PUC-SP

Mineração de Dados

Prof. Dr. Daniel Rodrigues da Silva

Mineração de Dados

Agrupamento

PUC-SP

Bibliografia básica:

Introdução à mineração de dados : conceitos básicos, algoritmos e aplicações. Leandro Nunes de Castro e Daniel Gomes Ferrari. – São Paulo : Saraiva, 2016.

Inteligência Artificial - Uma Abordagem de Aprendizado de Máquina . André Carlos Ponce de Leon Ferreira et al. 2 Ed. LTC, 2024.

PUC-SP

Agrupamento

Uma das habilidades mais básicas dos organismos vivos é a capacidade de agrupar objetos similares para produzir uma *taxonomia*, uma *classificação* ou um *agrupamento*.

A ideia de organizar coisas similares em categorias, chamadas aqui de *grupos* (***clusters***), é bastante antiga e reflete a capacidade de identificar características ou combinações de características similares em alguns objetos, como forma, cor, cheiro, posição, altura, peso, entre outras.

PUC-SP

A ***Análise de grupos***, também conhecida como ***agrupamento de dados***, é um termo genérico usado para designar um amplo espectro de métodos numéricos de análise de dados multivariados com o objetivo de descobrir grupos homogêneos de objetos.

O agrupamento de objetos em diferentes grupos pode simplesmente representar uma forma conveniente de organizar grandes bases de dados de maneira que elas sejam mais facilmente compreendidas ou pesquisadas e, também, para realizar tarefas muito mais sofisticadas, como tomada de decisão em processos críticos.

PUC-SP

A ***análise de grupos*** pode, então, ser definida como a organização de um conjunto de objetos (normalmente representados por vetores de características, ou seja, pontos em um espaço multidimensional) em grupos baseada na similaridade entre eles.

Dito de outra forma, agrupar objetos é o processo de particionar um conjunto de dados em subconjuntos (grupos) de forma que os objetos em cada grupo (idealmente) compartilhem características comuns, em geral proximidade em relação a alguma medida de similaridade ou distância.

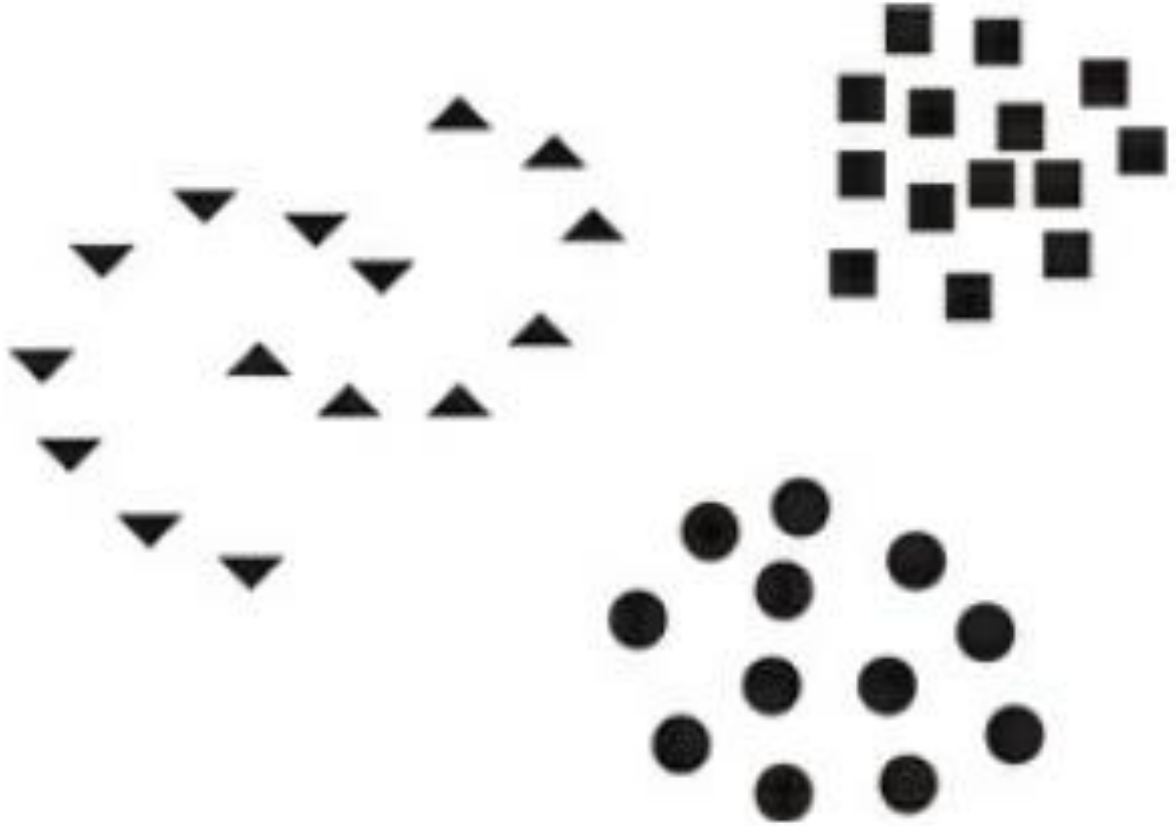
Intuitivamente, objetos pertencentes ao mesmo grupo são mais similares entre si do que a objetos pertencentes a grupos distintos.

