



Agenda

- Conceitos preliminares e classificador 1R (aquecimento)
- Classificadores Bayesianos
- Árvores de classificação
- k-Nearest Neighbors (k-NN)
- Classifier ensembles
- Random forests
- Noções sobre SVMs
- Redes Neurais
- Avaliação de classificadores

Agenda

- **Conceitos preliminares e classificador 1R (aquecimento)**
- Classificadores Bayesianos
- Árvores de classificação
- k-Nearest Neighbors (k-NN)
- Classifier ensembles
- Random forests
- Noções sobre SVMs
- Redes Neurais
- Avaliação de classificadores

Conceitos preliminares

- **Tarefa:** dado um conjunto de **exemplos** pré-classificados (rotulados), induzir (aprender) um modelo (classificador) para novos casos.
- **Aprendizado supervisionado:** classes são conhecidas para os exemplos usados para construir o modelo (classificador).
- Um classificador pode ser um modelo de regressão logística, um conjunto de regras lógicas, uma árvore de decisão, um modelo Bayesiano, uma rede neural etc.
- **Aplicações típicas:** aprovação de crédito, marketing direto, detecção de fraude etc.

Exemplo 1 – Dados tabulares

Outlook	Temperature	Humidity	Windy	Play
sunny	85	85	false	no
sunny	80	90	true	no
overcast	83	86	false	yes
rainy	70	96	false	yes
rainy	68	80	false	yes
rainy	65	70	true	no
overcast	64	65	true	yes
sunny	72	95	false	no
sunny	69	70	false	yes
rainy	75	80	false	yes
sunny	75	70	true	yes
overcast	72	90	true	yes
overcast	81	75	false	yes
rainy	71	91	true	no
rainy	63	84	true	?

Weather Data*
Considerando-se dados históricos, construir um modelo para os valores do atributo meta play.

Exemplo 2 – Classificação de imagens

$raz\tilde{a}o\ de\ aspecto = \frac{comprimento}{largura}$

img	cor	r. a.
1	amarelo	0.30
2	amarelo	0.27
3	amarelo	0.45
4	vermelho	1.01
5	vermelho	0.99
6	vermelho	1.10
7	vermelho	1.07

Aquecimento: 1R

Aprende uma árvore de decisão de um nível.

- Todas as regras usam somente um atributo.

Versão Básica:

- Um ramo para cada valor do atributo;
- Para cada ramo, atribuir a classe mais frequente;
- Taxa de erro de classificação: proporção de exemplos que não pertencem à classe majoritária do ramo correspondente;
- Escolher o atributo com a menor taxa de erro de classificação;

- Aplicação imediata para atributos nominais/categóricos/binários;
- Para atributos ordinais/contínuos há vários algoritmos de discretização para definir estratégias de corte nos valores dos atributos (<=, <, >, >=).

Algoritmo 1R

Para cada atributo:

Para cada valor do atributo gerar uma regra:

- Contar a frequência de cada classe;
- Encontrar a classe mais frequente;
- Formar uma regra que atribui à classe mais frequente este atributo-valor;
- Calcular a taxa de erro de classificação das regras;
- Escolher as regras com a menor taxa de erro de classificação.

Exemplo para a base Weather

Outlook	Temp.	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

Atributo	Regras	Erros	Total erros
Outlook	Sunny → No Overcast → Yes Rainy → Yes	2/5 0/4 2/5	4/14
Temp.	Hot → No* Mild → Yes Cool → Yes	2/4 2/6 1/4	5/14
Humidity	High → No Normal → Yes	3/7 1/7	4/14
Windy	False → Yes True → No*	2/8 3/6	5/14

* empate

Witten&Frank

Algoritmo 1R

- 1R foi descrito por Holte (1993):
 - Avaliação experimental em 16 bases de dados;
 - Em muitos *benchmarks*, regras simples não são muito piores do que árvores de decisão mais complexas.
- Fácil implementação;
- Muito usado para análise exploratória de dados;
- Árvores de Decisão estendem essa ideia;
- Mas antes de abordá-las, abordaremos um algoritmo eficaz e eficiente: Naive Bayes.

