MATE32

Algoritmos

Ricardo A. Rios
Tatiane N. Rios

Agenda

- Introdução
- Algoritmos Particionais
  - K-means
- Algoritmos Hierárquicos
  - Single-link
  - Average-link
  - Complete-link

Introdução

- Análise dos dados para **identificação de estruturas** similares organizadas como clusters;
- Cada algoritmo de agrupamento emprega um **critério de agrupamento**, que impõe uma estrutura aos dados.
- Os **algoritmos de agrupamento** são divididos em:
  - Particionais
  - Hierárquicos
  - Baseados em grid
  - Baseados em densidade

Introdução

- Conjunto de dados $X = \{x_1, x_2, \ldots, x_n\}$
- Para um **agrupameto** do tipo **hard** tem-se:
  - Uma partição de X em k clusters sendo $\pi = \{C_1, C_2, \ldots, C_k\}$, com $k < n$ tal que
    - $Cj \neq \varnothing, \quad j = 1, \ldots, k$
    - $\cup^{k}_{j=1} C_j = X$
    - $C_j \cap C_l = \varnothing \quad j,l = 1, \ldots, k; \; j \neq l$

Algoritmos Particionais

- Técnicas iterativas;
- Passos gerais:
  - Criação de uma partição inicial;
  - Instâncias são movidas entre clusters para melhorar o critério de agrupamento;
  - Computacionalmente eficientes;

Algoritmos Particionais

- Critério de agrupamento:
  - Erro quadrático (compactação dos grupos);
- Objetivo:
  - Obter um particionamento que minimiza o erro quadrático para um número fixo de clusters
    - Minimizar o erro quadrático = Maximizar a variação entre grupos;

Algoritmos Particionais

- O erro quadrático para um agrupamento contendo k grupos é a soma da variação dentro dos clusters.

$$E = \sum_{j=1}^{k} \sum_{x_i \in C_j} d(x_i, \bar{x}^{(j)})^2$$

$$\bar{x}_j = \frac{1}{n_k} \sum_{x_i \in C_k} x_i$$

Algoritmos Particionais

- Exemplos:
  - K-means
  - k-means ótimo
  - k-means sequencia
  - SOM
  - CLICK

Algoritmos Particionais

- Objetivo: encontrar uma partição contendo k clusters (fixo) que minimiza E;
- Partição resultante é chamada de partição de variância mínima;
- Minimizar E é um problema NP-hard;
- Algoritmos gulosos => convergência ótimos locais;

K-means

- Algoritmo mais conhecido;
- Técnica de realocação iterativa;
- Pode convergir para um ótimo local;
- Versão tradicional: clusters compactos e com formato esférico;
- Versão com distância de mahalanobis => clusters hiperelipsoidais;

K-means

- Execução Geral
  - Inicialização (ex. aleatória) de k centróides para os clusters;
  - Cada objeto da base de dados é associado a um centróide mais próximo;
  - Centróides são recalculados de acordo com os objetos mais próximos;
  - Algoritmo para quando não há mais atualização dos centróides;
  - Complexidade: O(n)

K-means

- A atualização dos centróides pode ser realizada calculando a distância média dos objetos do grupo.
- A cada passo do algoritmo:
  - Centróides são movidos na direção dos objetos associados;
  - Algoritmo para quando não houver variação dos centróides;

K-means

- Limitações:
  - Escolha do valor para k
  - Grupos com formatos não esféricos
  - Outliers influenciam a movimentação dos clusters

Algoritmos hierárquicos

- Geram, a partir de uma matriz de proximidade, uma sequência de partições aninhadas.

