g$;
- $g \perp D^{\frac{1}{2}}1$;
- $vol(V) = f^tDf = (D^{-\frac{1}{2}}g)^tD^{-\frac{1}{2}}g = g^tg = \|g\|^2$.

Assim, o problema é:

$$\min_{g\in\mathbb{R}^n} g^tD^{-\frac{1}{2}}LD^{-\frac{1}{2}}g, \text{ onde } g \perp D^{\frac{1}{2}}1 \text{ e } \|g\|^2 = vol(V). \tag{5.7}$$

Observe que, $D^{\frac{1}{2}}1$ é o primeiro autovetor de L_{sym}, e $vol(V)$ é constante. Portanto o problema em ((5.7)) está na forma padrão do teorema de $Rayleigh-Ritz$, e a solução g é dada pelo segundo autovetor de L_{sym}. Substituindo, novamente, $f = D^{-\frac{1}{2}}g$ podemos ver que f é o segundo autovetor de L_{rw}, ou equivalentemente, o autovetor generalizado de $Lv = \lambda Dv$.

Teorema 5.7 (Aproximação NCut para $k > 2$)**.** *Para $U \in \mathbb{R}^{n\times k}$ o problema em* (5.5) *pode ser escrito da seguinte forma:*

$$\min_{U\in\mathbb{R}^{n\times k}} Tr(U^tD^{-\frac{1}{2}}LD^{-\frac{1}{2}}U) \text{ tal que } U^tU = I.$$

Demonstração. Definimos os vetores indicadores $h_i = (h_{i,1}, \ldots, h_{i,n})^t$ por:

$$h_{i,j} = \begin{cases} \frac{1}{\sqrt{vol(A)}}, & se \quad j \in A_i \\ 0, & se \quad j \in A_i^c. \end{cases} \tag{5.8}$$

Tomando a matriz H como sendo a matriz contendo esses k vetores indicadores na coluna, podemos ver que:

- $H^tDH = I$.

  Observe que,

$$(H^tDH)_{ij} = \sum_{k=1}^{n} d_k h_{i,k} h_{j,k}.$$

  Assim,

$$(H^tDH)_{ii} = \sum_{k=1}^{n} d_k h_{i,k}^2 = \frac{1}{vol(A_i)} \sum_{k \in A_i} d_k = 1$$

  Para $i \neq j$, temos que $(H^tDH)_{ij} = 0$.

- $h_i^t D h_i = 1$.

  Note que:

$$h_i^t D h_i = \sum_{k=1}^{n} d_k h_{i,k}^2 = \sum_{k \in A_i} \frac{d_k}{vol(A_i)} = \frac{1}{vol(A_i)} \sum_{k \in A_i} d_k = 1.$$

- $h_i^t L h_i = \frac{cut(A_i, A_i^c)}{vol(A_i)}$.

  Já vimos que:

$$h_i^t L h_i = \frac{1}{2} \sum_{i=1}^{n} w_{jk} (h_{i,j} - h_{i,k})^2.$$

  Assim,

$$h_i^t L h_i = \frac{1}{2} \Bigg(\sum_{j \in A_i^c, k \in A_i} w_{jk} \frac{1}{vol(A_i)} + \sum_{j \in A_i, k \in A_i^c} w_{jk} \frac{1}{vol(A_i)} \Bigg).$$

  Logo,

$$h_i^t L h_i = \frac{cut(A_i, A_i^c)}{vol(A_i)}.$$

Então podemos escrever o problema de minimização de $NCut$ como segue:

$$\min_{A_i, \ldots, A_k} Tr(H^t L H), \ \textit{onde} \ H^t D H = I \ \textit{e} \ H \ \textit{como definido em} \ (5.8).$$

"Relaxando" a condição de descrição e substituindo $U = D^{\frac{1}{2}} H$ temos:

- $H^t L H = U^t D^{-\frac{1}{2}} L D^{-\frac{1}{2}} U$;
- $I = H^t D H = U^t D^{-\frac{1}{2}} D D^{-\frac{1}{2}} U = U^t U$.

E obtemos o problema:

$$\min_{U \in \mathbb{R}^{n \times k}} Tr(U^t D^{-\frac{1}{2}} L D^{-\frac{1}{2}} U) \text{ tal que } U^t U = I.$$

□

Novamente este é um problema padrão de minimização do traço que tem como solução a matriz U que contém os primeiros k autovetores de L_{sym} nas colunas. Substituindo, novamente, $H = D^{-\frac{1}{2}} U$ podemos ver que a solução H consiste dos k primeiros autovetores da matriz L_{rw}, ou os k primeiros autovetores generalizados de $Lv = \lambda Dv$. Isto produz algoritmo do agrupamento espectral não normalizado de acordo com Shi e Malik (2000).

5.3 Observações

Há varios comentários que devemos fazer sobre essa obtenção do agrupamento espectral. O mais importante é que não há garantia de que a solução do problema "relaxado" comparado com a solução exata seja boa. Isto é, se $A_1, \ldots, A_k$ é a solução exata da minimização do $RCut$ e $B_1, \ldots, B_k$ é a solução construída pelo agrupamento espectral não normalizado então $RCut(A_1, \ldots, A_k) - RCut(B_1, \ldots, B_k)$ pode ser arbitrariamente grande.

O exemplo a seguir para o caso $k = 2$ pode ser encontrado em [7].

Exemplo 5.1. O autor considera uma classe muito simples de grafos chamada "grafos baratas", como na figura abaixo, onde cada aresta tem peso 1, estes grafos parecem essencialmente com uma escada com alguns "degraus" removidos.

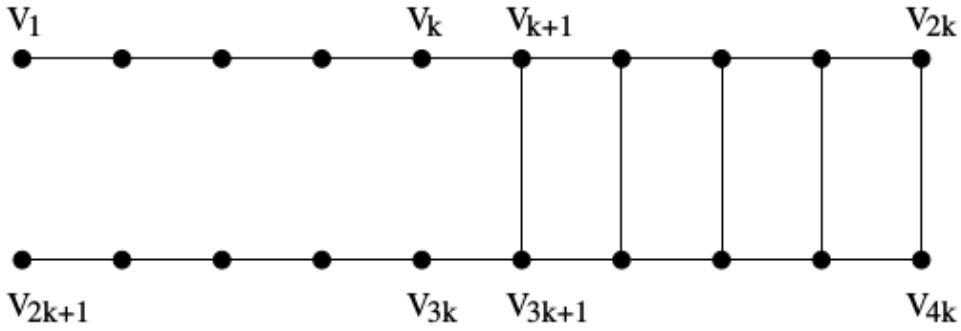

Figura 5.1: O grafo "barata" de Guattery and Miller, [7].

Obviamente, o $RCut$ ideal apenas corta a escada por um corte vertical tal que $A = \{v_1, \ldots, v_k, v_{2k+1}, \ldots, v_{3k}\}$ e $A^c = \{v_{k+1}, \ldots, v_{2k}, v_{3k+1}, \ldots, v_{4k}\}$. Este corte é perfeitamente equilibrado, com $|A| = |A^c| = 2k$ e

$$cut(A, A^c) = \sum_{i \in A, j \in A^c} w_{ij} = w_{kk+1} + w_{3k3k+1} = 2$$

Entretanto, estudando as propriedades do segundo autovetor do grafo Laplaciano não normalizado do grafo barata, os autores provam que agrupamento espectral não normalizado sempre corta a escada horizontalmente, construindo os conjuntos $B = \{v_1, \ldots, v_{2k}\}$ e $B^c = \{v_{2k+1,\ldots,v_{4k}}\}$. Isto também resulta em um corte equilibrado:

$$cut(B, B^c) = \sum_{i \in B, j \in B^c} w_{ij} = w_{k+1k+3} + \cdots + w_{2k4k} = k.$$

Então

$$RCut(A, A^c) = \frac{cut(A, A^c)}{|A|} + \frac{cut(A^c, A)}{|A^c|} = \frac{2}{k},$$

enquanto que

$$RCut(B, B^c) = \frac{cut(B, B^c)}{|B|} + \frac{cut(B^c, B)}{|B^c|} = 1.$$

Isto significa que o valor do $RCut$ do corte obtido pelo agrupamento espectral é $\frac{k}{2}$ vezes pior que o corte ideal.

O mesmo exemplo também funciona para $NCut$. Em geral sabe-se que algoritmos eficientes que aproximam um corte equilibrado do grafo em um fator constante não existe. Pelo contrário, este problema de aproximação é um NP hard.[3].

Claramente, o relaxamento que discutimos acima não é o único. Por exemplo, um relaxamento completamente diferente é mostrado em [2], e tem outros relaxamentos úteis. A razão do porque o relaxamento do agrupamento espectral é tão atraente, embora muitas vezes não leve a uma solução particularmente boa, é o fato que isto resulta em um problema de álgebra linear padrão fácil de resolver.

Finalmente, não há nenhum princípio sobre o uso do algoritmo k-means para construir partições discretas a partir do vetores y_i. Qualquer outro algoritmo que pode resolver este problema pode ser usado, e várias outras técnicas são regularmente usadas.

Um exemplo pode ser encontrado em [11], onde ele tenta construir uma partição separando os pontos y_i por hiperplanos em $\mathbb{R}^k$. Entretanto pode-se argumentar que pelo menos a distância euclidiana entre os pontos y_i é uma quantidade significativa para se olhar.

Na próxima seção veremos que a distância euclidiana entre pontos y_i está relacionada com a "distância comutativa" no grafo. Em [17], os autores mostram que a distância euclidiana entre y_i também está relacionada a uma distância mais geral, conhecida com "diffusion distance".

6 Caminhos Aleatórios

Outra linha de argumento que explica o agrupamento espectral é baseado em caminhos aleatórios no grafo.

Definição 6.1. *Um caminho aleátorio no grafo é um processo estocástico que salta aleatoriamente de vértice em vértice.*

Iremos ver a seguir que agrupamento espectral pode ser interpretado como tentar encontrar uma partição do grafo tal que o caminho aleátorio permaneça dentro do mesmo agrupamento e raramente salta entre agrupamentos. Intuitivamente isto faz sentindo, ainda mais quando juntamos com a explicação do corte do grafo. Uma partição com um corte "baixo" também terá a propriedade que os caminhos aleatórios não têm muitas oportunidades de saltar entre agrupamentos.

Definição 6.2. *Formalmente, a probabilidade de transição de pular num passo do vértice i pro vértice j é proporcional ao peso da aresta, w_{ij}, e é dada por:*

$$p_{ij} = \frac{w_{ij}}{d_i}.$$

A matriz transição $P = (p_{ij})_{ij=1,\dots,n}$ do caminho aleátorio é, então, definida por:

$$P = D^{-1}W.$$

Definição 6.3. *Um grafo G é bipartido se existe uma bipartição $\{U, W\}$ de V tal que toda aresta de G tem uma ponta U e a outra ponta em W, onde uma bipartição de V é um par, $\{U, W\}$ de conjuntos não vaziostal que $U \cup W = V$ e $U \cap W = \emptyset$.*

Definição 6.4. *Um processo estocástico é estacionário se para qualquer $i_1, \dots, i_n$ e qualquer m a distribuição conjunta de $(X_{i_1}, \dots, X_{i_n})$ é a mesma que a a distribuição conjunta de $(X_{i_1+m}, \dots, X_{i_n+m})$.*

Se o grafo é conexo e não bipartido, então o caminho aleátorio sempre possui uma única distribuição estacionária. $\Pi = (\pi_1, \dots, \pi_n)^t$, onde $\pi_i = \frac{d_i}{vol(G)}$.

Há uma pequena relação entre L_{rw} e P, uma vez que podemos escrever $L_{rw} = I - P$, tem-se que λ é um autovalor de L_{rw} com autovetor v se, e só se, $1 - \lambda$ é um autovalor de P com autovetor v. Assim, muitas propriedades dos grafos podem se expressar em termos de P ver [12]. Então do ponto de vista de caminhos aleatórios não é uma surpresa que os maiores autovetores de P e os menores autovetores de L_{rw} possam ser usados para descrever propriedades de agrupamentos no grafo.

6.1 Relação entre Caminhos Aleatórios e NCut

Uma equivalência formal entre $NCut$ e probabilidade de transição de caminho aleatorio pode ser encontrado em [14].

