Medidas de dissimilaridade e Escalonamento Multidimensional

Mineração de Dados

Universidade Federal do ABC

Introdução

Pré-processamento

Escalonamento Multidimensional

Introdução

- Em diversos algoritmos precisaremos de uma medida de (dis)similaridade entre objetos
  - Lembra dos k vizinhos mais próximos para estimar densidade?
- Dependendo do tipo dos dados, diversas medidas são possíveis
  - Veremos apenas as mais comuns
  - Alguns algoritmos assumem certos tipos de medidas, quando for o caso isto será ressaltado

Introdução

- Começaremos considerando medidas de **dissimilaridade**
  - Na maioria dos casos é trivial transformar os valores para similaridade

$$s = -d, s = 1 - d, s = \frac{1}{1+d}$$

Introdução

- Maioria dos algoritmos é descrito assumindo que se tem acesso à uma função de dissimilaridade/distância ou à uma matriz de dissimilaridades/distâncias

$$d(\mathbf{x}_i, \mathbf{x}_j) \text{ dissimilaridade entre o objeto } \mathbf{x}_i \text{ e } \mathbf{x}_j$$

$$D_{N,N} = \begin{pmatrix} d(\mathbf{x}_1, \mathbf{x}_1) & \cdots & d(\mathbf{x}_1, \mathbf{x}_N) \\ \vdots & \vdots & \vdots \\ d(\mathbf{x}_N, \mathbf{x}_1) & \cdots & d(\mathbf{x}_N, \mathbf{x}_N) \end{pmatrix}$$

Função de distância

- Se $d(\mathbf{a}, \mathbf{b})$ é uma métrica, esta computa a distância entre dois pontos $\mathbf{a}$ e $\mathbf{b}$, temos que:
  - $d(\mathbf{a}, \mathbf{b}) \geq 0, \forall \mathbf{a}, \mathbf{b}$
  - $d(\mathbf{a}, \mathbf{b}) = 0$, apenas se $\mathbf{a} = \mathbf{b}$
  - $d(\mathbf{a}, \mathbf{b}) = d(\mathbf{b}, \mathbf{a})$
  - $d(\mathbf{a}, \mathbf{c}) \leq d(\mathbf{a}, \mathbf{b}) + d(\mathbf{b}, \mathbf{c})$
- Algumas medidas de distância não obedecem todas as regras, neste caso, não são propriamente métricas

Função de distância Euclidiana

- A função de distância mais utilizada é a distância Euclidiana:
  - Sim, aquela que você já conhece :)
  - Seja **a** e **b** dois vetores em $\mathbb{R}^M$:

$$d_{\mathrm{EUC}}(\mathbf{a}, \mathbf{b}) = \sqrt{\sum_{m=1}^{M} (a_m - b_m)^2}$$

- Muitas vezes precisamos apenas da relação de ordem entre as distâncias
- Portanto, é comum considerar a distância Euclidiana ao quadrado

Função de distância Euclidiana

- Forma vetorial

$$d^2_{\text{EUC}}(\mathbf{a}, \mathbf{b}) = (\mathbf{a} - \mathbf{b})^T(\mathbf{a} - \mathbf{b})$$

$$d^2_{\text{EUC}}(\mathbf{a}, \mathbf{b}) = \mathbf{a}^T\mathbf{a} + \mathbf{b}^T\mathbf{b} - 2\mathbf{a}^T\mathbf{b}$$

- Em alguns contextos essa forma de descrever a distância Euclidiana é útil
  - Similaridade do coseno
  - Escalonamento multidimensional

Função de distância Manhattan

- Também chamada de *city-block* e *taxicab*

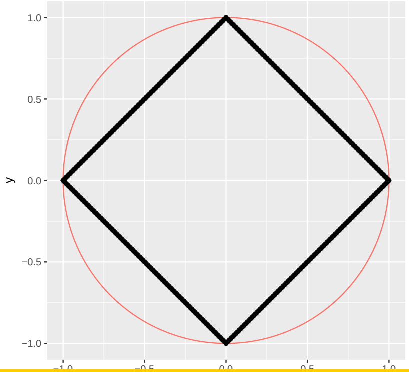
$$d_{\mathrm{MNH}}(\mathbf{a}, \mathbf{b}) = \sum_{m=1}^{M} |a_m - b_m|$$