Assim, um grupo pode ser definido em função da coesão interna (*homogeneidade*) e do isolamento externo (*separação*) de seus objetos. O conceito de *grupos naturais* foi introduzido por '*Carmichael et al.*' que postularam que tais grupos são os que satisfazem duas condições:

(1) existência de regiões contínuas do espaço, relativamente densamente populadas por objetos; e

(2) que essas regiões estão rodeadas por regiões relativamente vazias. Esse conceito pode ser facilmente ilustrado por figuras contendo conjuntos de pontos distribuídos no espaço (Ver Figura a seguir:))

Exemplos de conjuntos de pontos contendo grupos naturais



PUC-SP

O agrupamento de dados é uma técnica comum em análise de dados que é utilizada em diversas áreas, incluindo aprendizagem de máquina, mineração de dados, reconhecimento de padrões, análise de imagens e bioinformática.

Diversas outras nomenclaturas possuem significado similar, como *taxonomia numérica*, *análise de clusters*, *reconhecimento não supervisionado de padrões* e *análise tipológica*.

PUC-SP

Muitas vezes, há diferentes agrupamentos possíveis para a mesma base de dados e, portanto, a utilidade de um agrupamento depende do propósito da análise.

Por exemplo, um conjunto de carros pode ser agrupado de acordo com a cor, o consumo de combustível, o continente de fabricação, fabricante, o peso, a velocidade ou outros atributos de interesse.

A análise de grupos pode ser aplicada em diversas áreas do conhecimento, por exemplo, em:

Medicina: para a identificação de categorias de diagnósticos, pacientes e remédios;

Biologia: para propor uma taxonomia de animais e plantas;

Agricultura: para categorizar plantas, solos e frutos em diferentes tipos;

PUC-SP

Marketing: para identificar grupos de clientes, produtos e serviços;

Meteorologia: para identificar diferentes padrões climáticos;

Arqueologia: para definir relações entre diferentes tipos de objetos;

Finanças: para identificar o perfil de clientes fraudadores;

E em muitas outras áreas do conhecimento humano.

PUC-SP

Complexidade da tarefa de agrupamento

A maioria dos algoritmos de agrupamento se concentra em obter *k* grupos de objetos semelhantes de acordo com algum critério preestabelecido.

O número de possibilidades de se *classificar n* objetos em *k* grupos é dado por:

$$N(n,k) = \frac{1}{k!}\sum_{i=1}^{k}(-1)^{k-i}\left(\frac{k}{i}\right)i^{n}$$

Veja o exemplo dado ao lado.

N(*n*, *k*)		*k*	
		2	4
n	5	15	10
	10	511	34.105
	15	16.383	$4{,}25 \times 10^{7}$
	20	524.287	$4{,}52 \times 10^{10}$

Observe a influência do número de objetos *n* e de grupos *k* na quantidade de possibilidades de se agrupar os *n* objetos em *k* grupos.

Note que com um pequeno aumento nesses valores tem-se um crescimento significativo na função.

Como bases de dados com centenas, milhares e até milhões de objetos são muito comuns, esse crescimento significativo torna o problema complexo, dada a quantidade de elementos do espaço de busca envolvido.

PUC-SP

PUC-SP

Ao considerar que o valor de k é desconhecido, o número total de maneiras de se agrupar n objetos em k *grupos* é:

$$\sum_{k=1}^{n} N(n,k)$$

Como o número de separações possíveis desses n objetos em k grupos aumenta aproximadamente na razão $\frac{k^n}{k!}$, torna-se inviável computacionalmente buscar uma solução ótima global para esse problema, sobretudo para valores grandes de k e n.

Por outro lado, a escolha do valor de *k* é uma tarefa complicada, pois alguns desses valores não implicam grupos naturais.

Nesse sentido, pode-se executar o algoritmo de agrupamento diversas vezes, variando-se o valor de *k*, para depois escolher a solução cujas características se parecem melhores ou, ainda, aquelas soluções que forneçam a interpretação mais *significativa* dos dados (essa estratégia requer conhecimento de domínio, geralmente).

Uma alternativa consiste em escolher a melhor solução – valor mais adequado de *k* de acordo com algum critério numérico.

A determinação do número *ótimo* de grupos em um conjunto de dados é um dos mais difíceis aspectos do processo de agrupamento, e muitos algoritmos de busca e otimização têm sido aplicados com este objetivo

PUC-SP

O PROCESSO DE AGRUPAMENTO DE DADOS

PUC-SP

O agrupamento de dados é um processo que pode ser dividido em cinco etapas principais, como ilustrado na a seguir.

As etapas iniciais podem ser ajustadas para melhorar o agrupamento resultante utilizando o resultado (*feedback*) do próprio agrupamento.

Processo de agrupamento de dados

A etapa de pré-processamento dos dados, pode envolver todas as etapas típicas de pré-processamento de dados já comentadas , tais como limpeza, integração, redução, transformação e discretização.

As demais etapas do processo de agrupamento serão descritas em detalhes na sequência.

Aqui, é importante salientar que nenhuma técnica de agrupamento é universalmente aplicável e, além disso, diferentes técnicas podem permitir a extração de diferentes informações (agrupamentos) de uma mesma base de dados.

É essencial, portanto, que ao analisar um problema de agrupamento de dados se tenha um bom conhecimento sobre o algoritmo a ser empregado, os detalhes do processo de aquisição e pré-processamento dos dados, e o domínio do problema.

Medidas de similaridade: Os métodos de agrupamento visam agrupar objetos similares entre si e dissimilares a objetos pertencentes a outros grupos.