Proposição 6.5. *Seja G conexo e não bi-particionado. Assuma que percorremos um caminho aleatorio $(X_t)_{t\geq 0}$ começando em X_0 na distribuição estacionária Π. Para conjuntos disjuntos $A, B \subset V$, denote por $P(A|B) = P(X_1 \in B / X_0 \in A)$. Então,*

$$NCut(A, A^c) = P(A^c|A) + P(A|A^c).$$

Demonstração. Primeiramente observe que:

$$P(X_0 \in A, X_1 \in B) = \sum_{i\in A, j\in B} P(X_0 = i, X_1 = j) = \sum_{i\in A, j\in B} \Pi_i p_{ij} = \frac{1}{vol(G)} \sum_{i\in A, j\in B} w_{ij}.$$

Usando isto, obtemos que:

$$P(B|A) = \frac{P(X_0 \in A, X_1 \in B)}{P(X_0 \in A)} = \frac{1}{vol(G)} sum_{i\in A, j\in B} w_{ij} \left(\frac{vol(A)}{vol(G)}\right)^{-1} = \frac{cut(A, B}{vol(A)}.$$

Assim,

$$P(A|A^c) = \frac{cut(A, A^c)}{vol(A^c)}$$

e

$$P(A^c|A) = \frac{cut(A, A^c)}{vol(A)}$$

□

A proposição nos dá uma interpretação do $NCut$, e portanto do agrupamento espectral normalizado. Nos diz que quando minimizamos o $NCut$, na verdade olhamos para um corte no grafo tal que um caminho aleatorio raramente transita de A para A^c ou vice-versa.

6.2 Distância Comutativa

Uma segunda conexão entre caminho aleatórios e grafos Laplacianos pode ser feita atráves da distância comutativa.

Definição 6.6. *A distância comutativa (também chamada de distância de resistência), $c(i, j)$, entre dois vértices i e j é o tempo esperado que o caminho aleatorio leva para viajar do vértice i para o vértice j e voltar.*

A distância comutativa tem várias propriedades que a tornam particularmente atraente para o aprendizado. Ao contrário da distância do menor caminho no grafo, a distância comutativa entre dois vértices decresce se existe muitos jeitos curtos de ir do vértice i para o vértice j. Então, ao invés, de olhar para o menor caminho, a distância comutativa olha para um conjunto de caminhos curtos. Pontos que estão conectados por um caminho curto e que estão no mesmo agrupamento do grafo estão muito mais próximos entre si do que pontos que estão conectados por um caminho curto mas estão em agrupamentos diferentes. Nota-se que a distância comutativa no grafo pode ser

calculada com a ajuda da inversa generalizafa (também conhecida por pseudo inversa), $L^\dagger$, do grafo Laplaciano L. Para definir a inversa generalizada de L, lembramos que a matriz L pode ser decomposta como $L = V\Lambda V^t$, **ver apêndice A**, onde V é a matriz contendo os autovetores nas colunas e Λ é a matriz diagonal com os autovalores $\lambda_1, \ldots, \lambda_n$ na diagonal. Como pelo menos um dos autovalores de L é zero, a matriz L não é inversível. Definimos, então, a inversa generalizada:

Definição 6.7. *$L^\dagger = V\Lambda^\dagger V^t$, onde a matriz diagonal $\Lambda^\dagger$ tem os elementos $\frac{1}{\lambda_i}$ se $\lambda_i \neq 0$ e 0 se $\lambda_i = 0$, na diagonal.*

As entradas de $L^\dagger$ podem ser calculadas como:

$$l^\dagger_{ij} = \sum_{k=2}^{n} \frac{1}{\lambda_k} v_{ik} v_{jk}.$$

Além disso, como todos os autovalores são não negativos, $L^\dagger$ é semi-definida positiva. Para mais propriedades de $L^\dagger$ ver [8] ou Apêndice A.

Proposição 6.8. *Seja G um grafo, conexo, não orientado. Denote por c_{ij} a distância comutativa entre o vértice i e o vértice j, e $L^\dagger = (l^\dagger_{ij})_{i,j=1,\ldots,n}$ a inversa generalizada de L. Então, temos:*

$$c_{ij} = vol(G)(l^\dagger_{ii} - 2l^\dagger_{ij} + l^\dagger_{jj}) = vol(G)(e_i - e_j)^t L^\dagger (e_i - e_j).$$

Este resultado foi publicado em [10], onde ele foi provado usando a teoria de métodos de teoria da rede elétrica. Para uma prova usando análise para caminhos aleatórios ver [5]. Como ambas provas são técnicas serão omitidas.

Existem outros modos de expressar a distância comutativa com a ajuda dos grafos Laplacianos. Por exemplo, um método em termos de autovetores da Laplaciana normalizada, L_{sym}, pode ser encontrada no *corolário 3.2* de [12] e um método de calcular a distância comutativa com a ajuda de determinantes de certas sub-matrizes de L pode ser encontrada em [8].

A proposição acima tem uma consequência importante, pois mostra que $\sqrt{c_{ij}}$ pode ser considerado como uma função de distância euclidiana nos vértices do grafo. O que significa que podemos construir um mergulho que leva os vértices v_i do grafo em pontos $z_i \in \mathbb{R}^n$, tal que a distância euclidiana entre os pontos z_i coincide com a distância comutativa no grafo, ie,$\|z_i - z_j\|^2 = c_{ij}$.

7 Teoria da Perturbação

Já vimos que se o grafo consiste de k componentes conexas, então a multiplicidade do autovetor 0 de ambas, L e L_{rw}, é k e o autoespaço é gerado pelo vetor indicador das componentes conexas. O argumento de perturbação diz que a similaridade entre os agrupamentos é exatamente 0. Se tivermos uma situação onde a similaridade entre os agrupamentos são bem pequenas, então os autovetores dos primeiros k autovalores devem estar bem próximos aos do caso ideal. Assim ainda podemos recuperar o agrupamento a partir desses autovetores.

A teoria da perturbação estuda a questão de como autovalores e autovetores de uma matriz A mudam se adicionarmos uma pequena perturbação H, isto é, consideramos a matriz pertubada $\overline{A} = A + H$. A maioria dos teoremas de perturbação afirma que a distância entre os autovalores e autovetores de A e $\overline{A}$ é limitada por uma constante vezes a norma de H. A constante normalmente depende de qual autovalor estamos olhando, e quanto este autovalor está longe do resto do espectro.

Considere primeiramente o caso ideal onde a similaridade entre agrupamentos é 0. Aqui, os primeiros k autovetores de L e L_{rw} são os vetores indicadores dos agrupamentos. Neste caso, os pontos $y_i \in \mathbb{R}^n$ construido no algoritmo do agrupamento espectral tem a forma $(0, \ldots, 0, 1, 0, \ldots, 0)^t$ onde a posição do 1 indica a componente conexa onde este ponto está. Em particular, todos y_i que estão na mesma componente conexa coincidem. O algoritmo k-means irá encontrar, trivialmente, a partição correta colocando um ponto central em cada um dos pontos $(0, \ldots, 0, 1, 0, \ldots, 0)^t \in \mathbb{R}^n$.

No caso quase ideal, onde ainda temos agrupamentos distintos mas a similaridade entre eles não é exatamente 0, consideramos as matrizes Laplacianas como sendo versões pertubadas da matriz do caso ideal. Então a teoria da perturbação nos diz que os autovetores estarão bem próximos aos vetores indicadores ideais. Os pontos y_i podem não coincidir com $(0, \ldots, 0, 1, 0, \ldots, 0)^t$, mas diferem dele por um pequeno erro. Portanto, se a perturbação não for muito grande, então o algoritmo k-means ainda irá separar os grupos entre si.

Formalmente, esses resultados são baseado no teorema de Davis-Kahan da teoria de matriz perturbada, o qual limita a diferença entre autoespaços de matrizes simétricas sob perturbações. Para maiores detalhes, veja seção VII de [1].

Na teoria da perturbação, as distâncias entre subespaços são usualmente medidas usando ângulos canônicos (também conhecidos ângulos principais).

Definição 7.1. *Sejam $\mathbb{V}_1$ e $\mathbb{V}_2$ dois subespaços p-dimensionais de $\mathbb{R}^d$ e V_1, V_2 duas matrizes tais que suas colunas formam sistemas ortonormais para $\mathbb{V}_1, \mathbb{V}_2$, respectivamente. Então o cosseno, $cos\theta_i$, dos ângulos canônicos θ_i entre $\mathbb{V}_1$ e $\mathbb{V}_2$ são os valores singulares de $V_1^t V_2$. Para $p = 1$, a definição de ângulo canônico coincide com a definição de ângulo normal. A matriz $sen\theta(\mathbb{V}_1, \mathbb{V}_2)$ denotará a matriz diagonal com os*

ângulos canônicos na diagonal.

Teorema 7.2 (Davis-Kahan). *Sejam $A, H \in \mathbb{R}^{n\times n}$ matrizes simétricas, e seja $\|\cdot\|$ a norma de Frobenius. Considere $\overline{A} = A + H$ a versão perturbada de A. Seja $S_1 \subset \mathbb{R}$ um intervalo. Denote por $\sigma S_1(A)$ o conjunto de autovalores de A que está contido em S_1, e por V_1 o autoespaço correspondente a todos esses autovalores. Denote por $\sigma S_1(\overline{A})$ e $\overline{V_1}$ os análogos para $\overline{A}$. Defina a distância entre S_1 e o espectro de A fora de S_1 como*

$$\delta = \min\{|\lambda - s|; \lambda \text{ é autovalor de } A, \lambda \notin S_1, s \in S_1\}.$$

Então a distância, $d(V_1, \overline{V_1}) = \|sen\theta(V_1, \overline{V_1})\|$, entre os espaços V_1 e $\overline{V_1}$ é limitada por

$$d(V_1, \overline{V_1}) \leq \frac{\|H\|}{\delta}.$$

Vamos tentar entender esse teorema, para simplificar no caso da Laplaciana não normalizada (o caso da normalizada é análogo). A matriz A corresponderá ao grafo Laplaciano L no caso ideal, ou seja, assumiremos que o grafo tem k componentes conexas. A matriz $\overline{A}$ corresponde ao caso perturbado, onde devido aos ruídos as k componentes conexas do grafo já não estão completamente desconexas, mas estão conectadas por poucas arestas com peso baixo.

Denotaremos o Laplaciano correspondente por $\overline{L}$, os autovalores de L por $\lambda_1, \ldots, \lambda_n$ e os autovalores de $\overline{L}$ por $\overline{\lambda_1}, \ldots, \overline{\lambda_n}$. Escolher o intervalo S_1, é o ponto crucial, queremos escolher S_1 de modo que ambos os primeiros k autovalores de L e os primeiros k autovalores de $\overline{L}$ estão em S_1. Isto é mais fácil, quanto menor for a perturbação $H = L - \overline{L}$ maior é o "eigengap" (eigengap é a lacuna entre dois autovalores consecutivos), $|\lambda_k - \lambda_{k+1}|$.

Se conseguirmos encontrar tal conjunto, então o teorema Davis-Kahan nos diz que os autoespaços correspondente ao primeiros k autovalores do caso ideal L e os primeiros k autovalores do caso perturbado $\overline{L}$ estão bem próximos entre si e que sua distância é limitada por $\frac{\|H\|}{\delta}$. Então, como os autovetores no caso ideal são seccionalmente constante nas componentes conexas, isto será, aproximadamente, verdadeiro para o caso perturbado. Quão bom é "aproximadamente" depende da norma da perturbação, $\|H\|$ e a distância δ, entre S_1 e o $k+1$ autovetor de L.

No caso em que o conjunto S_1 é escolhido como o intervalo $[0, \lambda_k]$, δ coincide com o intervalo espectral $|\lambda_{k+1} - \lambda_k|$. Podemos ver do teorema que quanto maior este eigengap é, mais próximos são os autovetores do caso ideal e do caso perturbado, e portanto o melhor agrupamento espectral funciona.

Se a perturbação H é muito grande ou o eigengap é muito pequeno, podemos não encontrar o conjunto S_1 que contém os k primeiros autovalores de L e os k primeiros autovalores de $\overline{L}$. Neste caso, é preciso escolher um conjunto que contém os k primeiros autovalores de L, mas talvez um pouco a mais (ou um pouco a menos) de autovalores de $\overline{L}$. Então o resultado do teorema se torna mais fraco.