- Pensando em cidades pode ser visto como percorrendo as ruas

Comparação entre distância Euclidiana e Manhattan

- Basta lembrar da equação do círculo e da reta

$$(x-c)^2 + (y-c)^2 = r^2$$
$$ax + by + c = 0$$



Função de distância de Mahalanobis

- Considera as variâncias e covariâncias (correlações) das variáveis
  - Σ equivale a matriz de covariância dos dados
- Aparece na distribuição Gaussiana multivariada
- Se $\Sigma = \mathbf{I}$ equivale a distância Euclidiana
  - Se Σ for diagonal (covariâncias iguais a 0) equivale a distância euclidana normalizada (pela variância)

$$d^2_{\text{MAH}}(\mathbf{a}, \mathbf{b}) = (\mathbf{a} - \mathbf{b})^T \Sigma^{-1} (\mathbf{a} - \mathbf{b})$$

Função de distância entre atributos ordinais

- O mais comum é assumir que os rankings estão em escala intervalar
  - Dessa forma, pode-se aplicar a distância Euclidiana/Manhattan nos rankings
- Por exemplo:
  - Aluno 1 = [A, B, C, A, B]; Aluno 2 = [B, B, B, B, B]
  - Rankings = [F = 0, D = 1, C = 2, B = 3, A = 4]

$$d_{MNH}(\text{A1}, \text{A2}) = |4-3| + |3-3| + |2-3| + |4-3| + |3-3| = 3$$

Função de distância entre atributos binários

- Medidas para este tipo de atributos são baseadas nas seguintes quantidades:
  - f_{00} número de atributos em que $\mathbf{x}$ e $\mathbf{y}$ são iguais a 0
  - f_{11} número de atributos em que $\mathbf{x}$ e $\mathbf{y}$ são iguais a 1
  - f_{10} número de atributos em que $\mathbf{x}$ é 1 e $\mathbf{y}$ é 0
  - f_{01} número de atributos em que $\mathbf{x}$ é 0 e $\mathbf{y}$ é 1
- Coeficiente de casamento simples:

$$s_{SMC} = \frac{f_{11} + f_{00}}{f_{01} + f_{10} + f_{11} + f_{00}}$$

$$d_{SMC} = 1 - s_{SMC}$$

Função de distância entre atributos binários

- E quando o 0 não é informativo?
  - Comparação entre matrículas de alunos
  - Comparação entre compras em um supermercado
- Coeficiente de Jaccard

$$s_J = \frac{f_{11}}{f_{01} + f_{10} + f_{11}}$$

$$d_J = 1 - J$$

- Qual o valor de d_{SMC} e d_J para $\mathbf{x} = (1, 0, 0, 0, 0)$ e $\mathbf{y} = (0, 1, 0, 0, 1)$?

Função de distância entre atributos nominais (caso geral)

- A mais utilizada é a de Casamento Simples

$$s_{CS} = \sum_{m=1}^{M} \mathbb{1}\{x_m = y_m\}$$

$$d_{CS} = M - s_{CS}$$

- Para algumas aplicações existem medidas mais úteis
  - *Edit distance*
  - *Qwerty distance*

Funções de distância

- Existem diversas medidas utilizadas para computar distância/dissimilaridade:
  - Baseadas em correlação (Pearson, Spearman, Kendall, etc)
  - Medida do coseno (muito usada em Mineração de Textos)
  - Medidas específicas para comparar imagens (*Structural Similarity*)
  - ...
- Alguns algoritmos assumem certas premissas sobre as distâncias/dissimilaridades
  - Adaptações podem ser necessárias no algoritmo
  - Pode ser necessário utilizar uma medida específica para certo algoritmo

Pré-processamento de atributos

- Considere a distância Euclidiana e os seguintes dados:
  - $\mathbf{x} = (23, 2500), \mathbf{y} = (55, 3000)$
  - Os atributos são: idade e salário
  - Os atributos tem o mesmo *peso* no cálculo da distância?
  - Como resolver?

Pré-processamento de atributos

- Seja $\mathbf{z}$ o vetor correspondente a um atributo e z um de seus valores
- Normalizar entre $[0, 1]$:

$$z' = \frac{z - min(\mathbf{z})}{max(\mathbf{z}) - min(\mathbf{z})}$$

- Transformar para média igual 0 e desvio padrão igual a 1

$$z' = \frac{z - \bar{z}}{\sigma_z}$$

- Quando usar um ou outro?