Assim, em geral é necessária uma medida de similaridade (proximidade) ou dissimilaridade (distância).

A maioria dos métodos de agrupamento assume, como ponto de partida, uma *matriz de dados* $\boldsymbol{X}$ de dimensão $n \; x \; m$, $\boldsymbol{X} \in \mathbb{R}^{n \; x \; m}$, que representa n objetos com m atributos cada:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{X_{11}} & \cdots & \boldsymbol{X_{1m}} \\ \vdots & \ddots & \vdots \\ \boldsymbol{X_{n1}} & \cdots & \boldsymbol{X_{nm}} \end{bmatrix}$$

PUC-SP

Outra estrutura de dados muito comum é a *matriz de* ***dissimilaridade*** **ou** ***distância***, **D**, com dimensão $n \; x \; n$, $\boldsymbol{D} \in \mathbb{R}^{\boldsymbol{n \; x \; m}}$, na qual cada elemento da matriz corresponde a uma medida quantitativa da proximidade (ou equivalentemente da distância $d(i,j) \; ou \; d_{ij}$) entre pares de objetos:

$$\boldsymbol{D} = \begin{bmatrix} \boldsymbol{0} & & & \\ \boldsymbol{d(2,1)} & \boldsymbol{0} & & \\ \boldsymbol{d(3,1)} & \boldsymbol{d(3,2)} & \boldsymbol{0} & \\ \boldsymbol{d(n,1)} & \boldsymbol{d(n,2)} & \boldsymbol{d(n,n-1)} & \boldsymbol{0} \end{bmatrix}$$

PUC-SP

Grande parte dos algoritmos de agrupamento utiliza medidas de dissimilaridade para avaliar, de modo indireto, a proximidade entre objetos.

Para isso, é preciso identificar primeiramente se a base de dados possui dados do tipo categórico, numérico ou ambos.

PUC-SP

Medidas para dados categóricos

Quando os atributos da base de dados são categóricos, medidas de similaridade são em geral empregadas.

Essas medidas costumam ser normalizadas no intervalo **[0,1]**, embora ocasionalmente também sejam expressas em valores percentuais **[0%,100%]**.

PUC-SP

Medidas de similaridade para dados binários

O tipo mais comum de dados categóricos ocorre quando todas as variáveis são binárias e a medida de distância mais comumente utilizada para este tipo de dado é a chamada **distância Hamming (*H*):**

$$H = \sum_{l=1}^{n} \delta_l$$

Sendo: $\delta_l = \begin{cases} 1 \ , & se \ x_{il} \neq x_{jl} \\ 0 \ , & caso \ contrário \end{cases}$

PUC-SP

Outras medidas de similaridade importantes nesse caso são expressas como função das diferenças entre os m atributos do objeto, por meio da chamada de ***matriz de contingência*** **ou** ***matriz de confusão.***

Exemplos dessas medidas são apresentados na babixo

	Resultado	Atributo l do objeto i		
		1	0	Total
Atributo l do objeto j	1	a	b	$a + b$
	0	c	d	$c + d$
	Total	$a + c$	$b + d$	$a + b + c + d$

Medidas de similaridade para dados binários

PUC-SP

Medida	Fórmula
S1: Coeficiente de *Matching*	$s_{ij} = (a + d) \div (a + b + c + d)$
S2: Coeficiente de Jaccard	$s_{ij} = (a) \div (a + b + c)$
S3: Rogers & Tanimoto	$s_{ij} = (a + d) \div [(a + 2(b + c) + d)]$
S4: Sokal & Sneath	$s_{ij} = a \div [a + 2(b + c)]$
S5: Gower & Legendre	$s_{ij} = (a + d) \div [a + 0{,}5(b + c) + d]$
S6: Gower & Legendre 2	$s_{ij} = a \div [a + 0{,}5(b + c)]$

PUC-SP

Observando as medidas de similaridade da Tabela acima, é possível perceber algumas de suas características e diferenças relevantes:

As medidas S1, S3 e S5 são coeficientes simétricos, no sentido de que tratam a igualdade (***matching***) positiva, *a*, e negativa, *d*, da mesma forma.

As medidas S2, S4 e S6 desconsideram a igualdade zero-zero (*d*) e, portanto, a presença de um atributo em comum entre os objetos é considerada, enquanto a ausência comum, não.

Como é usual em análise de dados, a opção por uma ou outra medida depende do contexto e, muitas vezes, de testes preliminares.

É importante salientar, entretanto, que em muitas situações práticas a simetria ou assimetria da medida é utilizada para destacar uma situação na qual os casos têm pesos distintos.

Por exemplo, em um sistema de detecção de fraude em transações de cartão de crédito, identificar uma transação normal como fraudulenta causa uma perda menor para a administradora do cartão do que autorizar uma transação fraudulenta. Nesses casos, as medidas assimétricas são úteis para que se meça de maneira diferenciada cada tipo de situação

PUC-SP

Exemplo: As medidas apresentadas para dados categóricos, serão utilizados os objetos da ***base de dados Zoo,*** sendo que o atributo pernas foi desconsiderado por não ser um atributo binário.

A Tabela a seguir apresenta os valores de similaridade entre o objeto **Cobra do Mar** e os outros objetos da amostra da base.

Analisando a medida de Hamming (H), tem-se oobjeto que representa o animal Robalo sendo o mais semelhante ao objeto Cobra do mar, diferenciando-se em apenas 3 atributos: Ovíparo, Venenoso e Barbatana.