7.1 Comentários sobre a abordagem de perturbação

Um pouco de cautela é necessária quando usamos argumentos da teoria da perturbação para justificar algoritmos do agrupamento baseado em autovetores de matrizes. Em geral, qualquer bloco simétrico, (matriz diagonal simétrica) tem a propriedade de que existe uma base de autovetores que são zero fora dos blocos individuais e dentro dos blocos assume valor real. Baseado neste argumento, vários autores usam os autovetores da matriz similaridade, S, ou da matriz adjacência ponderada, W, para encontrar agrupamentos. No entanto, ser bloco diagonal no caso ideal de agrupamentos completamente separados pode ser considerada como uma condição necessária para a utilização, bem sucedida, dos autovetores, mas não suficiente. Ao menos mais duas propriedades dever ser satisfeitas:

Primeiro, precisamos ter certeza que a ordem dos autovalores e autovetores é significativa. No caso da Laplaciana isto é sempre verdade, uma vez que qualquer componente conexa possui exatamente um autovetor que tem valor próprio 0. Assim, se o grafo tem k componente conexas e tomamos os k primeiros autovetores próprios do Laplaciano, então sabemos que temos exatamente um autovetor por componente. No entando isto pode não ser verdade para outras matrizes, tais como, S ou W. Por exemplo, poderia ser o caso em que os dois maiores autovalores da matriz bloco diagonal de similaridade, S, vem do mesmo bloco. Em tal situação, se tomarmos os k primeiros autovetores de S, alguns blocos serão representados várias vezes, enquanto os outros blocos iremos perder completamente (se não tomarmos certas precauções). Esta é a razão pela qual o uso os autovetores de S ou W para agrupamento deve ser desencorajado.

A segunda propriedade é que no caso ideal, as entradas dos autovetores nas componentes deve ser "limitado, com segurança, para longe" de 0. Assuma que um autovetor na primeira componente conexa tem uma entrada $v_{1,i} = \varepsilon$ na posição i. No caso ideal, o fato dessa entrada ser diferente de zero indica que o ponto correspondente i pertence ao primeiro agrupamento, então o caso ideal deveria ser o caso em que $v_{1,j} = 0, j \neq i$. Agora considere a mesma situação, mas com dados perturbardos. O autovetor perturbado $\overline{v}$ normalmente não terá mais qualquer componente diferente de zero, mas se o ruído não for muito grande, então a teoria da perturbação nos diz que as entradas $\overline{v_{1,i}}$ e $\overline{v_{1,j}}$ irão assumir valores pequenos, digamos ε_1 e ε_2. Na prática, não é claro como deveríamos interpretar esta situação, ou acreditamos que entradas pequenas em $\overline{v}$ indica que os pontos não pertencem ao primeiro agrupamento (o que classifica incorretamente o primeiro ponto de dado i), ou pensamos que as entradas já indicam a que classe pertence e classificamos ambos os pontos no primeiro agrupamento(o que classifica incorretamente o ponto j).

Para ambas matrizes, L e L_{rw}, os autovetores na situação ideal são vetores indicadores, então o problema descrito acima não pode ocorrer. Entretanto, isto não é verdade para matriz L_{sym}, que é usado no algoritmo do agrupamento espectral normalizado em [18]. Mesmo no caso ideal, os autovetores da matriz são dados por $D^{\frac{1}{2}}1_{A_i}$. Se o grau dos vértices diferem muito, e em particular, se existem vértices que possuem grau muito baixo, as entradas correspondentes nos autovetores são bem pequenas. Para contrariar o problema descrito acima, aparece o passo da linha normalizada no algoritmo de Ng et al. (2002). No caso ideal, a matriz V no algoritmo tem exatamente uma entrada diferente de zero por linha. Depois da normalização da linha, a matriz U no algoritmo de Ng et al. (2002) consiste, portanto, dos vetores indicadores dos agrupamentos. Entretanto, note que, diferentes coisas podem acontecer. Assuma que temos, $\overline{v_{1,i}} = \varepsilon_1$ e

$\overline{v_{2,i}} = \varepsilon_2$, valores pequenos. Se normalizarmos a i-ésima linha de V, ambos, ε_1 e ε_2 serão multiplicados por um fator de $\frac{1}{\sqrt{\varepsilon_1^2+\varepsilon_2^2}}$ e se tornam bastante grande. Agora estamos num problema similar ao descrito anteriormente: ambos os pontos, provavelmente, serão classificados no mesmo agrupamento, mesmo que pertençam a agrupamentos diferentes. Este argumento mostra que agrupamento espectral usando a matriz L_{sym} pode ser problemático se os autovetores tiver entradas particularmente pequenas. No entanto, observe que tais entradas pequenas no autovetor apenas ocorrem se algum dos vértices tiver um grau pequeno (uma vez que os autovetores de L_{sym} são dados por $D^{\frac{1}{2}}1_{A_i}$). Podemos concordar que em tal caso, o conjunto de dados deve ser considerado um "outlier" de qualquer modo, e então não importará em qual agrupamento o ponto deve estar. (Em estatística, outlier, valor aberrante ou valor atípico, é uma observação que apresenta um grande afastamento das demais da série, ou que é inconsistente.)

Para resumir, a conclusão é que ambos, agrupamento espectral não normalizado e o normalizado com L_{rw} são justificados pela teoria de aproximação da perturbação. O agrupamento espectral normalizado com L_{sym} também pode ser justificado pela teoria da perturbação, mas deve ser tratado com mais cuidado se o grafo conter vértices com grau muito baixo.

8 Preparando os Detalhes Práticos

Nesta seção iremos discutir brevemente algumas questões que surgem quando realmente "usamos" agrupamento espectral.

Há varias escolhas a serem feitas e parâmetros a serem definidos. Entretanto, a breve discussão nesta seção destina-se principalmente a sensibilizar para os problemas gerais que podem ocorrer.

8.1 Construindo a Similaridade de Grafos

Escolher a similaridade de grafos e seus parâmetros para agrupamento espectral não é uma tarefa trivial. O problema já começa com a escolha da função similaridade,s_{ij}, em si. Em geral, deve-se tentar assegurar que as vizinhanças locais induzidas por esta função similaridade são significativas, mas em particular em configuração de agrupamento isto é algo muito difícil de se avaliar.

Em última análise, a escolha da função de similaridade depende do domínio de onde provêm os dados, e nenhuma regra geral pode ser dada.

A segunda escolha refere-se à construção de similaridades de grafos, ou seja, que tipo de grafo escolhemos e como definimos o parâmetro que controla a sua conectividade (por exemplo, o parâmetro ε do grafo ε-vizinhança ou o parâmetro k do grafo k-vizinhos mais próximos). Para ilustrar o comportamento de diferentes grafos usamos o exemplo apresentado em [23].

Exemplo 8.1. Como distribuição básica escolhemos uma distribuição em $\mathbb{R}^2$ com três agrupamentos, sendo, duas "luas" e um Gaussiano. A densidade da lua inferior foi escolhida para ser maior do que a lua superior.

A primeira figura mostra uma amostra desta distribuição.

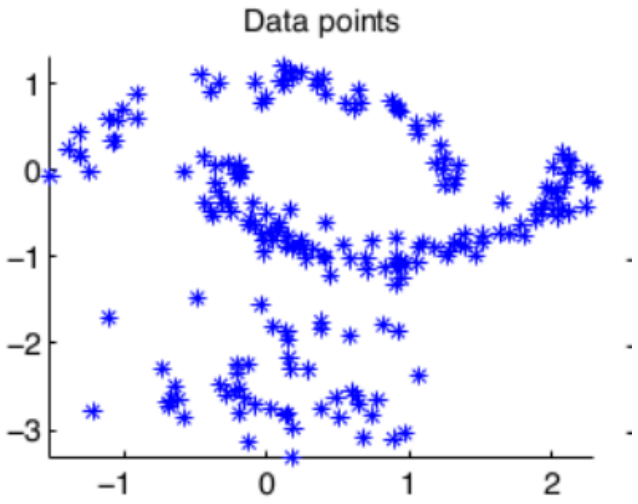

Figura 8.1: Conjunto de pontos de Von Ulrike, [23].

As outras três mostram as diferentes similaridades de grafos nesta amostra.

No grafo da ε, podemos ver que é dificil escolher um parâmetro, ε, útil. Com $\varepsilon = 0.3$ como na figura, os pontos no meio da lua já são fortemente conectados, enquanto os pontos na Gaussiana são "apenas" conectados. Este problema sempre ocorre se tivermos dados "em escalas diferentes", ou seja, as distâcias entre pontos de dados são diferentes em diferentes regiões do espaço.

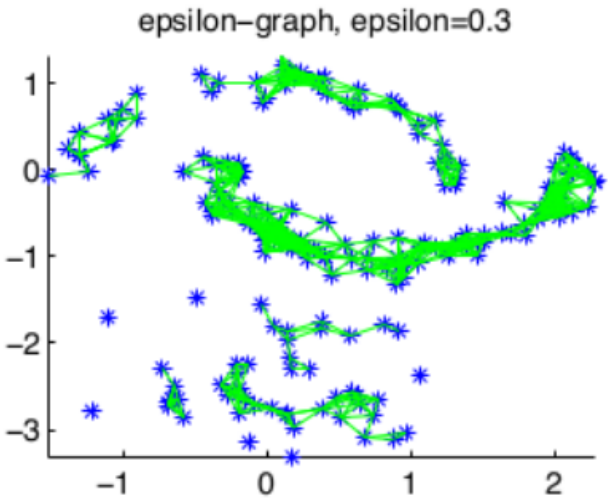

Figura 8.2: ε-vizinhança de Von Ulrike, [23].

Por outro lado, o grafo dos k-vizinhos mais próximos pode conectar pontos "em diferentes escalas". Podemos ver que pontos de Gaussiana com pouca densidade estão conectados com pontos de maior densidade da lua. Esta é uma propriedade geral do grafo dos k-vizinhos mais próximos que pode ser muito útil. Também podemos ver que tal grafo pode cair em várias componentes conexas, se houver regiões de alta densidade que estão razoavelmente longe de outras. Este é o caso das duas luas neste exemplo.

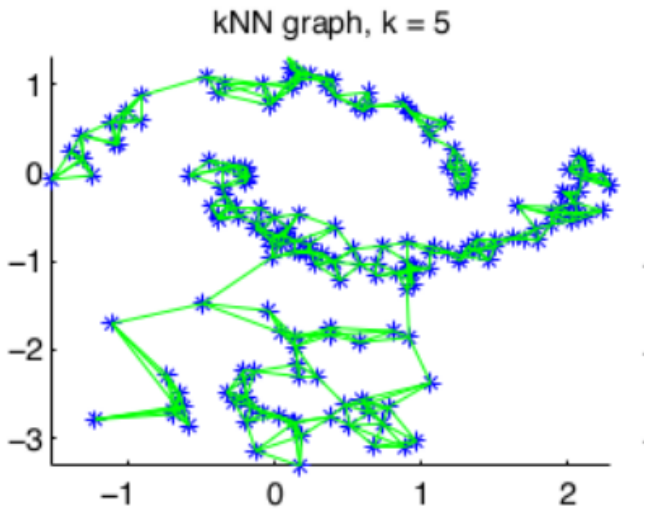

Figura 8.3: k-vizinhos mais próximos de Von Ulrike, [23].

Podemos ver na última figura que o grafo dos k-vizinhos mutualmente mais próximos tem a propriedade de conectar pontos dentro de regiões de densidade constante, mas não conecta regiões de densidades diferentes entre si.

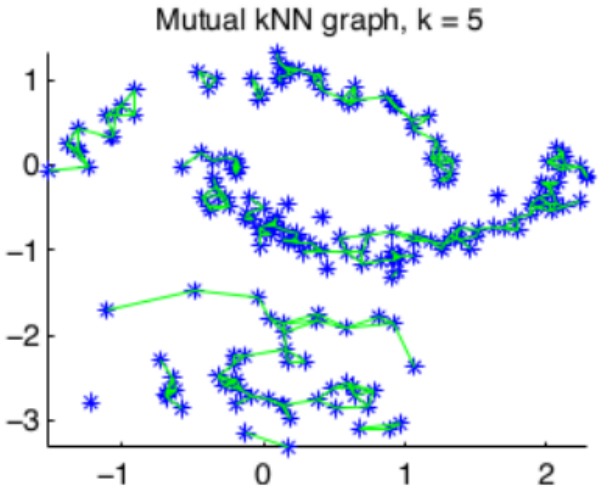

Figura 8.4: k-vizinhos mutualmente mais próximos de Von Ulrike, [23].

Assim o grafo dos k-vizinhos mutualmente mais próximos pode ser considerado como estando entre o grafo ε-vizinhança e o grafo dos k-vizinhos mutualmente mais próximos, pois é capaz de agir em diferentes escalas, mas não mistura essas escalas umas com as outras. Portanto o grafo dos k-vizinhos mais próximos mútuos parece particularmente adequado se quisermos detectar agrupamentos de diferentes densidades.