- Podem ser divididos em duas abordagens:
  - Aglomerativa
  - Divisiva

Algoritmos hierárquicos

- Abordagem Aglomerativa
  - Começa com n clusters com um único objeto e forma a sequência de partições agrupando os clusters sucessivamente

- Abordagem Divisiva
  - Começa com um cluster com todos os objetos e forma a sequência dividindo os clusters sucessivamente

Algoritmos hierárquicos

- Aspectos positivos:
  - Flexibilidade com respeito ao nível de granularidade
  - Fácil utilização de qualquer forma de similaridade ou distância
  - Possibilidade de utilizar qualquer tipo de atributo

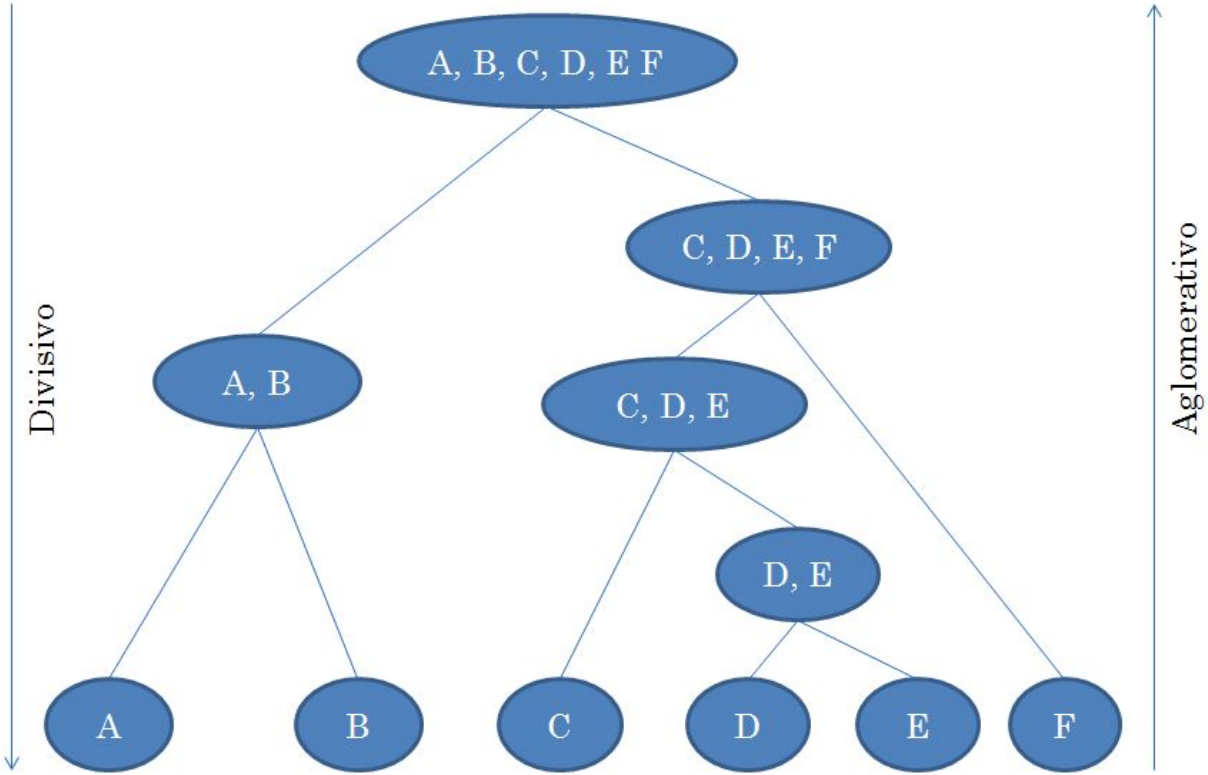
Algoritmos hierárquicos

- Aspectos negativos
  - Critério de terminação vago
  - Não há melhoria nos clusters: uma vez criado, permanece até o final

Algoritmos hierárquicos

- Abordagens clássicas utilizam métricas de integração (linkage metrics):
  - Medidas de distância entre clusters
- Técnicas aglomerativas
  - agrupam os pares de clusters mais próximos, de acordo com uma métrica de integração.
- Técnicas divisivas
  - Dividem os grupos que possam gerar partições diferentes.