Similaridade entre o objeto Cobra do mar e os outros objetos da amostra da base Zoo, desconsiderando o atributo Pernas

PUC-SP

Animal	H	S1	S2	S3	S4	S5	S6
Urso	7	0,53	0,42	0,36	0,26	0,70	0,26
Pato	7	0,53	0,42	0,36	0,26	0,70	0,26
Robalo	3	0,80	0,70	0,67	0,54	0,89	0,54
Sapo	5	0,67	0,58	0,50	0,41	0,80	0,41
Abelha	10	0,33	0,29	0,20	0,17	0,50	0,17
Polvo	6	0,60	0,54	0,43	0,37	0,75	0,37

Contagem dos atributos binários da comparação dos objetos Robalo e Cobra do mar

	Resultado	Cobra do mar		
		1	0	Total
Robalo	1	5	2	7
	0	1	7	8
	Total	6	9	15

PUC-SP

Medidas de similaridade para dados nominais

A medida de similaridade mais simples para dados nominais é um coeficiente de similaridade $s_{ij} = 1$ entre os objetos i e j indicando que eles são idênticos, com $s_{ij} = 0$ indicando que eles diferem maximamente para todas as variáveis.

Uma forma de calcular a dissimilaridade d_{IJ} entre dois objetos i e j é fazendo uma comparação simples atributo a atributo:

$$\boldsymbol{d_{ij} = \frac{m - M}{m}}$$

onde M é o número de *casamentos* (*matches*, ou seja, de atributos para os quais i e j possuem o mesmo valor) e m o número total de atributos.

Pesos podem ser especificados para aumentar o efeito de m ou para especificar ponderações maiores aos casamentos de atributos com um grande número de valores distintos.

PUC-SP

Utilizando a base de dados Balões para exemplificar o cálculo da medida de similaridade para dados nominais, a tabela abaixo apresenta a comparação entre os dois primeiros objetos. Utilizando a equação dada é possível determinar a dissimilaridade entre os dois objetos:

$$d_{12} = \frac{4-3}{4} = 0{,}25 \ .$$

Obj.	Cor	Tamanho	Ação	Pessoa
1	amarelo	pequeno	esticar	adulto
2	amarelo	pequeno	esticar	criança
Match	SIM	SIM	SIM	NÃO

PUC-SP

Medidas de similaridade para dados ordinais

Um atributo ordinal se assemelha a um atributo nominal, exceto pelo fato de que o atributo ordinal está ordenado por meio de algum critério.

Seja *j* um atributo de um conjunto de variáveis ordinais descrevendo *n* objetos.

O cálculo da dissimilaridade em relação à *j* envolve os seguintes passos:

Assuma que o valor de j para o i-ésimo objeto é x_{ij} e que j possui p_j valores ordenados, representando o ranking $1, \cdots, p_j$. Substitua cada x_{ij} pelo seu ranking correspondente, $r_{ij} \in \{1, \cdots, p_j\}$.

Como cada atributo ordinal pode possuir uma diferente quantidade de valores, sendo, portanto, necessário mapear o domínio de cada atributo no intervalo [0,1] de forma que todos os atributos possuam o mesmo peso. Isso pode ser feito substituindo o ranking r_{ij} do j-ésimo atributo do i-ésimo objeto por:

$$\boldsymbol{z_{ij} = \frac{r_{ij} - 1}{p_j - 1}}$$

Feito isso, a dissimilaridade entre dois objetos pode ser calculada empregando-se qualquer medida de distância entre dados contínuos a serem descritas mais adiante.

Para exemplificar o processo de conversão de um atributo ordinal para um atributo contínuo considere o atributo Pernas da base de dados Zoo. O atributo pode assumir os valores 0, 2, 4, 5, 6 e 8, sendo que será considerada a ordem crescente dos valores para conversão do atributo. Na primeira linha da Tabela a seguir têm-se os valores que os objetos podem assumir para o atributo Pernas, e a segunda linha apresenta os valores dos rankings associados aos valores originais do atributo. Ao final, utilizando a Equação dada , os novos valores do atributo são calculados e substituídos na base de dados.

Valores para conversão do atributo Pernas da base Zoo

x_{ij}	0	2	4	5	6	8
r_{ij}	1	2	3	4	5	6
z_{ij}	0,0	0,2	0,4	0,6	0,8	1,0

PUC-SP

PUC-SP

Medidas de similaridade para atributos razão

A forma mais simples de tratar um atributo do tipo razão é como se ele fosse um atributo numérico contínuo e utilizar qualquer medida de proximidade ou distância apropriada para esse tipo de atributo.

PUC-SP

Medidas de similaridade para dados contínuos:

Quando todos os atributos são contínuos, a proximidade entre objetos é tipicamente quantificada
por *métricas* ou *medidas de distância*, sendo que uma medida de distância é conside rada uma métrica se ela satisfaz a *desigualdade triangular*:

$d_{ij} + d_{il} \geq d_{jl}$, para pares de objetos (i, j) , (i, l) e (j, l).

Essencialmente, a desigualdade triangular afirma que a distância que liga dois pontos diretamente é sempre menor que a distância que liga esses dois pontos, mas passa por um ponto intermediário.