Em geral, agrupamento espectral pode ser bastante sensível à mudanças no grafo de similaridade e a escolha do seus parâmetros.

8.2 Calculando os Autovetores

Para implementar o agrupamento espectral em prática é necessário calcular os k primeiros autovalores de uma matriz Laplaciana potencialmente grande. Felizmente, se usarmos o grafo dos k-vizinhos mais próximos ou o grafo ε-vizinhança então todas as matrizes do grafo Laplaciano são escassas, uma vez que as entradas de L que são diferentes de zero é finita. Existem métodos eficientes para computar os primeiros autovetores de matrizes esparças, os mais populares sendo o método do poder ou métodos de subespaços de Krylov tais como o método de Lanczos(ver [6]). A velocidade de convergência desses algoritmos depende do tamanho do eigengap quanto maior for este eigengap, mais rápido os algoritmos que calculam os primeiros k autovetores convergem.

Note que um problema geral ocorre se um dos autovetores em consideração tem multiplicidade maior do que um. Por exemplo, na situação ideal de k agrupamentos desconexos, o autovalor 0 tem multiplicidade k. Como nós já vimos, neste caso, o autoespaço é estendido pelos k vetores indicadores de cada agrupamento. Mas, infelizmente, os vetores calculados pelos algoritmos númericos não necessariamente convergem para alguma base ortonormal do autoespaço, e geralmente depende de detalhes de implementação a que base exatamente converge o algoritmo. Mas isso não é tão ruim. Observe que todos os vetores no espaço estendido pelo vetor indicador do agrupamento, 1_{A_i}, tem a forma

$$v = \sum_{i=1}^{k} a_i 1_{A_i},$$

para alguns coeficientes a_i, ou seja, eles são constantes por partes nos agrupamentos. Assim, os vetores retornados por esse algoritmo ainda codificam as informações sobre os agrupamentos, que podem então, ser usados pelo algoritmo k-means para reconstruir os agrupamentos.

8.3 O Número de agrupamentos

Escolher o número k de agrupamentos é um problema geral para todos os algoritmos de agrupamento e uma variedade de métodos mais ou menos bem sucedidas foram concebidas para este problema, por exemplo, a estatística de intervalos em "gap", ou abordagens de estabilidade. Uma ferramenta que é articulada particularmente para agrupamento espectral é a heurística do eigengap, que pode ser usado para todos os três grafos laplacianos. Aqui o objetivo é escolher o número k tal que todos os autovalores $\lambda_1, \ldots, \lambda_k$ são bem pequenos, mas λ_{k+1} é relativamente grande. Existem varias justificativas para esse procedimento. O primeiro é baseado na teoria de pertubações, onde observamos que no caso ideal de k agrupamentos completamente desconectados, o autovalor 0 tem multiplicidade k, e então existe um intervalo para o autovalor $\lambda_{k+1} > 0$. Outras explicações podem ser dada pela teoria de agrupamento espectral. Aqui, muitos invariantes geométricos do grafo podem ser expressos ou limitados com a ajuda dos primeiros autovalores do grafo Laplaciano. Em particular, os tamanhos dos cortes estão intimamente relacionados com o tamanho dos primeiros autovalores, ver [16] e [17].

Exemplo 8.2. Vamos considerar conjuntos de dados semelhantes ao do inicio do capítulo, mas para variar a dificuldade do agrupamento consideramos o Gaussiano com

variação crescente. Como podemos ver nos histogramas abaixo:

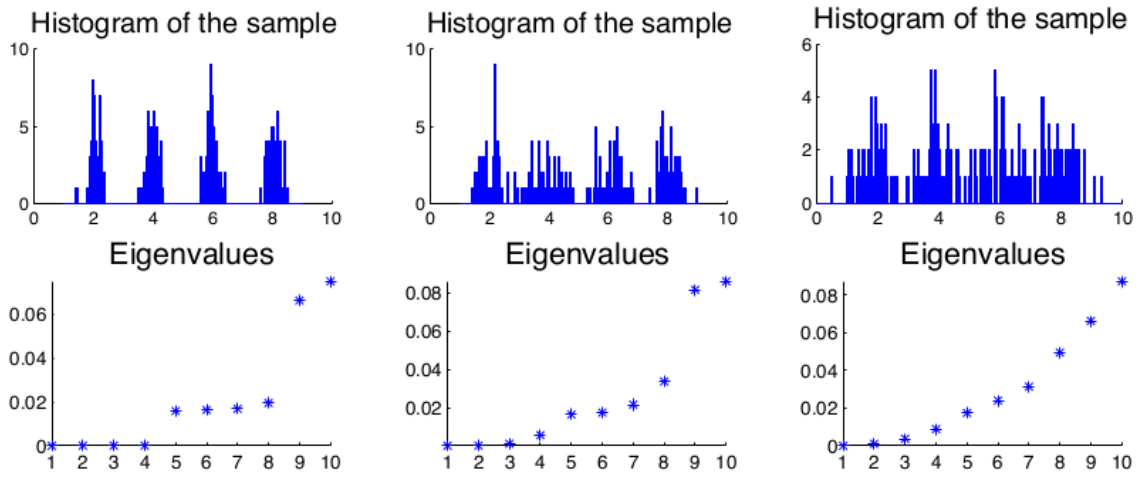

Figura 8.5: Três conjuntos de dados e os 10 menores autovalores de L_{rw} do grafo 10-vizinhos mais próximos de Von Ulrike, [23].

Construímos o grafo do k-vizinhos mais próximos para $k = 10$ e plotamos os autovalores de L_{rw} das diferentes amostras. O primeiro conjunto de dados consiste de quatro agrupamentos disjuntos e podemos ver que os quatro primeiros autovalores estão bem próximos de zero. E há um intervalo maior entre o quarto e o quinto autovalor, em que $|\lambda_5 - \lambda_4|$ é relativamente grande. De acordo, com o heurística do eigengap essa diferença indica que o conjunto de dados contém quatro agrupamentos.

O mesmo comportamento também pode ser visto para o grafo totalmente conexo, como podemos ver na figura a seguir.

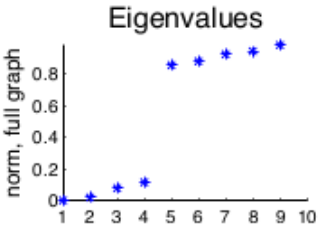

Figura 8.6: Autovalores de L_{rw} baseado no grafo totalmente conexo do primeiro histograma de Von Ulrike, [23].

Assim podemos ver que a heurística do eigengap funciona bem se os agrupamentos nos dados forem bem pronunciados.

No entanto quanto mais os agrupamentos se sobrepõem menos eficaz é essa heurística. Podemos ver isso no segundo conjunto de dados, onde ainda existe uma lacuna entre o quarto e o quinto autovalor, mas não é tão claro para detectar como no primeiro caso.

Finalmente no último conjunto de dados não há lacunas bem definidas, pois as diferenças entre todos os autovalores são aproximadamente iguais. Mas, por outro

lado, os agrupamentos neste conjunto de dados se sobrepõem tanto que também pode indicar que não há agrupamentos claros nos dados de qualquer forma.

Isso mostra que, como a maioria dos métodos para escolher o número de agrupamentos, a heurística do eigengap geralmente funciona bem se os dados tiverem agrupamentos muito bem pronunciados, mas em casos ambíguos também retorna resultados ambíguos.

8.4 A Escolha do Grafo Laplaciano

Uma questão fundamental relacionada ao agrupamento espectral é a questão de qual dos três grafos laplacianos deve ser usado para calcular os autovetores. Antes de resolver esta questão deve-se sempre olhar para o grau de distribuição do grafo de similaridade.

Definição 8.1. *O grau de distribuição de um vértice no grafo é a fração de vértices com um certo grau, ou seja, se existir n vértices no grafo e n_s deles tem grau k então o grau de distribuição de k é n_s/n.*

Se o grafo é muito regular e a maioria dos vértices tem aproximadamente o mesmo grau, então todos os laplacianos são muito semelhantes entre si e funcionam igualmente bem para agrupamento. No entanto, se os graus no grafo são amplamente distribuídos, então os laplacianos diferem consideravelmente. Existem vários argumentos que defendem a utilização do agrupamento espectral normalizado ao invés do não normalizado, e no caso do normalizado usar o autovalores de L_{rw} em vez de L_{sym}.

Objetivos de agrupamento satisfeito por diferentes algoritmos

O primeiro argumento a favor do agrupamento espectral normalizado vem do ponto de vista da partição do grafo. Para simplificar, discutiremos o caso $k = 2$. Em geral, agrupamentos tem dois objetivos diferentes:

1 Queremos encontrar uma partição tal que os pontos em diferentes agrupamentos são diferentes uns dos outros, ou seja, queremos minimizar a similaridade entre agrupamentos. Nas configurações de grafo, isso significa minimizar:

$$\sum_{i\in A, j\in A^c} w_{ij}.$$

2 Queremos encontrar uma partição tal que pontos num mesmo agrupamento são semelhantes entre si, ou seja, queremos maximizar a similaridade dentro do agrupamento, isso significa que devemos maximizar

$$\sum_{i,j\in A} w_{ij} \text{ e } \sum_{i,j\in A^c} w_{ij}.$$

Tanto RCut e NCut implementam diretamente o primeiro ponto incorporando explicitamente $cut(A, A^c)$ na sua função. No entanto, relativamente ao segundo ponto, ambos os algoritmos se comportam de forma diferente. Observe que:

$$\sum_{i,j\in A} w_{ij} = \sum_{i\in A, j\in A\cup A^c} w_{ij} - \sum_{i\in A, j\in A^c} w_{ij} = \sum_{i\in A} d_i - cut(A, A^c) = vol(A) - cut(A, A^c).$$

Assim, a similaridade dentro do agrupamento é maximizada se $cut(A, A^c)$ é pequeno e se $vol(A)$ é grande. No caso do $NCut$, isso também é parte da função objetivo, pois queremos maximizar tanto $vol(A)$ quanto $vol(A^c)$. Assim, $NCut$ implementa o segundo objetivo. Isso pode ser visto de forma ainda mais explicita, se considerarmos outra função de corte do grafo, a saber,

$$MinMaxCut(A_1, \ldots, A_k) = \sum_{i=1}^{n} \frac{cut(A_i, A_i^c)}{\sum_{i,j \in A} w_{ij}}.$$

Aqui o denominador contém, diretamente, a similaridade dentro dos agrupamentos, ao invés das somas de similaridades dentro do agrupamento, como no $NCut$. Mas como uma boa solução de $NCut$ terá um valor pequeno de $cut(A, A^c)$ de qualquer maneira, $NCut$ e $MinMaxCut$ são minimizados por cortes semelhantes. Além disso, o relaxamento do $MinMaxCut$ leva exatamente ao mesmo problema de otimização que o relaxamento do $NCut$, ou seja, o agrupamento espectral com autovetores de L_{rw}.

Considere, agora, o caso do $RCut$. Aqui o objetivo é maximizar $|A|$ e $|A^c|$ ao invés de $vol(A)$ e $vol(A^c)$. Mas $|A|$ e $|A^c|$ não estão, necessariamente, relacionados com a similaridade dentro do agrupamento, pois a similaridade dentro do agrupamento depende das arestas e não do número de vértices em A. Apenas imagine um conjunto A que não tem muito vértices, todos os quais tem apenas uma ponderação muito baixa entre si. Assim, a minimização do $RCut$ não tenta maximizar a similaridade dentro do agrupamento, e o mesmo então, é verdade para o relaxamento do agrupamento espectral não normalizados. Portanto, esse é o primeiro ponto importante para deixar na mente, agrupamento espectral normalizado implementa ambos os objetivos, enquanto que agrupamento espectral não normalizado implementa apenas o primeiro.

Problemas de consistência

Um argumento completamente diferente para a superioridade do agrupamento espectral normalizado vem de uma análise estatística de ambos os algoritmos. Pode se provar que em condições muito suaves, ambos os algoritmos do agrupamento espectral normalizados são estatisticamente consistentes. Isto significa que se assumirmos que os dados foram amostrados aleatoriamente e de acordo com alguma distribuição de probabilidade de algum espaço subjacente, e se deixarmos o tamanho aumentar para o infinito, então o resultado do agrupamento espectral normalizado converge, e a partição do limite é geralmente uma partição sensível ao espaço subjacente. Esses resultados não são necessariamente válidos para agrupamento espectral não normalizados.