Pré-processamento de atributos

- Muitos algoritmos não aceitam atributos nominais
  - Podemos converter para um conjunto de atributos binários (representação *one-of-K*)
- Exemplos:
  - Atributo Função: {Aluno, Técnico, Docente} -> {00, 01, 10}

Pré-processamento de atributos

- Pode ser necessário o caminho inverso (contínuo -> discreto)
  - Algoritmos que criam regras podem ser mais eficientes com atributos discretos
  - Essa transformação é chamada de *discretização*
  - Existem diversas abordagens, hoje iremos falar de duas simples:
    - Largura fixa
    - Frequência fixa

Discretização - Largura fixa

- Separar o intervalo dos dados ($[min(\mathbf{z}), max(\mathbf{z})]$) em intervalos de tamanho igual especificado pelo usuário
- Exemplo:
  - Separar em intervalos de tamanho 5

$$\mathbf{z} = (32, 34, 43, 45, 51, 59, 62, 67, 68, 69, 70, 71, 72)$$

- O *bucket* que o valor da observação está corresponde ao novo valor do atributo
  - Limite inferior do primeiro *bucket* e superior do último podem ser $-\infty$ e $+\infty$

Discretização - Largura fixa

- Exemplo:
  - Separar em intervalos de tamanho 5

$$\mathbf{z} = (32, 34, 43, 45, 51, 59, 62, 67, 68, 69, 70, 71, 72)$$

$$[32, 37) = \{32, 34\} \quad [37, 42) = \{\} \quad [42, 47) = \{43, 45\}$$

$$[47, 52) = \{52\} \quad [52, 57) = \{\} \quad [57, 62) = \{\}$$

$$[62, 67) = \{62\} \quad [67, 72) = \{67, 68, 69, 70, 71\} \quad [72, 77) = \{72\}$$

Discretização - Frequência fixa

- Separar o intervalo dos dados ($[min(\mathbf{z}), max(\mathbf{z})]$) em intervalos com *aproximadamente* o mesmo número de objetos, sendo o número de intervalos especificado pelo usuário
- Exemplo, separar em 5 intervalos:
  - 13 itens, 5 intervalos $13/5 = 2.6$, logo nem todos os intervalos vão ter o mesmo número de itens

$$\mathbf{z} = (32, 34, 43, 45, 51, 59, 62, 67, 68, 69, 70, 71, 72)$$

- O *bucket* que o valor da observação está corresponde ao novo valor do atributo
- Evita que um determinado *bucket* tenha muitos itens enquanto outros ficam vazios

Discretização - Frequência fixa

- Exemplo, separar em 5 intervalos:
  - 13 itens, 5 intervalos $13/5 = 2.6$, logo nem todos os intervalos vão ter o mesmo número de itens

$$\mathbf{z} = (32, 34, 43, 45, 51, 59, 62, 67, 68, 69, 70, 71, 72)$$

$$[32, 45) = \{32, 34, 43\} \quad [45, 62) = \{45, 51, 59\}$$

$$[62, 69) = \{62, 67, 68\} \quad [69, 72) = \{69, 70, 71\} \quad [72, +\infty) = \{72\}$$

- O *bucket* que o valor da observação está corresponde ao novo valor do atributo
- Evita que um determinado *bucket* tenha muitos itens enquanto outros ficam vazios

Introdução

Pré-processamento

Escalonamento Multidimensional

Escalonamento Multidimensional

- Vimos como obter uma matriz de distância a partir de um conjunto de dados
- E se tivermos apenas a matriz de distância e quisermos visualizar os dados *de forma aproximada*
- Por que não teríamos os dados?
  - Confidencialidade
  - Dados intrinsicamente relacionais (distâncias obtidas de forma subjetiva)

Escalonamento Multidimensional

- Para este problema podem ser utilizadas técnicas de *Multidimensional Scaling*
  - Existem várias técnicas possíveis
  - Abordaremos a mais tradicional, derivada a partir de distância Euclidiana

Escalonamento Multidimensional

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_N^T \end{bmatrix}$$

$$d^2_{\mathrm{EUC}}(\mathbf{a}, \mathbf{b}) = \mathbf{a}^T\mathbf{a} + \mathbf{b}^T\mathbf{b} - 2\mathbf{a}^T\mathbf{b}$$

$$d^2_{\mathrm{EUC}}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T\mathbf{x}_i + \mathbf{x}_j^T\mathbf{x}_j - 2\mathbf{x}_i^T\mathbf{x}_j$$

$$d^2_{ij} = b_{ii} + b_{jj} - 2b_{ij}$$

$$B = \mathbf{X}\mathbf{X}^T \to b_{ij} = \sum_{m=1}^{M} x_{im}x_{jm}$$

Escalonamento Multidimensional

- Vamos assumir que temos a matriz $D_{N,N}$ mas não temos $\mathbf{X}$
  - Precisamos encontrar os valores da matriz B a partir da matriz D