Medida	Fórmula
D1: Distância Euclidiana	$d_{ij} = \left(\sum_{k=1}^{m} (x_{ik} - x_{jk})^2 \right)^{1/2}$
D2: Distância de Manhattan	$d_{ij} = \sum_{k=1}^{m} \lvert x_{ik} - x_{jk} \rvert$
D3: Distância de Minkowski	$d_{ij} = \left(\sum_{k=1}^{m} (x_{ik} - x_{jk})^r \right)^{1/r}, (r \geq 1)$
D4: Distância de Canberra	$d_{ij} = \begin{cases} 0 & , x_{ik} = x_{jk} = 0 \\ \sum_{k=1}^{m} \lvert x_{ik} - x_{jk} \rvert \,/\, (\lvert x_{ik} \rvert + \lvert x_{jk} \rvert) & , x_{ik} \neq 0 \text{ ou } x_{jk} \neq 0 \end{cases}$
D5: Correlação de Pearson	$\phi_{ij} = \frac{\sum_{k=1}^{m} (x_{ik} - \bar{x}_i)(x_{jk} - \bar{x}_j)}{\sqrt{\sum_{k=1}^{m} (x_{ik} - \bar{x}_i)^2} \cdot \sqrt{\sum_{k=1}^{m} (x_{jk} - \bar{x}_j)^2}}$
D6: Medida do Cosseno	$\phi_{ij} = \frac{\sum_{k=1}^{m} x_{ik} x_{jk}}{\sqrt{\sum_{k=1}^{m} x_{ik}^2} \cdot \sqrt{\sum_{k=1}^{m} x_{jk}^2}}$

PUC-SP

De modo equivalente ao caso de atributos categóricos, muitas medidas de dissimilaridade ou similaridade foram propostas para a criação das matrizes de dissimilaridade de dados contínuos, como resumido na tabela ao Lado.

PUC-SP

As medidas apresentadas podem ser divididas em *medidas de distância* (D1 a D4) e *medidas tipo correlação* (D5 e D6).

A distância Euclidiana (D1) é a mais usada, pois representa a distância física entre pontos em um espaço m-dimensional.

Salientamos que a distância de *Manhattan* e a Euclidiana são casos particulares da distância *de Minkowski* para $r = 1$ e $r = 2$, respectivamente.

Tanto a correlação quanto a medida do cosseno estão definidas no intervalo $[-1, 1]$, com o valor 1 refletindo a relação mais forte possível e o valor –1 refletindo a relação mais fraca possível.

Amostra da base de dados Ruspini. A distância Euclidiana entre os objetos é calculada da seguinte forma:

$$d_{1,21} = [(4-58)^2 + (53 - 13)^2]^{1/2} = 67,20$$

$$d_{1,36} = [(4 - 28)^2 + (53 - 147)^2]^{1/2} = 97,02$$

$$d_{1,59} = [(4 - 74)^2 + (53 - 96)^2]^{1/2} = 82,15$$

$$d_{21,36} = [(58 - 28)^2 + (13 - 147)^2]^{1/2} = 137,32$$

$$d_{21,59} = [(58 - 74)^2 + (13 - 96)^2]^{1/2} = 84,53$$

$$d_{36,59} = [(28 - 74)^2 + (147 - 96)^2]^{1/2} = 68,68$$

Objeto	Atributo 1	Atributo 2
1	4	53
21	58	13
36	28	147
59	74	96

Métodos de agrupamento

PUC-SP

Há diversos algoritmos de agrupamento na literatura e a escolha de um deles depende da aplicação e dos tipos dos dados.

Considere uma base de dados $X = \{x_1, x_2, \cdots, x_n\}$, com n objetos, onde x_j, $j = 1, ..., n$, corresponde a um vetor de dados com m atributos.

Uma partição dos dados é uma coleção $C = \{C_1, C_2, \cdots, C_k\}$ de k subconjuntos, $C_i \neq \emptyset$, tal que $C_1 \cup C_2 \cup \cdots C_k = X$.

De forma abrangente os métodos de agrupamento podem ser divididos em

Hierárquicos: os métodos hierárquicos criam uma decomposição hierárquica dos dados. Esses métodos podem ser *aglomerativos* ou *divisivos*, baseados em como o processo de decomposição é efetuado.

Algoritmos Particionais Baseados em Erro Quadrático : dado um conjunto com n objetos, um método particional constrói k partições dos dados, sendo que cada partição representa um *cluster* ($k \leq n$).

Dado o número k de partições, um método particional cria uma partição inicial e emprega um *algoritmo de realocação iterativa* que tem por objetivo melhorar o particionamento movendo objetos entre grupos.

PUC-SP

Representação dos grupos: É o processo de extrair uma representação simples e compacta dos grupos obtidos a partir do agrupamento da base.

As formas típicas de representação dos grupos são (Veja Figura abaixo):

Protótipos: correspondem a vetores representativos dos grupos, por exemplo, um *centroide* (ponto médio) de um grupo. Outro exemplo é o caso de um elemento típico de um grupo, como uma banana que representa todo o grupo de bananas e uma maçã que representa todo o grupo de maçãs.