Pode se provar que o agrupamento espectral não normalizado pode falhar na convergência, ou que ele pode convergir para soluções que constroem agrupamentos consistindo de um único ponto do espaço de dados.

Existe uma condição simples e necessária que precisa ser satisfeita para evitar tais soluções triviais: os autovalores de L correspondente aos autovetores utilizados em agrupamento espectral não normalizados deve ser significativamente abaixo do grau minimo no grafo. Isto significa que se usarmos os primeiros k autovetores, então

$$\lambda_i \ll \min_{j=1,\ldots,n} d_j,$$

deve valer para todo $i = 1, \ldots, k$. A razão é que autovetores correspondentes aos autovalores com $\lambda >> \min d_j$, aproxima as funções de Dirac, ou seja, elas são aproximadamente zero em todas menos em uma coordenada. Se esses autovetores são usados para agrupamento, então eles separam um vértice onde o autovetor é diferente de zero,

de todos os outros vértices e nós claramente não queremos construir tal partição. Como uma ilustração desse fenômeno, considere de o exemplo :

Exemplo 8.3. Considere novamente o exemplo do início do capítulo, tomamos os primeiros autovalores e autovetores do grafo Laplaciano não normalizado com base no grafo totalmente conexo para diferentes escolhas do parâmetro σ na função Gaussiana. Nas figuras, os autovalores $\lambda >> \min d_j$ são plotados como estrelas e o menores são plotados como diamantes. A linha tracejada indica tal minimo.

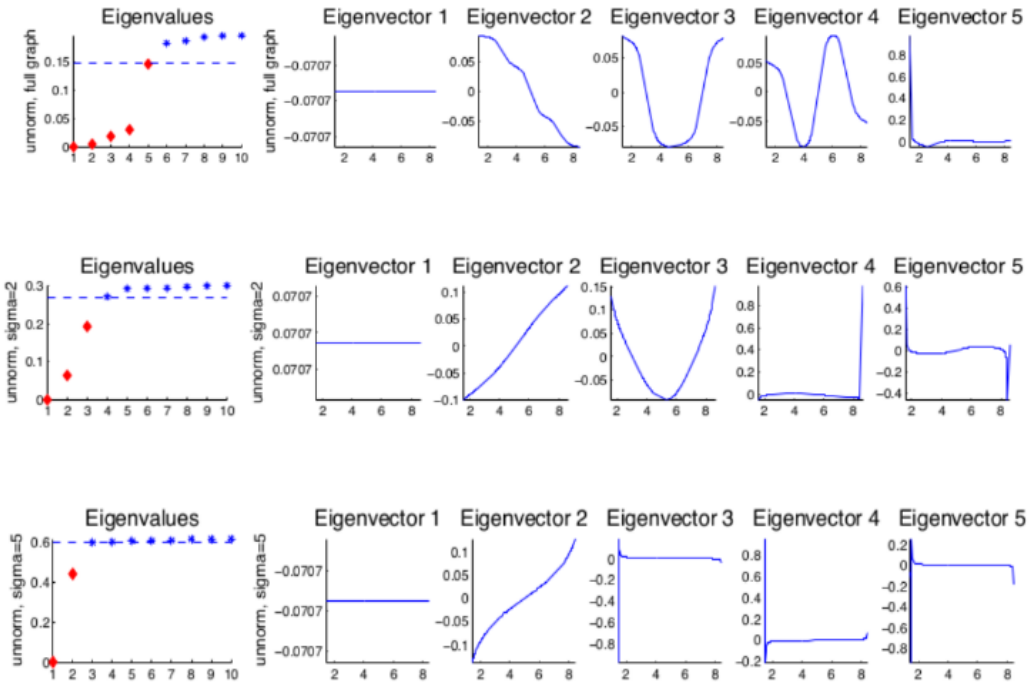

Figura 8.7: Autovalores de L com parâmetros $\sigma = 1$, $\sigma = 2$, $\sigma = 5$.

Em geral podemos ver que os autovetores correspondentes aos autovalores que estão muito abaixo da linha tracejada são autovetores úteis.

No caso $\sigma = 1$, podemos ver na figura que os autovalores $2, 3$ e 4 estão significativamente abaixo da linha tracejada e os autovetores correspondentes são úteis.

Se aumentarmos o parâmetro, podemos observar que os autovalores tendem a se mover para linha do $\min d_j$. No caso, $\sigma = 2$, apenas os três primeiros autovalores estão abaixo da linha tracejada. E no caso $\sigma = 5$ apenas os dois primeiros estão abaixo da linha tracejada, todos referentes ao primeiro histograma da figura 8.5. Em [22] temos que para $\sigma = 50$ só um autovalor abaixo da linha tracejada.

Podemos ver que um autovalor se aproxima ou está acima de $\min d_j$, seu autovetor correspondente se aproxima da função Dirac. Claro, esses autovetores não são adequadas para a construção de agrupamentos. Gostaríamos de salientar que esses problemas apenas dizem respeito aos autovetores da matriz L, e eles não ocorrem para L_{rw} e L_{sym}.

Qual Laplaciana normalizada?

Analisando as diferenças entre os dois algoritmos do agrupamento espectral normalizado usando L_{rw} e L_{sym}, todas as três explicações de agrupamento espectral são

a favor de L_{rw}. A razão é que os autovetores de L_{rw} são os vetores indicadores, enquanto os autovetores de L_{sym} são multiplicados por $D^{\frac{1}{2}}$, o que pode levar a resultados indesejados. Como o uso do L_{sym} também não tem vantagens computacionais, nós defendemos o uso do L_{rw}.

9 Aplicação

Neste capítulo vamos introduzir algumas aplicações do primeiro algoritmo apresentado no capítulo 4:

1) Construir o grafo $G = (V, E)$ e a sua matriz Laplaciana, L.

2) Calcular os k primeiros autovetores, $v_1, \ldots, v_k$, de L.

3) Considerar V a matriz formada pelos k autovetores nas colunas.

4) Tomar y_i o vetor correspondente a i-ésima linha de V.

5) Aplicar o algoritmo k-means nos vetores y_i.

Agradecemos a Prof. Dra. Leandra Bordignon (UFAC) que gentilmente disponibilizou os dados que foram utilizados nas aplicações a seguir, onde nosso conjunto de dados são folhas de soja, em particular, a quinta folha de cada vaso. Inicialmente os dados eram fotos da folha de soja, coloridas, sem padrão de tamanho, os professores Dr. Thiago de Melo e Dr. Jamil Viana trabalharam nessas fotos, tirando ruídos, centralizando cada folha e convertendo para escalas de cinza e obtiveram a seguinte imagem:

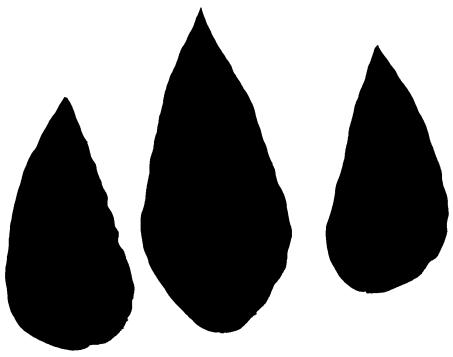

Figura 9.1: Folha de Soja

Cada folha de soja possui três folíolos como na imagem, calculadas as áreas de cada folíolo podemos transformar essas imagens em pontos de $\mathbb{R}^3$. Nosso conjunto de dados para as aplicações são, então, esses pontos.

Exemplo 9.1. Nesta primeira aplicação consideramos um conjunto de 12 folhas de sojas que foram submetidas a condições climáticas diferentes, sendo 7 com alto teor de carbono (círculos azuis na imagem) e 5 com carbono normal (círculos vermelhos na imagem), todas expostas a altas temperaturas.

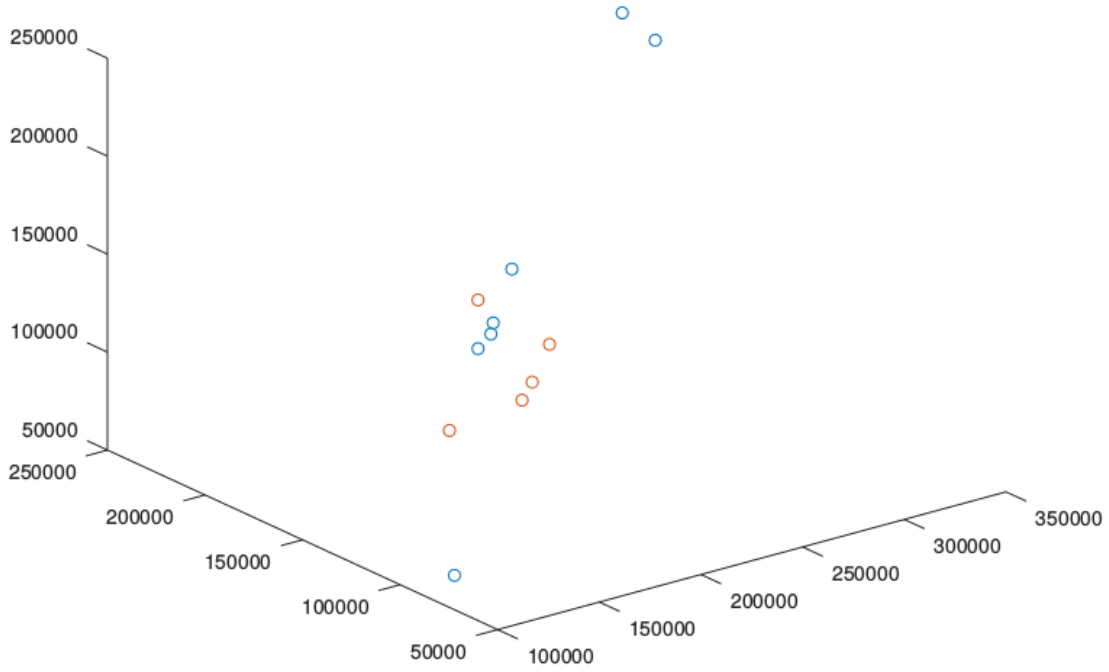

Figura 9.2: 12 folhas de soja transformados em pontos de $\mathbb{R}^3$

A partir do conjunto de pontos usamos a linguagem "Python" para calcular a matriz Laplaciana usando a similaridade k-vizinhos mais próximos, para $k = 6$:

$$L = \begin{pmatrix} 6 & 0 & 0 & -1 & 0 & 0 & -1 & -1 & -1 & 0 & -1 & -1 \\ 0 & 7 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 7 & -1 & -1 & -1 & 0 & -1 & -1 & -1 & 0 & 0 \\ -1 & -1 & -1 & 9 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 \\ 0 & -1 & -1 & -1 & 7 & -1 & -1 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & -1 & -1 & -1 & 5 & 0 & 0 & 0 & -1 & 0 & 0 \\ -1 & -1 & 0 & -1 & -1 & 0 & 8 & -1 & -1 & 0 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & 0 & -1 & 9 & -1 & 0 & -1 & -1 \\ -1 & 0 & -1 & -1 & 0 & 0 & -1 & -1 & 7 & 0 & -1 & -1 \\ 0 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 5 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & 5 & -1 \\ -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & -1 & 5 \end{pmatrix}.$$

No passo (2), calculamos os autovalores de L, usando o programa "Octave", aqui mostraremos apenas a parte inteira de cada autovalor:

$$0, 2, 5, 6, 7, 8, 9, 10$$

Podemos ver dos valores acima que a maior lacuna está entre o segundo autovalor e o terceiro, 2 e 5 respectivamente, (isso condiz com o que desejamos, uma vez que temos

duas condições diferentes). Então de acordo com o heurística do eigengap devemos tomar $k = 2$ e tomamos a matriz V como no passo (3) do algoritmo acima.

$$V = \begin{pmatrix} 1 & -0.30389 \\ 1 & 0.22688 \\ 1 & 0.21570 \\ 1 & 0.10354 \\ 1 & 0.22688 \\ 1 & 0.40858 \\ 1 & -0.14987 \\ 1 & -0.10354 \\ 1 & -0.21570 \\ 1 & 0.40858 \\ 1 & -0.40858 \\ 1 & -0.40858 \end{pmatrix}.$$

Tomando y_i como no passo (4) e usando o algoritmo k-means, para $k = 2$, temos os seguintes resultados:

$$I^t = \begin{pmatrix} 1 & 2 & 2 & 2 & 2 & 2 & 1 & 1 & 1 & 2 & 1 & 1 \end{pmatrix}.$$

$$C = \begin{pmatrix} 1 & -0.26503 \\ 1 & 0.26503 \end{pmatrix},$$

onde I representa cada agrupamento que cada vetor y_i irá pertencer e C mostra os centroides de cada agrupamento. Graficamente obtemos:

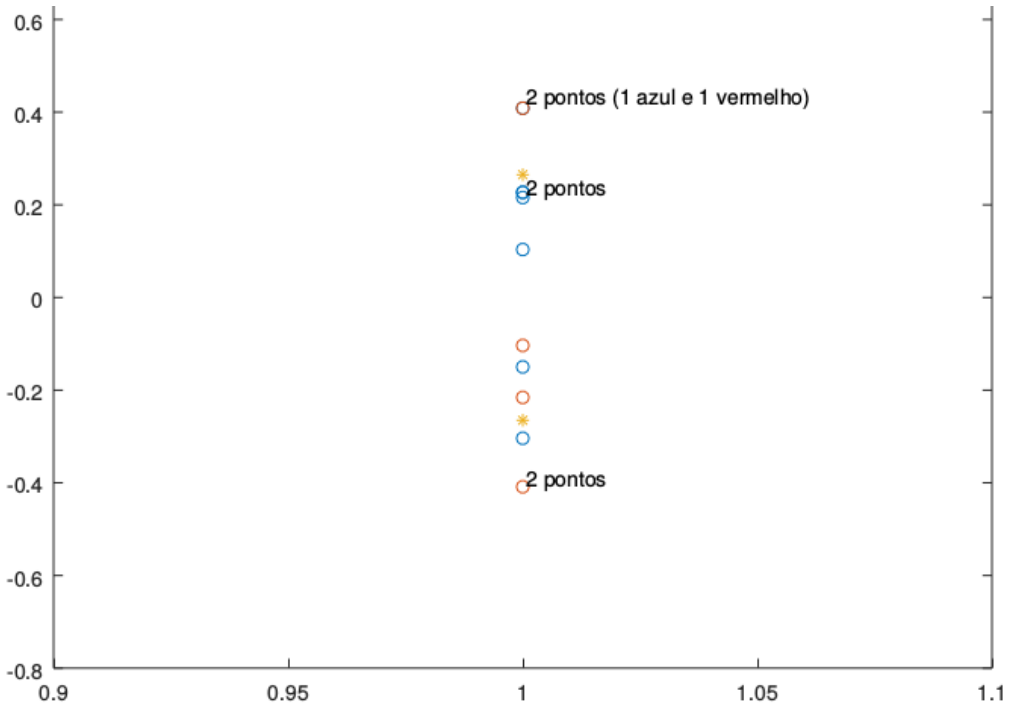

Figura 9.3: Agrupamentos, com 12 folhas, $k = 6$-vizinhos mais próximos e $k = 2$-means

Na figura acima as estrelas representam os centroides de cada agrupamento, podemos ver que a linha do zero separa os dois agrupamentos. Os círculos azuis representam as folhas que foram submetidas a alto teor de carbono e temperatura alta e os círculos vermelhos as folhas com teor de carbono normal e alta temperatura. Note que temos quatro círculos azuis no agrupamento acima da linha do zero e dois no outro agrupamento e um círculos vermelho no agrupamento acima da linha do zero, ou seja, temos três círculos que foram mandados para o conjunto errado através do agrupamento espectral.

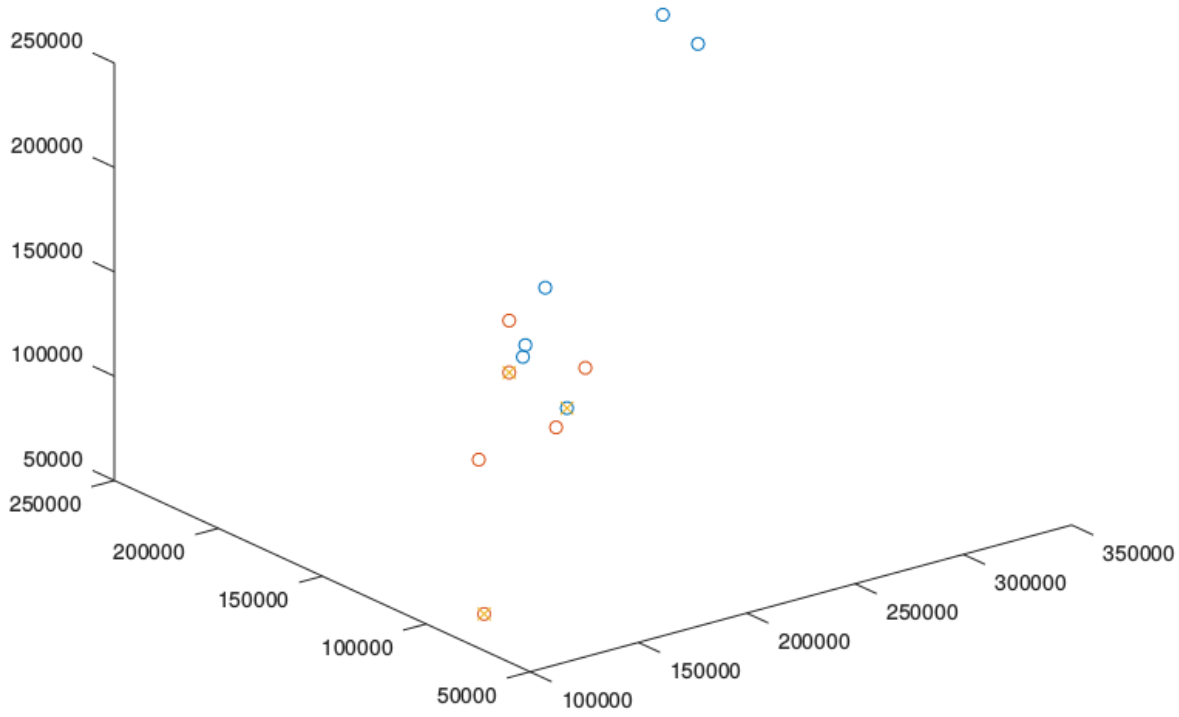

Figura 9.4: Pontos iniciais separados de acordo com o algoritmo

Na imagem 9.4 os círculos com um $\times$ representam os pontos que foram classificados de forma errônea.

Na aplicação a seguir seguiremos os mesmos passos, mas agora consideraremos três condições climáticas diferentes, tomamos a quinta folha de cada vaso, com 16 folhas com carbono alto e temperatura alta (círculo azul), 16 com carbono alto e temperatura normal (círculo vermelho) e 16 com carbono normal e temperatura alta (círculo laranja), totalizando 48 folhas.

Exemplo 9.2.

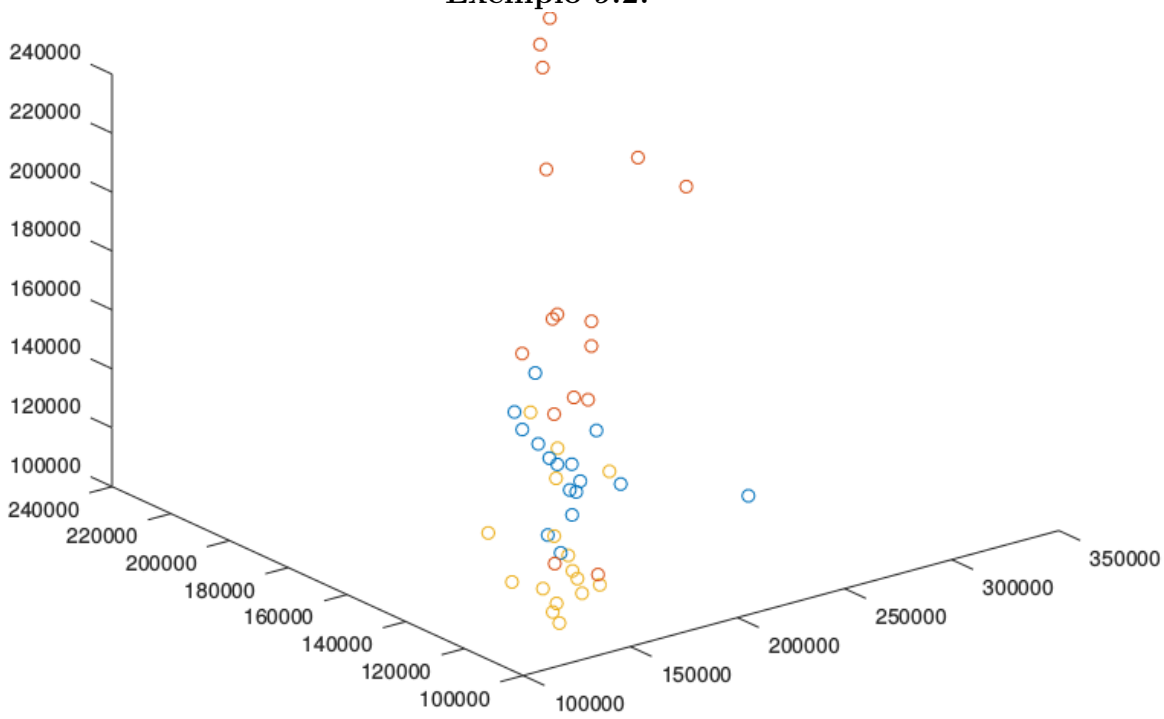

Figura 9.5: Pontos iniciais

Ao usar o algoritmo para $k = 16$ no k-vizinhos mais próximos obtemos os seguintes autovalores da Laplaciana:

$$0 \quad 1 \quad 6 \quad 13 \quad 15 \quad 16 \quad 17 \quad 18 \quad 19 \quad 20 \quad 21 \quad 22 \quad 23 \quad 24 \quad 25 \quad 26 \ .$$

De onde podemos ver, pelo heurística do eigengap que o número k que usaremos no k-means é $k = 3$. E obtemos a imagem 9.6, onde as estrelas são os centros e cada cor representa um agrupamento diferente.

Na imagem 9.7 temos o agrupamento da figura 9.6 mas agora cada cor representa uma condição climática diferente, novamente: os círculo azuis representam as folhas com carbono alto e temperatura alta, os círculo vermelhos as folhas com carbono alto e temperatura normal e os círculo laranjas as folhas com carbono normal e temperatura alta.

De onde já podemos ver o algoritmo não retornou os conjuntos iniciais separados de acordo com as condições climáticas, o que fica ainda mais claro na figura 9.8, que mostra o conjunto de pontos inicial separado de acordo com o algoritmo, ou seja, cada cor representa um agrupamento.

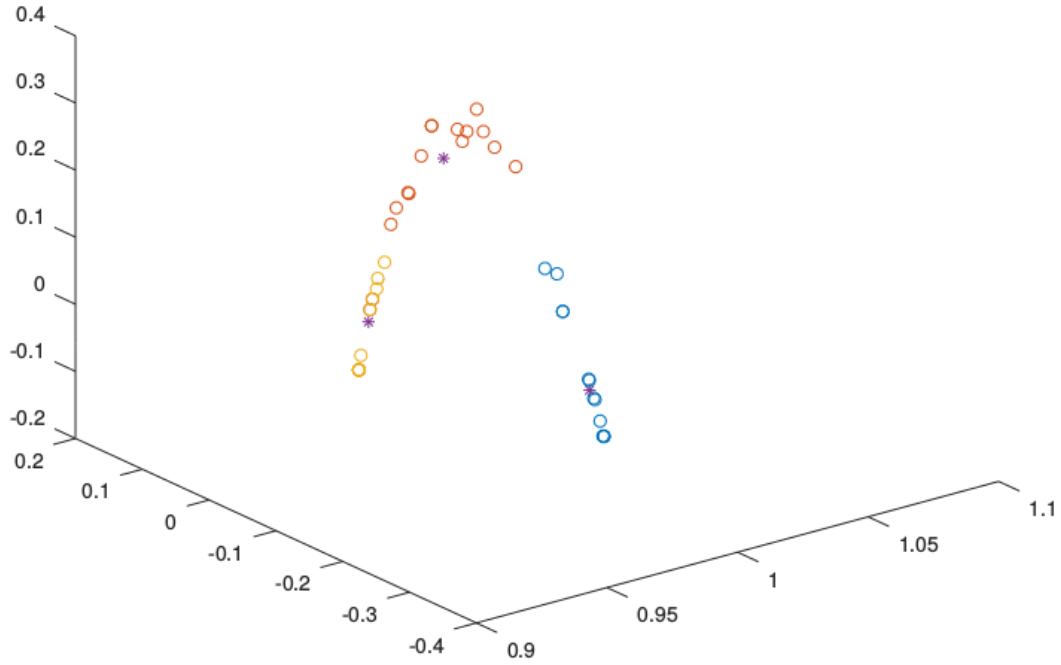

Figura 9.6: Agrupamentos, com 48 folhas, $k = 16$-vizinhos mais próximos e $k = 3$-means.