Holte, Robert C. Very Simple Classification Rules Perform Well on Most Commonly Used Datasets, *Machine Learning* 11 (1), pp. 63-90, 1993.

Agenda

- Conceitos preliminares e classificador 1R (aquecimento)
- Classificadores Bayesianos
- Árvores de classificação
- k-Nearest Neighbors (k-NN)
- Classifier ensembles
- Random forests
- Noções sobre SVMs
- Redes Neurais
- Avaliação de classificadores

13

Classificador Bayesiano

- Contrariamente ao 1R, o Naive Bayes (NB) usa todos os atributos.
- Presume que os atributos são igualmente importantes e condicionalmente independentes.
 - Valor de um atributo não influencia no valor de outro atributo, dada a informação da classe;
 - Válido para a nossa base de dados (Weather)?
- Na prática, tais premissas são frequentemente violadas, mas ainda assim o NB é muito competitivo:
 - Probabilidades estimadas não precisam necessariamente ser corretas, o que importa são as avaliações relativas.
- Parece haver consenso que, na prática, deve ser o primeiro algoritmo a testar.

14

Naive Bayes - Discussão

- Naive Bayes funciona bem mesmo quando suas premissas são violadas;
- Classificação não requer estimativas precisas da probabilidade, desde que a máxima seja atribuída à classe correta*;
- Entretanto, a existência de muitos atributos redundantes pode causar problemas → selecionar melhores atributos;
- Muitos atributos numéricos não seguem uma distribuição *Gaussiana* (→ GMM, *kernel density estimators* etc.);

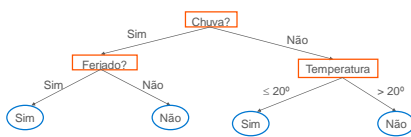
* Domingos & Pazzani, On the Optimality of the Simple Bayesian Classifier under Zero-One Loss, Machine Learning 29, 103-130, 1997.

Agenda

- Conceitos preliminares e classificador 1R (aquecimento)
- Classificadores Bayesianos
- **Árvores de classificação**
- k-Nearest Neighbors (k-NN)
- Classifier ensembles
- Random forests
- Noções sobre SVMs
- Redes Neurais
- Avaliação de classificadores

Árvores de classificação

- Aproximam funções discretas, representadas por uma árvore de decisão;
- Permite tradução via regras "se...então" → *interpretabilidade*;
 - Nós internos: teste em atributos *previsores*;
 - Nós externos: valor previsto para o atributo *meta*;



➤ Veremos conceitos do ID3 (Quinlan, 1986) e do C4.5 (Quinlan, 1993).

Ideia geral

1. Árvore é construída de maneira *top-down*, *recursivamente* e usando a ideia de *dividir para conquistar*;
2. Inicialmente, todos os exemplos de treinamento são *posicionados* na raiz da árvore;
3. Exemplos são *particionados* recursivamente com base em atributos selecionados, objetivando-se separar os exemplos por classes;
4. Condições de parada:
 - Todos os exemplos para um dado nó pertencem à mesma classe;
 - Não existem mais atributos para continuar o *particionamento*;

Ideia geral com medida intuitiva de pureza

- Como obter uma árvore de decisão para a seguinte base de dados?

SEXO	PAÍS	IDADE	COMPRAR
M	França	25	Sim
M	Inglatera	21	Sim
F	França	23	Sim
F	Inglatera	34	Sim
F	França	30	Não
M	Alemanha	21	Não
M	Alemanha	20	Não
F	Alemanha	18	Não
F	França	34	Não
M	França	55	Não

➤ Considerando que "COMPRAR" é nosso atributo-meta (classe) ...

Exemplo inspirado em Freitas & Lavington, Mining Very Large Databases with Parallel Processing, Kluwer, 1996.

Árvores de classificação – Exemplo

Para cada atributo predictor (SEXO, PAÍS, IDADE), montar uma tabela:

- Linhas: atributo predictor (variável independente);
- Colunas: atributo-meta (variável dependente);
- Células: nº de tuplas para a combinação de valores atributo-classe.

- Qual atributo discrimina melhor: SEXO ou PAÍS?