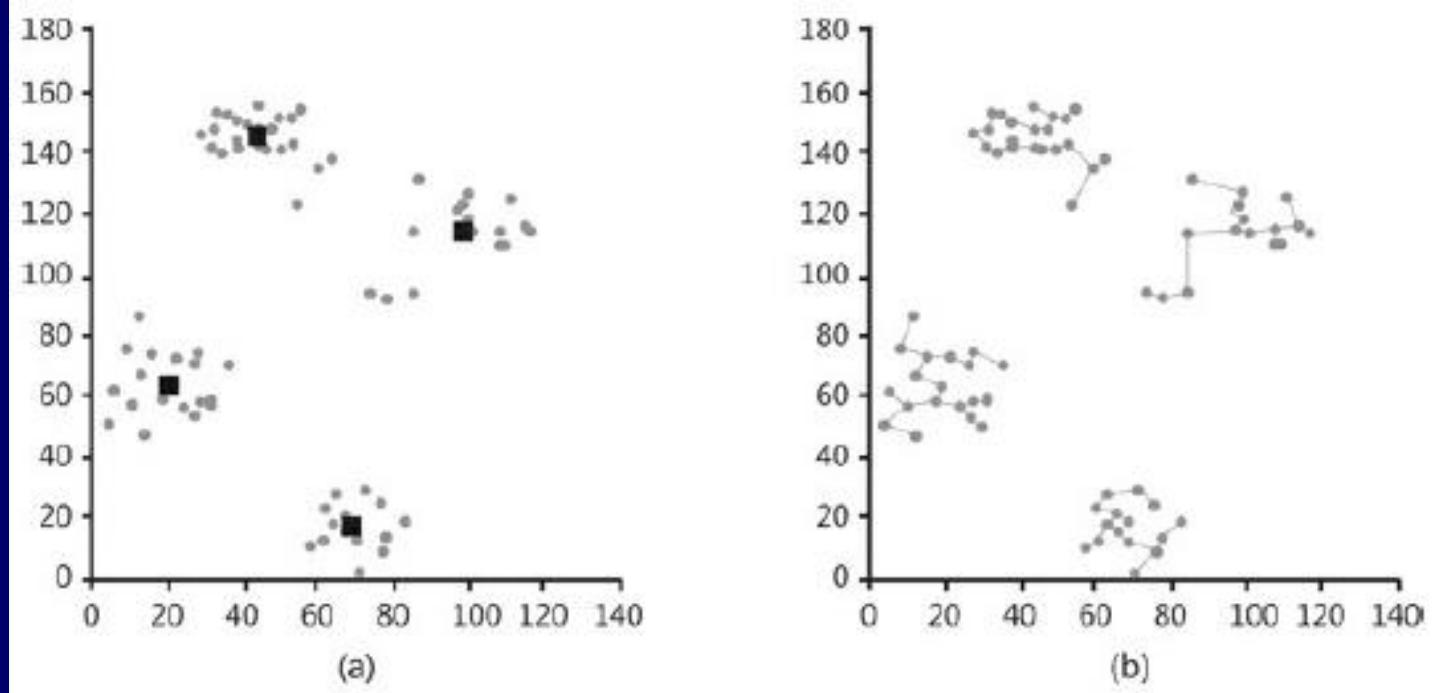
PUC-SP

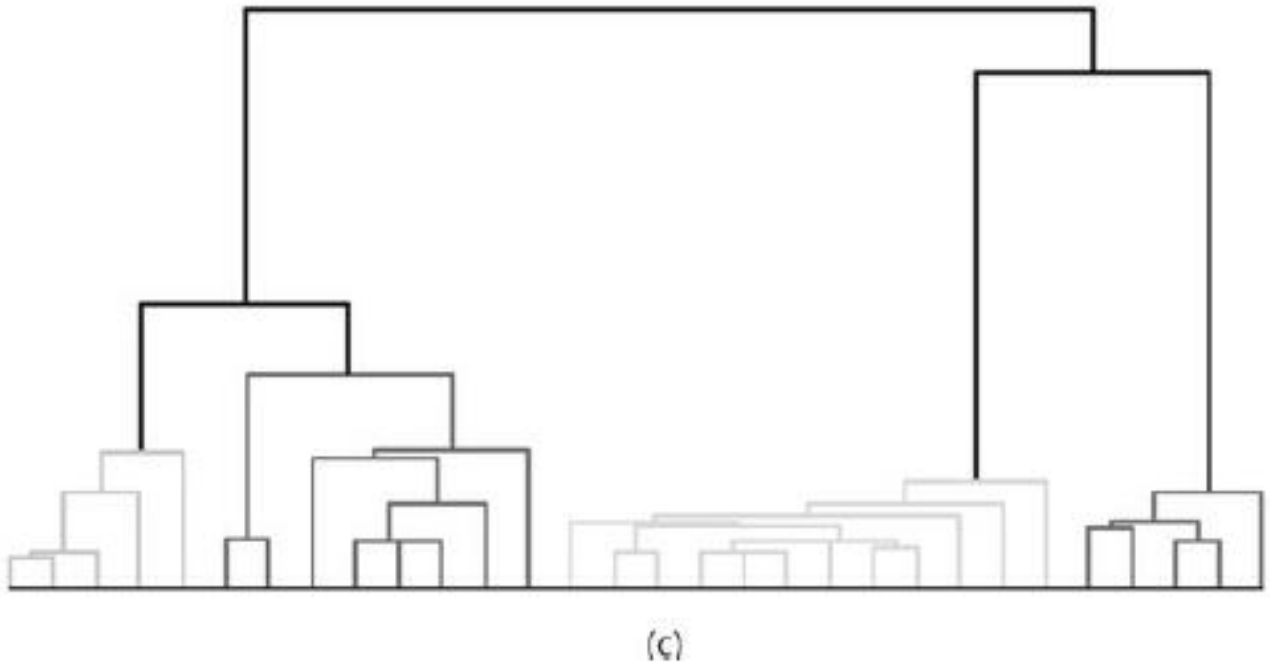
Estruturas em grafo: neste contexto, um grafo é um conjunto de nós e arcos no qual os nós correspondem aos objetos da base e os arcos, às conexões entre eles.