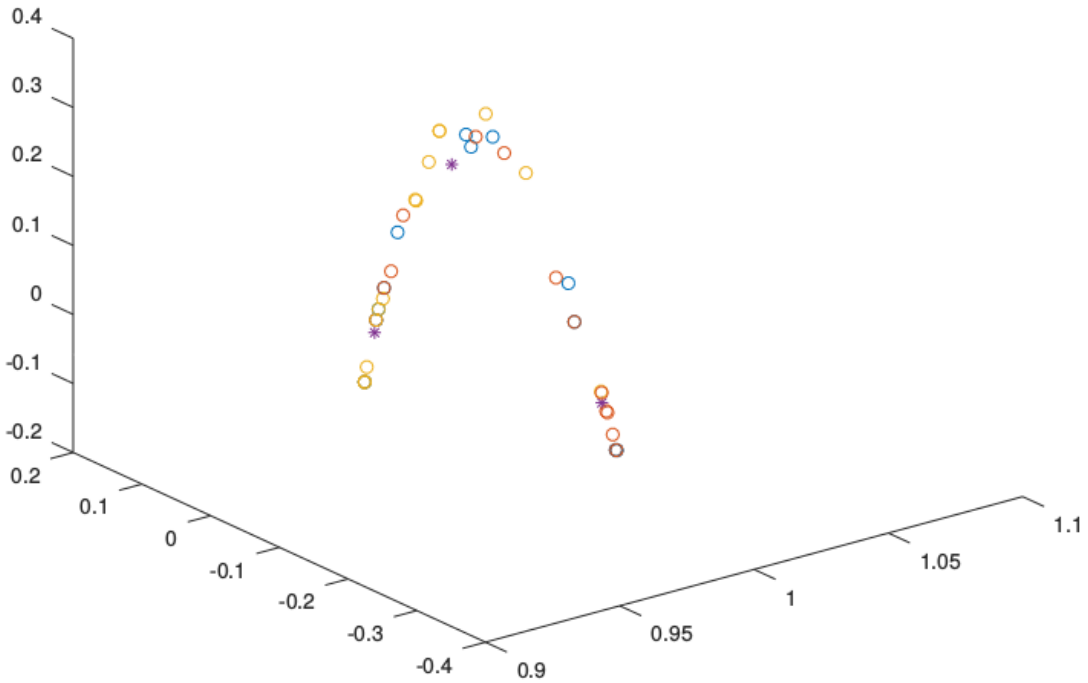

Figura 9.7: Agrupamentos, com 48 folhas.

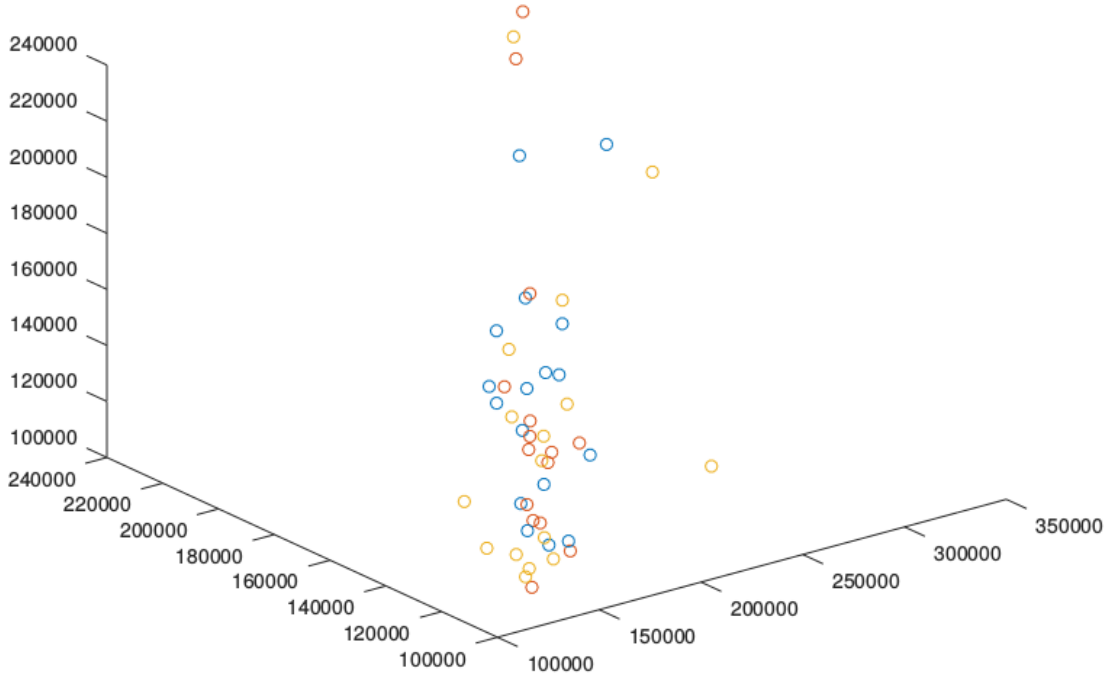

Figura 9.8: Pontos inicial separados de acordo com o algoritmo

Exemplo 9.3. Obtemos no exemplo 9.2, três agrupamentos, contudo ao trocarmos o valor de k de $k = 16$ para $k = 24$ no k-nearest neighboor, obtemos os seguintes autovalores para a Laplaciana:

$$0 \quad 6 \quad 23 \quad 24 \quad 25 \quad 27 \quad 28 \quad 29 \quad 30 \quad 32 \quad 33 \quad 34 \quad 35 \quad 36 \quad 37 \quad 39 \quad 41 \quad 42 \ .$$

De onde teriamos apenas 2 agrupamentos, pelo heurística do eigengap, e escolheriamos $k = 2$ no k-means, de onde obteriamos os seguintes agrupamentos:

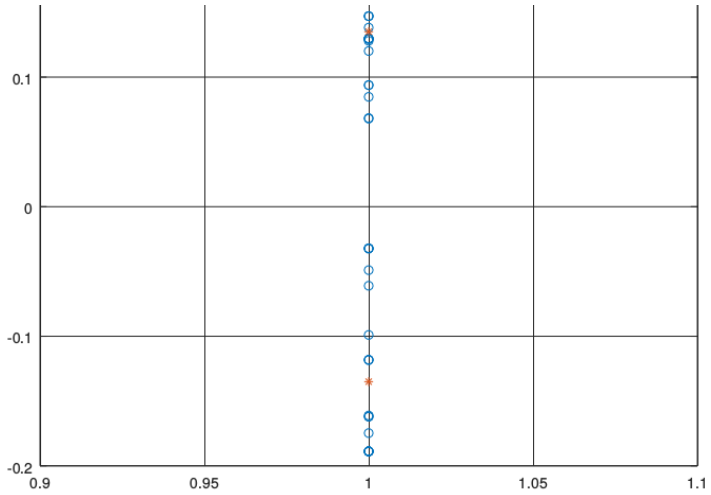

Figura 9.9: Agrupamentos, com 48 folhas, $k = 24$-vizinhos mais próximos e $k = 2$-means.

Exemplo 9.4. Se desconsiderarmos a teoria do heurística do eigengap no exemplo 9.3, uma vez que sabemos ter três agrupamentos distintos no nosso conjunto de dados e tomando $k = 3$ obteríamos o seguinte resultado, onde o terceiro agrupamento é composto de apenas um ponto.

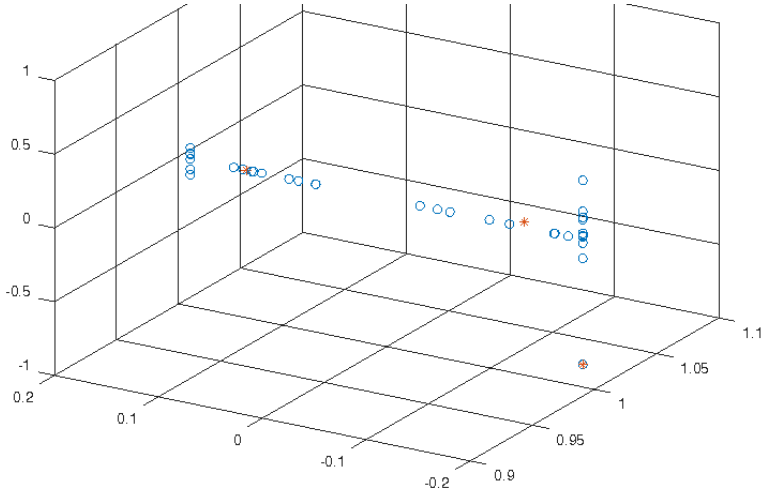

Figura 9.10: Agrupamentos, com 48 folhas, $k = 24$-vizinhos mais próximos e $k = 3$-means.

De todas as aplicações podemos ver como a escolha do parâmetro k para o k-vizinhos mais próximos e para o k-means é crucial e pode mudar todo o resultado.

Referências

[1] Bhatia, R.(1997) *Matrix Analysis.* Springer, New York.

[2] Bie T.D. e Cristianini, N. (2006). *Fast SDP relaxations of graph cut clustering, transduction, and other combinatorial problems.* Journal of Machine Learning Research, 7 (1409-1436).

[3] Bui, T.N. e Jones, C.(1992). *Finding good approximate vertex and edge partitions is NP-hard.* Inf. Process. Lett. 42(3) (153-159).

[4] Chung, F. (1997). *Spectral Graph Theory.* Conference Board of the Mathematical Sciences, Washington.

[5] Fouss, F., Pirotte, A., Renders, J.-M., e Saerens, M. (2006) *A novel way of computing dissimilaritues between nodes of a graph, with application to collaborative filtering and subspace of the graph nodes.* Information Systems Research Unit (ISYS/IAG).

[6] Golub, G. e Van Loan, C.(1996) *Matrix Computations.* Baltimore: Johns Hopkins University Press.

[7] Guaterry, S. e Miller, G. L.(1998). *On the quality of spectral separators.* SIAM Journal of Matrix Anal. Appl.,19(3) (701-719).

[8] Gutman, I. e Xiao, W. (2004) *Generalized inverse of the Laplacian matrix and some applications.* Bulletin de l'Académie serbe des sciences et des arts,129,(15-23).

[9] Hastie, T., Tibshirani, R., e Friedman, J. (2001) *The elements of statistical learning.* New York: Springer.

[10] Klein, D. e Randic, M. (1993) *Resistance distance.* Journal of Mathematical Chemistry, 12 (81-95).

[11] Lang, K.(2006) *Fixing two weaknesses of the spectral method.* Neural Computation, 16(6) (1299-1323).

[12] Lovásk, L. (1993) *Random walks on graphs: a survey.* In Combinatorics, Paul Erdos is eighty, Vol. 2, Keszthely (Hungary), (1-46).

[13] Luktepohl, H. (1997). *Handbook of Matrices.* Chichester: Wiley.

[14] Meila, M. e Shi, J. (2001) *A random walks view of spectral segmentation.* In 8th International Workshop on Artificial Intelligence and Statistics.

[15] Meila, M. e Shi, J. (2001) *Learning segmentation by random walks view of spectral.* In Neural Information Processing Systems, 18.

[16] Mohar, B. (1991). *The Laplacian spectrum of graphs.* In Graph Theory, Combinatorics, and Applications. Vol. 2.

[17] Nadler, B., Lafon, S., Coifman, R., e Kevrekidis, I. (2006) *Diffusion maps, spectral clustering and eigenfunctions of Fokker-Planck operators.* Advances in Neural Information Processing Systems 18 (955-962), Cambridge.

[18] Ng, A., Jordan, M., e Weiss, Y. (2002) *On spectral clustering: analysis and an algorithm.* Em T. Dietterich, S. Becker e Z. Ghahramani, Advance in Neural Information Processing Sistems 14.

[19] Rencher, A. e Schaalje, G. (2000) *Linear models in statistics.* A John Wiley & Sons, INC., Publication

[20] Seshu S., Read M. B., (1961) *Linear Graphs and Electrical Networks*, Addison-Wesley Publishing Company, Inc.

[21] Shi, J. e Malik, J. (2000). *Normalized cuts and image segmentation.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(8), (889-905).

[22] Von Luxburg, U., Bousquet, O., Belkin, M. (2004). *Consistency of spectral clustering.* Max Planck Institute for Biological Cybernetics.