SEXO	Classe	
	Sim	Não
M	2	3
F	2	3

PAÍS	Classe	
	Sim	Não
França	2	3
Inglatera	2	0
Alemanha	0	3

Se SEXO=F então Classe=Não; Senão Classe=Não. ⇒ Acurácia (A) = 60%

➤ Mas regra default (atribuir sempre CLASSE=Não) retorna A=60%!

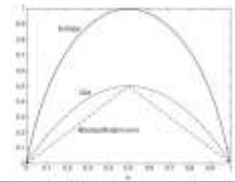
Se PAÍS=Inglatera então Sim; Senão Não ⇒ A = 80%.

O que dizer sobre o atributo IDADE? Como medir a informação?

Medindo impureza via Entropia

- Permite medir a *informação* fornecida por cada atributo;
- Caracteriza a impureza de um conjunto de exemplos. Para um nó da árvore com *p* exemplos positivos e *n* exemplos negativos temos.

$$entropia = -\frac{p}{p+n} \log_2 \left(\frac{p}{p+n} \right) - \frac{n}{p+n} \log_2 \left(\frac{n}{p+n} \right)$$



Voltemos ao exemplo anterior...

Entropia – Exemplo

Considerando o atributo meta "comprar" e 10 exemplos no nó raiz (4+,6-):

- Entropia: **E = 0,97** sendo **P(+)=0,4** e **P(-)=0,6**
- Para cada atributo teremos um valor de E. Para o atributo SEXO:

SEXO	Classe (COMPRAR)	
	Sim	Não
M	2	3
F	2	3

Ponderando o valor de E pelo número de exemplos que apresentam determinado valor para o atributo temos:

$$E(\text{SEXO}) = \frac{5}{10} (-2/5 \log_2 2/5 - 3/5 \log_2 3/5) + \frac{5}{10} (-2/5 \log_2 2/5 - 3/5 \log_2 3/5) \quad (M)$$
$$E(\text{SEXO}) = 0,97.$$

Ganho de Informação (GI) = 0,97 - 0,97 = 0,00.

➤ Não há ganho de informação ao particionar com base no SEXO.

Entropia – Exemplo

- Consideremos o atributo PAÍS:

PAÍS	COMPRAR		Total
	Sim (+)	Não (-)	
França	2	3	5
Inglatera	2	0	2
Alemanha	0	3	3
Total	4	6	10

$E(\text{PAÍS}) = \frac{5}{10} \cdot \text{Info}(\text{França}) + \frac{2}{10} \cdot \text{Info}(\text{Inglatera}) + \frac{3}{10} \cdot \text{Info}(\text{Alemanha}) = \frac{5}{10} (-\frac{2}{5} \log_2 \frac{2}{5} - \frac{3}{5} \log_2 \frac{3}{5}) + \frac{2}{10} (-\frac{2}{2} \log_2 \frac{2}{2} - \frac{0}{2} \log_2 \frac{0}{2}) + \frac{3}{10} (-\frac{0}{3} \log_2 \frac{0}{3} - \frac{3}{3} \log_2 \frac{3}{3}) = 0,485$

$E(\text{PAÍS}) = 0,485$

- Ganho de Informação (GI) / Redução na Entropia: $(0,97 - 0,485) = 0,485$.
- PAÍS (A=80%) é um atributo melhor do que SEXO (A=60%).

Lidando com atributos contínuos

ID Registro	Cabelo	Peso	Idade	Classe
Homer	0"	250	36	M
Marge	10"	150	34	F
Bart	2"	90	10	M
Lisa	6"	78	8	F
Maggie	4"	20	1	F
Abe	1"	170	70	M
Selma	8"	160	41	F
Otto	10"	180	38	M
Krusty	6"	200	45	M

Como lidar???

Lidando com atributos contínuos - Exemplo

Diagram illustrating a decision tree split on "Cabelo < 5?".