Estruturas em árvore: esse tipo de estrutura, como os dendrogramas, é um caso particular dos grafos que fornece uma representação hierárquica das relações entre os objetos e os grupos encontrados.

Em uma estrutura em árvore normalmente se escolhe um ponto da árvore para se realizar a partição dos grupos.

PUC-SP





PUC-SP

Base Ruspini:

(a) Protótipos (quadrados nos centros dos grupos).

(b) Grafo formado pelos subgrafos conectando os pontos de cada grupo.

(c) Árvore, cujas ramificações representam os grupos

Avaliação do agrupamento

PUC-SP

A avaliação da saída de um algoritmo de agrupamento depende do contexto e dos objetivos da análise. Por exemplo, em uma análise exploratória de uma base de dados de imóveis, objetos pertencentes ao mesmo grupo podem permitir a identificação de perfis de moradores de determinado bairro.

A saída do algoritmo de agrupamento também pode ser avaliada com relação à *qualidade do agrupamento*, o que pode ser feito por uma medida de *avaliação externa* – isto é, os grupos encontrados são comparados com uma estrutura de agrupamento conhecida *a priori* ou uma *avaliação interna*, ou seja, tenta-se determinar se a estrutura encontrada pelo algoritmo é apropriada aos dados.

Existem dois critérios para avaliação e seleção de um agrupamento de qualidade:

Compactação: os objetos de cada grupo devem estar o mais próximo possível um dos outros.

As medidas utilizadas para calcular a compactação de um grupo são geralmente denominadas de *intragrupo*.

Separação: os grupos devem estar o mais distante possível uns dos outros. As medidas para cálculo da separação entre grupos são normalmente denominadas *intergrupos*.

PUC-SP

Os grupos formados pelos objetos de uma base de dados podem ser avaliados por dois tipos de medidas:

Internas: são medidas que utilizam apenas informações intrínsecas aos objetos do agrupamento, baseando-se em medidas de similaridade e avaliando as distâncias intragrupos e/ou intergrupos.

Externas: são medidas que avaliam quão correto está um agrupamento dado um agrupamento ideal que se deseja alcançar.

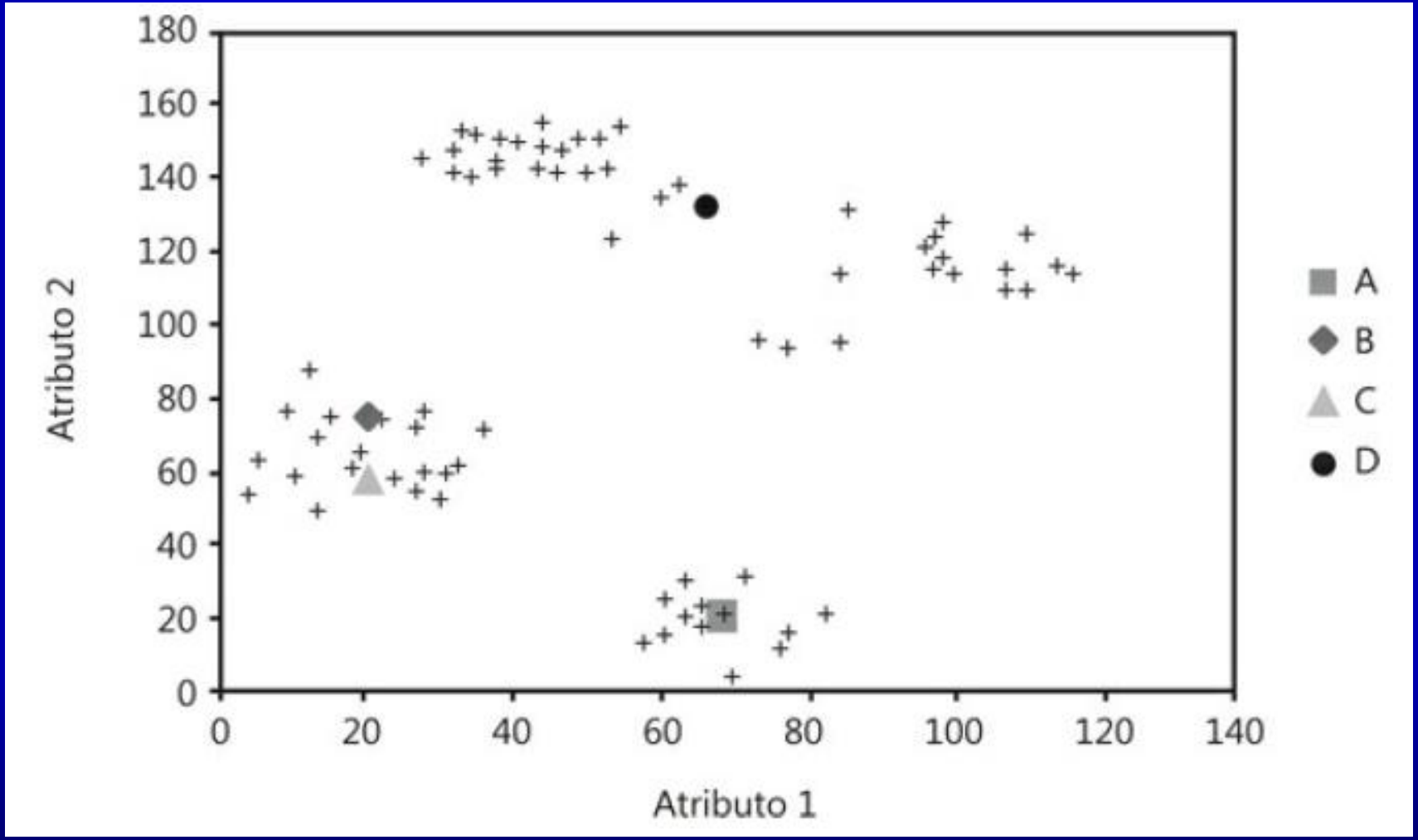
O cálculo dessas medidas requer o conhecimento prévio do grupo ao qual cada objeto pertence.