Algoritmos hierárquicos



Algoritmos hierárquicos

- Vários algoritmos se baseiam na ideia de um objeto representativo que resume as informações contidas no cluster.
- Um elemento representativo bastante usado é o centróide:
  - Seja um cluster $C_k = \{x_1, x_2, \ldots, x_{nk}\}$, com n_k objetos, o centróide do cluster é dado por:

$$\bar{x}^{(k)} = \frac{1}{n_k} \sum_{x_i \in C_k} x_i$$

Algoritmos hierárquicos

- Considere dois clusters $C_1=\{x_1,x_2, \ldots, x_{n1}\}$ e $C_2=\{x_1,x_2, \ldots, x_{n2}\}$, com os respectivos centróides c^1 e c^2

- Para quantificar distâncias entre os clusters, podem ser utilizadas distâncias como **Euclidiana** ou **Manhattan** entre os centróides, ou pares de objetos dos dois clusters.

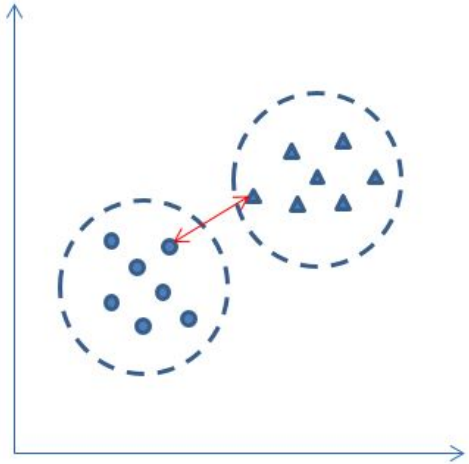
Algoritmos hierárquicos

- Métricas de integração:
  - Single-link: distância mínima entre quaisquer dois objetos, um de cada cluster.
  - Average-link: distância média entre os objetos dos dois clusters
  - Complete-link: distância entres os objetos mais distantes dos dois clusters.

Algoritmos hierárquicos

- Single-link (ligação mínima)

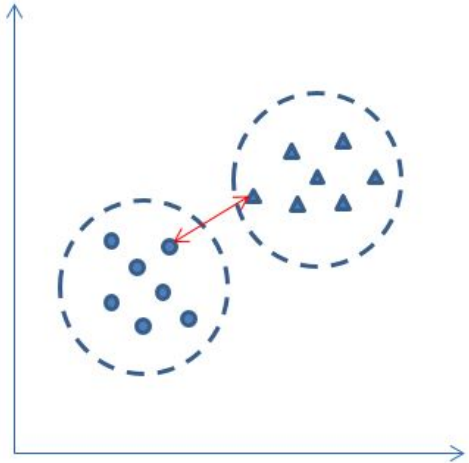
$$d(C_1, C_2) = \min_{\substack{x_i \in C_1, \\ x_j \in C_2}} d(x_i, x_j)$$



Algoritmos hierárquicos

- Single-link (ligação mínima)

$$d(C_1, C_2) = \min_{\substack{x_i \in C_1, \\ x_j \in C_2}} d(x_i, x_j)$$

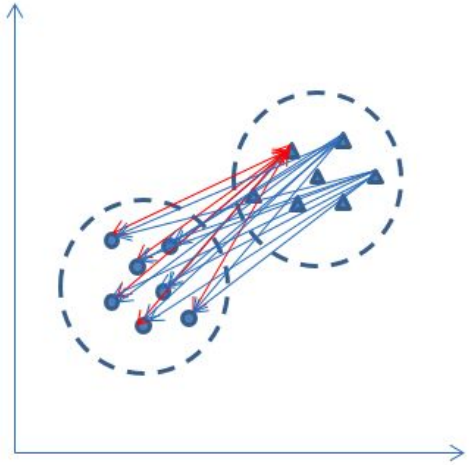


Indicados para manipular formas não elípticas, mas é bastante sensível a ruídos e outliers. Em geral, favorece clusters finos e alongados.