[23] Von Luxburg, U. (2007). *A tutorial on spectral clustering.* Max Planck Institute for Biological Cybernetics.

[24] Xiao, W. e Gutman, I. (2003) *Resistance distance and Laplacian spectrum*

A Propriedades de $L^\dagger$

Antes de entrarmos nas propriedades da Laplaciana generalizada, $L^\dagger$, vamos mostrar a igualdade

$$L = V\Lambda V^t.$$

Para isso, vale lembrar que:

(i) Se uma matriz L é simétrica, $L = L^t$, então dois autovetores correspondentes a autovalores distintos são ortogonais.

De fato, se λ_1 e λ_2 são autovalores distintos e v_1, v_2 são seus autovetores correspondentes, respectivamente, então vale:

$$v_1^t L v_2 = v_1^t(Lv_2) = v_1^t(\lambda_2 v_2) = \lambda_2(v_1^t v_2).$$

Analogamente, vale:

$$v_1^t L v_2 = \lambda_1(v_1^t v_2).$$

Logo,

$$\lambda_2(v_1^t v_2) = \lambda_1(v_1^t v_2).$$

De onde temos que:

$$(\lambda_1 - \lambda_2)(v_1^t v_2) = 0 \rightarrow v_1^t v_2 = 0.$$

(ii) Se L tem um conjunto completo de autovetores (ou seja, qualquer autovalor λ com multiplicidade k possui k correspondentes autovetores linearmente independentes), podemos escrever $L = V\Lambda V^{-1}$ onde Λ é uma matriz diagonal de autovalores de L e as colunas de V são os autovetores de L.

De fato, como $Lv_i = \lambda_i v_i, \forall i$ podemos escrever $LV = V\Lambda$ e como V possui inversa temos que:

$$L = V\Lambda V^{-1}.$$

Por fim note que as colunas de V são ortogonais, logo, $VV^t = I$ de onde temos que $V^t = V^{-1}$, substituindo essa igualdade em (ii) obtemos a igualdade desejada.

Definição A.1. *$L^\dagger = V\Lambda^\dagger V^t$, onde a matriz diagonal $\Lambda^\dagger$ tem os elementos $1/\lambda_i$ se $\lambda_i \neq 0$ e 0 se $\lambda_i = 0$, na diagonal.*

As entradas de $L^\dagger$ pode ser vistas como:

$$l_{ij}^\dagger = \sum_{k=2}^{n} \frac{1}{\lambda_k} v_{ik} v_{jk}. \tag{A.1}$$

Além disso, como todos os autovalores são não negativos, $L^\dagger$ é semi-definida positiva. Como L é real e simétrica, temos que $L^\dagger$ também o é.

Proposição A.2. *A matriz Laplaciana e a sua inversa generalizada satisfazem as relações:*

$$LJ = JL = 0; L^\dagger J = JL^\dagger = 0,$$

onde J é a matriz com todas as entradas iguais a 1.

Demonstração. A primeira igualdade sai direto do fato que que a soma dos elementos de L é igual a 0, ou seja,

$$\sum_{j=1}^{n} l_{ij} = 0, \forall i$$

Para a segunda igualdade note que:

$$\sum_{j=1}^{n} l_{ij}^\dagger = \sum_{j=1}^{n}\sum_{k=2}^{n} \frac{1}{\lambda_k} v_{ik} v_{jk} = \left(\sum_{k=2}^{n} \frac{1}{\lambda_k} v_{ik}\right)\left(\sum_{j=1}^{n} v_{jk}\right) = 0.$$

□

Da igualdade $L = V\Lambda V^t$, podemos escrever:

$$l_{ij} = \sum_{k=1}^{n} \lambda_k v_{ik} v_{jk}. \tag{A.2}$$

Ainda, como $VV^t = I = V^tV$ vale:

$$\sum_{k=1}^{n} v_{ki} v_{kj} = \sum_{k=1}^{n} v_{ik} v_{jk} = \delta_{ij}, \tag{A.3}$$

onde $\delta_{ij} = 1$ para $i = j$ e $\delta_{ij} = 0$ caso contrário.

Proposição A.3. *Se L e $L^\dagger$ pertencem a grafos conexos com n vértices, então:*

$$LL^\dagger = L^\dagger L = I - \frac{1}{n} J.$$

Demonstração. Das equações (A.1) e (A.2), temos que:

$$(LL^\dagger)_{ij} = \sum_{h=1}^{n} l_{ih} l_h^\dagger j = \sum_{h=1}^{n} \left(\sum_{k=1}^{n} \lambda_k v_{ik} v_{hk}\right)\left(\sum_{l=2}^{n} \frac{1}{\lambda_l} v_{hl} v_{jl}\right) = \sum_{k=1}^{n}\sum_{l=2}^{n} \frac{\lambda_k}{\lambda_l} v_{ik} v_{jl}\left(\sum_{h=1}^{n} v_{hk} v_{hl}\right).$$

Da equação (A.3) temos que:

$$(LL^\dagger)_{ij} = \sum_{k=1}^{n}\sum_{l=2}^{n} \frac{\lambda_k}{\lambda_l} v_{ik} v_{jl} \delta_{kl} = \sum_{l=2}^{n} v_{il} v_{jl} = \sum_{l=1}^{n} v_{il} v_{jl} - v_{i1} v_{j1}.$$

Novamente da equação (A.3) e do fato de $v_1 = \frac{1}{\sqrt{n}}(1, \ldots, 1)^t$, ou seja, $v_{i1} = \frac{1}{\sqrt{n}}$ segue que:

$$(LL^\dagger)_{ij} = \delta_{ij} - \frac{1}{n}.$$

□

Teorema A.4. *Seja G um grafo conexo, com, $\lambda_1 = 0, \lambda_2 \ldots, \lambda_n$ autovalores de L e $v_1, \ldots, v_n$ autovetores de L. Então $v_1, \ldots, v_n$ também são autovetores de $L + \frac{1}{n}J$ com autovalores $1, \lambda_2 \ldots, \lambda_n$.*

Demonstração. Tome $k > 1$. Então

$$\left(L - \frac{1}{n}J\right)v_k = Lv_k + \frac{1}{n}Jv_k = \lambda_k v_k,$$

uma vez que v_k é ortogonal a v_1 para todo $k = 2, \ldots, n$ temos que $\sum_{j=1}^{n} v_{jk} = 0$. Logo, $Jv_k = 0$.

Considere, agora, $k = 1$ então $Lu_1 = (0, \ldots, 0)^t$ e $Ju_1 = u_1$. Portanto,

$$\left(L - \frac{1}{k}J\right)v_1 = v_1.$$

□

Teorema A.5. *Se G é um grafo conexo, então a matriz inversa de $L + \frac{1}{n}J$ existe e é igual a $L^\dagger + \frac{1}{n}J$.*

Demonstração. Como vimos no teorema A.4 todos os autovalores de $L + \frac{1}{n}J$ são diferentes de zero logo existe $(L + \frac{1}{n}J)^{-1}$. Usando os resultados vistos anteriormente, e do fato de que $J^2 = nJ$, temos que:

$$(L + \frac{1}{n}J)(L^\dagger + \frac{1}{n}J) = LL^\dagger + \frac{1}{n}JL^\dagger + \frac{1}{n}LJ + \frac{1}{n^2}J^2 = (I - \frac{1}{n}J) + \frac{1}{n}J = I$$

□

Definição A.6. *A distância de resistência entre dois vértices x_i e x_j de um grafo G, r_{ij} é a resistência elétrica entre os vértices correspondentes. Além disso, definimos a matriz resistência R, onde as suas entradas $R_{ij} = \|r_{ij}\|$.*

Para mais detalhes sobre resistência elétrica ver [24]. Usando as leis de Ohm e Kirchhof, em [20] prova-se que:

$$r_{ij} = l^\dagger_{ii} + l^\dagger_{jj} - l^\dagger_{ij} - l^\dagger_{ji}. \tag{A.4}$$

Primeiramente, como $L^\dagger$ é simétrica podemos simplificar a equação (A.4):

$$r_{ij} = l^\dagger_{ii} + l^\dagger_{jj} - 2l^\dagger_{ij}. \tag{A.5}$$

Combinado (A.1) com a equação (A.5) temos:

$$r_{ij} = \sum_{k=2}^{n} \frac{1}{\lambda_k}(v_{ik}v_{ik} + v_{jk}v_{jk} - 2v_{ik}v_{jk}) = \sum_{k=2}^{n} \frac{1}{\lambda_k}(v_{ik} - v_{jk})^2. \tag{A.6}$$

Denotando $L^\dagger + \frac{1}{n}J = X$ temos de (A.4) a seguinte equação:

$$r_{ij} = X_{ii} + X_{jj} - 2X_{ij}, \tag{A.7}$$

Teorema A.7. *Se as matrizes $L, L^\dagger$ e R pertencem a um grafo conexo, então:*

$$LRL = -2L \text{ e } L^\dagger R L^\dagger = -2(L^\dagger)^3. \tag{A.8}$$

Demonstração.

$$(L^\dagger R L^\dagger)_{ij} = \sum_{k=1}^{n}\sum_{l=1}^{n} l^\dagger_{ik} r_{kl} l^\dagger_{lj} = \sum_{k=1}^{n}\sum_{l=1}^{n} l^\dagger_{ik}(l^\dagger_{kk} + l^\dagger_{ll} - 2l^\dagger_{kl}) l^\dagger_{lj}$$

Assim,

$$(L^\dagger R L^\dagger)_{ij} = \sum_{k=1}^{n} l^\dagger_{ik} l^\dagger_{kk}\Big(\sum_{l=1}^{n} l^\dagger_{lj}\Big) + \sum_{l=1}^{n} l^\dagger_{lj} l^\dagger_{ll}\Big(\sum_{k=1}^{n} l^\dagger_{ik}\Big) - 2\sum_{k=1}^{n}\sum_{l=1}^{n} l^\dagger_{ik} l^\dagger_{kl} l^\dagger_{lj}$$

Da proposição A.2 temos que:

$$\sum_{l=1}^{n} l^\dagger_{lj} = 0 \text{ e } \sum_{k=1}^{n} l^\dagger_{ik} = 0$$

e portanto

$$(L^\dagger R L^\dagger)_{ij} = -2\sum_{k=1}^{n}\sum_{l=1}^{n} l^\dagger_{ik} l^\dagger_{kl} l^\dagger_{lj} = -2(l^\dagger)^3_{ij}$$

De onde obtemos uma das igualdades. Multiplicando a mesma por L^2 do lado direito e do lado esquerdo temos:

$$L^2 L^\dagger R L^\dagger L^2 = -2L^2(L^\dagger)^3 L^2$$

Das proposições A.2 e A.3, temos que:

$$L^2 L^\dagger = L(LL^\dagger) = L(I - \frac{1}{n}J) = L$$

e

$$L^\dagger L^2 = (I - \frac{1}{n}J)L = L$$

Portanto,

$$L^2 L^\dagger R L^\dagger L^2 = LRL$$

e

$$L^2(L^\dagger)^3 L^2 = LL^\dagger L = (I - \frac{1}{n}J)L = L.$$

Juntando todas as informações temos:

$$LRL = L^2 L^\dagger R L^\dagger L^2 = -2L^2(L^\dagger)^3 L^2 = -2L.$$

□

Teorema A.8. *No caso de grafos conexos, a inversa generalizada porde ser expressa em termos da matriz resistência:*

$$L^\dagger = \frac{1}{2}\left[R - \frac{1}{n}(RJ + JR) + \frac{1}{n^2}JRJ\right].$$

Demonstração. Do teorema anterior temos que:

$$L^\dagger R L^\dagger = -2(L^\dagger)^3. \tag{A.9}$$

Multiplicando ambos os lados da equação por L temos que:

$$LL^\dagger R L^\dagger L = -2L(L^\dagger)^3 L.$$

Da proposição A.2 e A.3 temos que:

$$L(L^\dagger)^3 L = (I - \frac{1}{n}J)L^\dagger(I - \frac{1}{n}J) = L^\dagger(I - \frac{1}{n}J) = L^\dagger.$$

Assim,

$$-2L^\dagger = LL^\dagger R L^\dagger L = \left(I - \frac{1}{n}J\right)R\left(I - \frac{1}{n}J\right) = \left(R - \frac{1}{n}JR\right)\left(I - \frac{1}{n}J\right)$$

$$-2L^\dagger = R - \frac{1}{n}RJ - \frac{1}{n}JR + \frac{1}{n^2}JRJ.$$

□