Escalonamento Multidimensional

- Para restringir a solução, consideramos que os dados estão centrados na origem

$$\sum_{m=1}^{M} x_{im} = 0, \forall i \in \{1, \ldots, N\}$$

- Define-se também:

$$T = \sum_{n=1}^{N} b_{nn}$$

Escalonamento Multidimensional

- Somamos $d_{ij}^2 = b_{ii} + b_{jj} - 2b_{ij}$ sobre i, j e ambos obtendo três novas equações

$$\sum_{i=1}^{N} d_{ij}^2 = T + Nb_{jj}$$

$$\sum_{j=1}^{N} d_{ij}^2 = Nb_{ii} + T$$

$$\sum_{i=1}^{N} \sum_{j=1}^{N} d_{ij}^2 = 2NT$$

Escalonamento Multidimensional

- Podemos definir:

$$d_{\cdot j}^2 = \frac{1}{N}\sum_{i=1}^{N} d_{ij}^2$$

$$d_{i\cdot}^2 = \frac{1}{N}\sum_{j=1}^{N} d_{ij}^2$$

$$d_{\cdot\cdot}^2 = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} d_{ij}^2$$

$$b_{ij} = \frac{1}{2}(d_{r\cdot}^2 + d_{\cdot s}^2 - d_{\cdot\cdot}^2 - d_{ij}^2)$$

Escalonamento Multidimensional

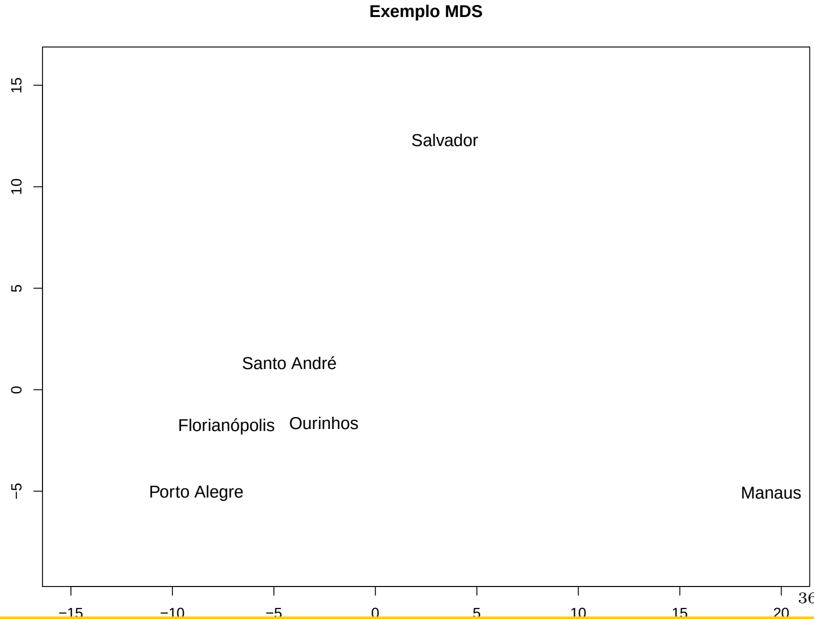
- Conseguimos obter a matriz B a partir de D
  - Lembre-se que assumimos que $B = \mathbf{X}\mathbf{X}^T$
- Podemos decompor $B = CDC^T = (CD^{1/2})(CD^{1/2})^T$, dado que ela é simétrica
  - decomposição espectral
  - C matriz de autovetores de B dispostos nas colunas
  - D matriz diagonal de autovalores de B
- Logo, podemos aproximar $\tilde{\mathbf{X}} = CD^{1/2}$
  - Se vamos visualizar os dados, usamos apenas os 2 ou 3 primeiros autovetores (autovalores)
  - $\tilde{\mathbf{X}}$ não é necessariamente igual a $\mathbf{X}$ (outro sistema de coordenadas)

Escalonamento Multidimensional

- Exemplo:
  - Base de dados com a distância (euclidiana) entre as coordenadas dos centros de 6 cidades brasileiras

Ourinhos	0.0	3.4	4.8	15.1	22.3	7.2
Santo André	3.4	0.0	4.4	13.4	24.6	7.9
Florianópolis	4.8	4.4	0.0	17.7	27.0	3.6
Salvador	15.1	13.4	17.7	0.0	23.7	21.3
Manaus	22.3	24.6	27.0	23.7	0.0	28.3
Porto Alegre	7.2	7.9	3.6	21.3	28.3	0.0

Escalonamento Multidimensional



Referências

P. Tan, M. Steinbach e V. Kumar, Introduction to Data Mining. **Seção 2.4**

E. Alpaydin, Introduction to Machine Learning. **Seção 6.5**