Initial Entropy: $Entropia(4F,5M) = -(\frac{4}{9})\log_2(\frac{4}{9}) - (\frac{5}{9})\log_2(\frac{5}{9}) = 0,9911$

Left branch (sim): $Entropia(1F,3M) = -(\frac{1}{4})\log_2(\frac{1}{4}) - (\frac{3}{4})\log_2(\frac{3}{4}) = 0,8113$

Right branch (não): $Entropia(3F,2M) = -(\frac{3}{5})\log_2(\frac{3}{5}) - (\frac{2}{5})\log_2(\frac{2}{5}) = 0,9710$

GI (Cabelo <= 5) = $0,9911 - (\frac{4}{9} * 0,8113 + \frac{5}{9} * 0,9710) = 0,0911$

Lidando com atributos contínuos - Exemplo

Diagram illustrating a decision tree split on "peso <= 160?".

Initial Entropy: $Entropia(4F,5M) = -(\frac{4}{9})\log_2(\frac{4}{9}) - (\frac{5}{9})\log_2(\frac{5}{9}) = 0,9911$

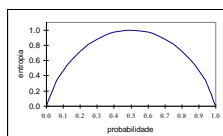
Left branch (sim): $Entropia(4F,1M) = -(\frac{4}{5})\log_2(\frac{4}{5}) - (\frac{1}{5})\log_2(\frac{1}{5}) = 0,7219$

Right branch (não): $Entropia(0F,4M) = -(\frac{0}{4})\log_2(\frac{0}{4}) - (\frac{4}{4})\log_2(\frac{4}{4}) = 0$

GI (peso <= 160) = $0,9911 - (\frac{5}{9} * 0,7219 + \frac{4}{9} * 0) = 0,5900$

> "Peso" discrimina melhor do que "Cabelo"...

Árvores de classificação - discussão



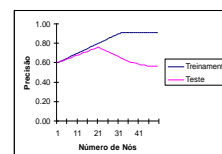
- Lembrando que $\log_2 1 = 0$ e definindo $\log_2 0 = 0$;
- Para problemas em que há "c" classes temos:

$$entropia = - \sum_{i=1}^c p_i \log_2 p_i$$

Onde p_i denota a probabilidade da classe "i".

Árvores de classificação – vantagens e desvantagens

- Compreensibilidade / facilidade para gerar regras;
- Possibilidade de *super-ajuste* (erros, ruído, poucos dados):



Definição: Uma hipótese $h \in H$ *super-ajusta* os dados de treinamento se existe uma alternativa $h' \in H$ tal que h apresenta um erro menor do que h' no conjunto de treinamento mas um erro maior na distribuição completa de exemplos.

Árvores de classificação – vantagens e desvantagens

- Procedimentos de poda:
 - conjunto de validação;
 - eliminar antecedentes das regras obtidas a partir da árvore;
- GI tem um *bias* (tendência, preferência) que favorece a escolha de atributos com muitos valores;
- Para minimizar/superar limitações:
 - procedimentos de poda;
 - outros critérios de escolha de atributos;
 - seleção de atributos *a priori*.

Agenda

- Conceitos preliminares e classificador 1R (aquecimento)
- Classificadores Bayesianos
- Árvores de classificação
- **k-Nearest Neighbors (k-NN)**
- Classifier ensembles
- Random forests
- Noções sobre SVMs
- Redes Neurais
- Avaliação de classificadores

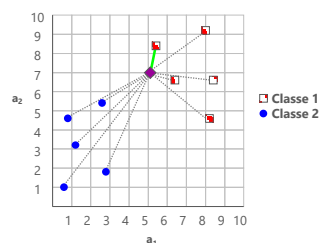
Lazy algorithms

- Não constroem descrições gerais e explícitas (função alvo) a partir dos exemplos de treinamento;
- Generalização é *adiada* até o momento da classificação;
- Armazena-se uma base de exemplos (*instances*) que é usada para realizar a classificação de uma nova *query* (exemplo *não visto*);
- Em muitos casos apresenta um alto custo computacional (por conta do cálculo de distâncias).



43

Noção Intuitiva (1-NN)



Fonte: Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBDO 2003, Manaus.