Algoritmos hierárquicos

- Average-link (ligação média)

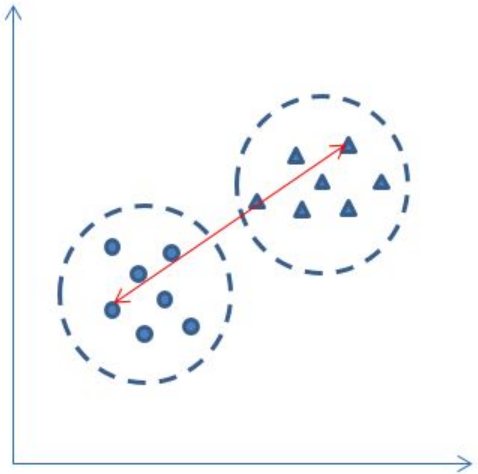
$$d(C_1, C_2) = \frac{1}{n_1 n_2} \sum_{\substack{x_i \in C_1, \\ x_i \in C_2}} d(x_i, x_j)$$



Algoritmos hierárquicos

- Complete-link (ligação máxima)

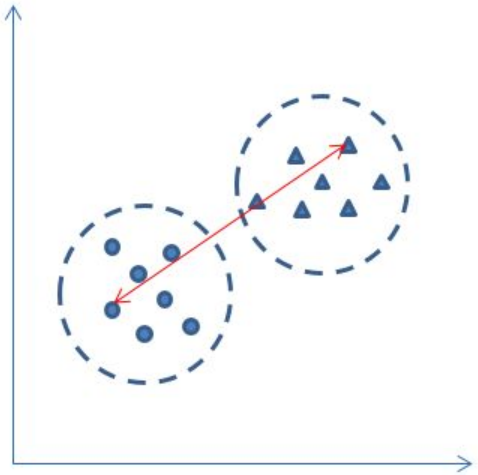
$$d(C_1, C_2) = \max_{\substack{x_i \in C_1, \\ x_j \in C_2}} d(x_i, x_j)$$



Algoritmos hierárquicos

- Complete-link (ligação máxima)

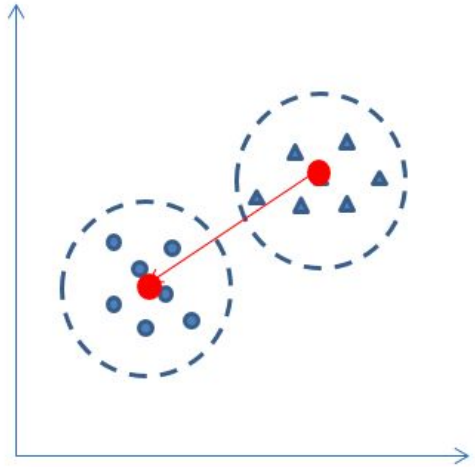
$$d(C_1, C_2) = \max_{\substack{x_i \in C_1, \\ x_j \in C_2}} d(x_i, x_j)$$



Menos suscetível a ruídos e outliers, mas tende a quebrar clusters grandes e tem problemas com formas convexas. Em geral, favorece a obtenção de clusters esféricos.