PUC-SP

Para ilustrar a avaliação de qualidade de um agrupamento com base nas medidas que serão descritas, considere a base de dados Ruspini agrupada por um método particional, utilizando a distância Euclidiana como medida de similaridade, que propõe os protótipos apresentados na Tabela a seguir e ilustrados na Figura seguinte para representar os grupos encontrados.

Protótipo	Atributo 1	Atributo 2	Objetos
A	68,93	19,40	21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35
B	20,25	75,38	3, 5, 7, 8, 11, 14, 16, 20
C	20,08	58,00	1, 2, 4, 6, 9, 10, 12, 13, 15, 17, 18, 19
D	66,98	132,80	36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75

Base de dados Ruspini com cada grupo representado por um protótipo distinto (os símbolos à direita representam os protótipos dos grupos)



ALGORITMOS DE AGRUPAMENTO

PUC-SP

Agora, descreveremos os algoritmos mais conhecidos para agrupamento de dados, contemplando diferentes abordagens do problema.

Os algoritmos particionais mais usados são:

- *k*-médias (*k-means*)

- *k*-medoides (*k-medoids*)

e variações de ambos.

ALGORITMOS DE AGRUPAMENTO

PUC-SP

A maioria dos algoritmos hierárquicos são variações dos métodos mais populares dessa categoria:

- *single-link*
- *complete-link*.

Também serão descritos algoritmos baseados em densidade (DBSCAN), em grafos (MST) e em particionamento não exclusivo (*fuzzy k*-médias).

Implementação do K-Means


```
# processamento de dados, algebra linear
import numpy as np
import pandas as pd

# visualização de dados
import seaborn as sns
import matplotlib.pyplot as plt

# Normalização
from sklearn.preprocessing import StandardScaler
from sklearn import preprocessing

# K-Means
from sklearn.cluster import KMeans

# Métrica Silhouette mede a eficiência da escolha do valor k
#(quantidade de grupos)
from sklearn.metrics import silhouette_score

import warnings
warnings.filterwarnings("ignore") # Ignorar Avisos
```

PUC-SP

```
df1 = pd.read_csv('clientes-shopping.csv')
df1.head(3)

sns.scatterplot(data=df1, x="Annual Income (k$)", y="Spending Score (1-100)", hue="Gender")
plt.show()


# Remove colunas não numéricas: id, gênero e idade
df2 = df1.drop(['CustomerID', 'Gender', 'Age'], axis = 1)
df2.head(3)


# Normaliza valores por coluna
norma = preprocessing.MinMaxScaler()
df2 = norma.fit_transform(df2)


modelo = KMeans(n_clusters = 5)
modelo.fit(df2) # treina o modelo com os dados passados
df1['Cluster'] = modelo.predict(df2) # Cria coluna 'Cluster' com a função predict
df1.sample(5)
```

PUC-SP

```
# Exemplo pegando só o grupo 0
df3 = df1[df1['Cluster'] == 0]
df3.head(3)
```

```
# Mostra os grupos
sns.scatterplot(data=df1, x="Annual Income (k$)", y="Spending Score (1-100)", hue='Cluster', style='Cluster', palette='tab10')
plt.show()
```

```
# Mostra as estatísticas descritivas para todos os clusters
val = df1.drop(['CustomerID', 'Gender', 'Age'], axis = 1).groupby('Cluster').describe()
val
```

```
# Mostrar as coordenadas dos centroides.
modelo.cluster_centers_
```

PUC-SP

```
# Silhueta para a quantidade de cluster escolhida
float(silhouette_score(df2, df1['Cluster']))

# Silhueta para 10 valores de k
for i in range(2, 11):
    cluster = KMeans(n_clusters = i)
    pred = cluster.fit_predict(df2)
    score = silhouette_score(df2, pred)
    print('Silhueta para ' + str(i) + ' clusters : ' + str(score))
```

PUC-SP

```
#Determinando a quantidade ideal de cluster
w = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters = i, max_iter = 300, n_init = 10, random_state = 0)
    kmeans.fit(df2)
    w.append(kmeans.inertia_)

# Mostra o Gráfico
plt.plot(range(1, 11), w)
plt.title('Curva de Cotovelo')
plt.xlabel('Numero de Clusters')
plt.ylabel('W') #within cluster sum of squares
plt.show()
```

Conclusões do k-Means

PUC-SP

- k-Means particionou os dados em 5 grupos distintos, com as seguintes características:

- Grupo A - Recebe pouco dinheiro, e tem score baixo.

- Grupo B - Recebe muito dinheiro, e tem score alto.

- Grupo C - Recebe muito dinheiro, e tem score baixo.

- Grupo D - Recebe pouco dinheiro, e tem score alto.

- Grupo E - Recebe médio dinheiro, e tem score médio.