Algoritmos hierárquicos

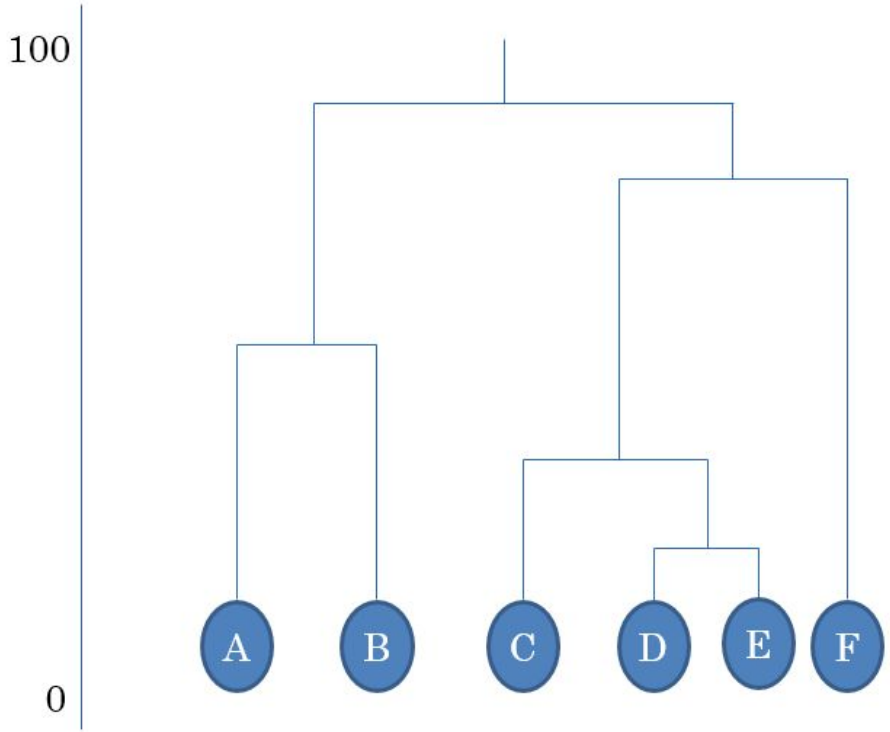
- Distância entre centróides



Algoritmos hierárquicos

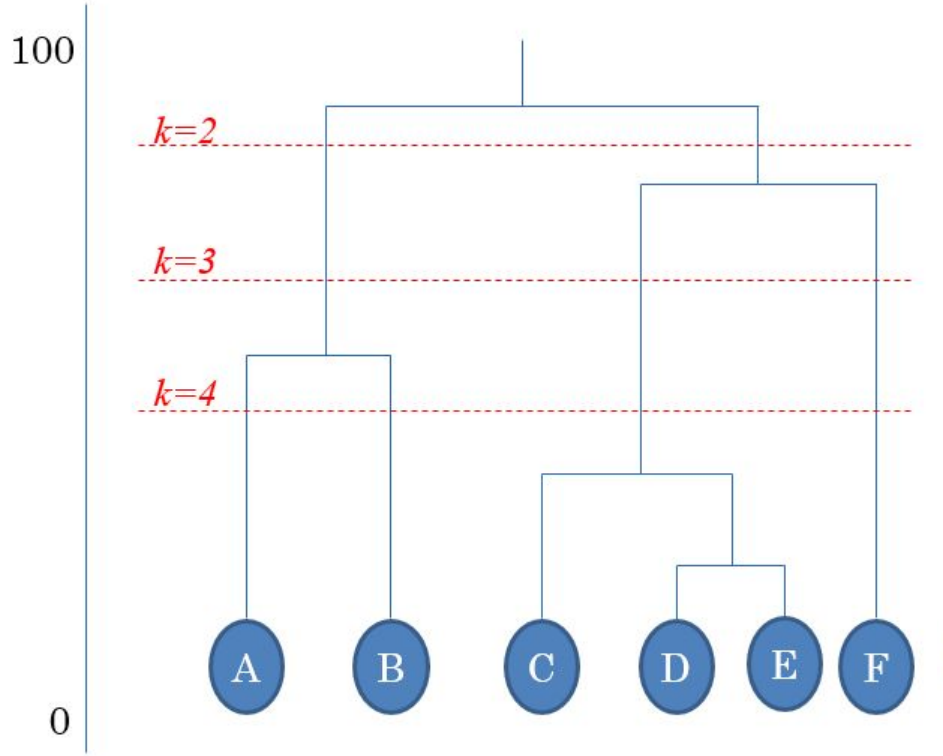
- As soluções são tipicamente representadas por um **dendograma**, que consiste em uma árvore binária que representa uma hierarquia de partições.
  - Um dendograma é formado por camadas de nós, cada uma representando um cluster.
  - Linhas conectam nós representando clusters aninhados.
  - O corte de um dendograma na horizontal representa uma partição.

Algoritmos hierárquicos



ramificações, em geral, é proporcional à distância dos clusters que foram agrupados/divididos

Algoritmos hierárquicos



Cortes em cada nível do dendograma representam diferentes partições dos dados, com diferentes números de clusters.

Referências

- Faceli et al., Inteligência Artificial – Uma Abordagem de Aprendizado de Máquina, LTC, 2015.
- Mitchell, T. M., Machine Learning, McGraw-Hill, 1997.
- Witten et al., Data Mining – Practical Machine Learning Tools and Techniques, 3d edition, Elsevier, 2011.
- Xu, R. and Wunsch, D.C., Clustering, 1a ed, Wiley, 2009
- J. Han; M. Kamber. Data Mining: Concepts and Techniques. Morgan Kaufmann, 2000.