44

Conceitos fundamentais

- Exemplos correspondem a pontos no \mathbb{R}^n ;
- Vizinhos definidos em função de uma medida de distância;
- Por exemplo, considerando-se dois vetores $\mathbf{x}=[x_1, x_2, \dots, x_n]$ e $\mathbf{y}=[y_1, y_2, \dots, y_n]$, a distância Euclidiana é:

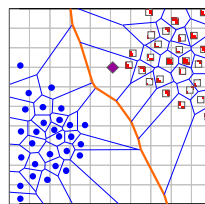
$$d_E(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

➤ Classificação por meio da classe majoritária da vizinhança.



45

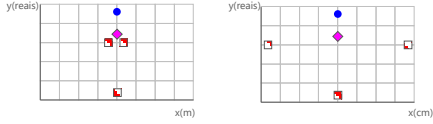
Superfície de decisão



Fonte: Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBDO 2003, Manaus.

46

Sensibilidade em relação à escala



Como diminuir este problema?

- Normalizações (linear, escore-z)
- Transformações
- Testar via validação-cruzada


Fonte: Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community SBBD 2003, Manaus.

Como lidar com atributos nominais?

Mudar a função de distância – e.g., usando coeficiente de casamento simples (*simple matching*):

$$d_{SM}(x, y) = \sum_{i=1}^n s_i \quad \begin{aligned} (x_i = y_i) &\Rightarrow s_i = 0 \\ (x_i \neq y_i) &\Rightarrow s_i = 1 \end{aligned}$$

- Há várias outras medidas de distância (e.g., ver Kaufman & Rousseeuw, Finding Groups in Data, 1990);
- Como lidar com bases de dados formadas por diferentes tipos de atributos (ordinais, contínuos, nominais, binários)?



Chega de democracia: ponderando os votos

Função alvo discreta:

$$f(x_q) \leftarrow \arg \max_{v \in V} \sum_{i=1}^k w_i \delta(v, f(x_i)) \quad \begin{aligned} (a = b) &\Rightarrow \delta(a, b) = 1 \\ (a \neq b) &\Rightarrow \delta(a, b) = 0 \end{aligned}$$

Função alvo contínua (regressão):

$$y = f(x_q) = \frac{\sum_{i=1}^k w_i f(x_i)}{\sum_{i=1}^k w_i}$$

Ponderação:

$$w_i = \frac{1}{d(x_q, x_i)}$$

Exercício

Considere a seguinte base de dados:

Instância	a ₁	a ₂	a ₃	Classe
1	0	250	36	A
2	10	150	34	B
3	2	90	10	A
4	6	78	8	B
5	4	20	1	A
6	1	170	70	B
7	8	160	41	A
8	10	180	38	B
9	6	200	45	?

Perguntas:

- a) Qual é a função de distância a ser empregada?
- b) Como escolher *k*?
- c) Teremos problemas se usarmos os dados da forma como estão?

Agenda

- Conceitos preliminares e classificador 1R (aquecimento)
- Classificadores Bayesianos
- Árvores de classificação
- k-Nearest Neighbors (k-NN)
- **Classifier ensembles**
- Random forests
- Noções sobre SVMs
- Redes Neurais
- Avaliação de classificadores



53

Intuição

Para a previsão do tempo, considere que cada coluna representa um dia da semana:

Realidade	☀️	☁️	☀️	☀️	☀️	☀️	☀️
Modelo 1	☀️	☀️	☀️	☀️	☀️	☀️	☀️
Modelo 2	☀️	☀️	☀️	☀️	☀️	☀️	☀️
Modelo 3	☀️	☀️	☀️	☀️	☀️	☀️	☀️
Modelo 4	☀️	☀️	☀️	☀️	☀️	☀️	☀️
Modelo 5	☀️	☀️	☀️	☀️	☀️	☀️	☀️
Combinação	☀️	☀️	☀️	☀️	☀️	☀️	☀️

Slide feito por Carla Gomes

54

Grande sucesso na prática



55

Como combinar?



Pontil Jr, Moacir P. "Combining Classifiers: from the creation of ensembles to the decision fusion."

56

O que esperar?

Suponha que:

- Erros de classificadores não são correlacionados;
- Tenhamos 5 classificadores;
- Classificação final via "voto majoritário";
- Cada classificador-componente tenha acurácia de 70%;

Qual é a acurácia esperada do ensemble?

- $10(.7^3) + 5(.7^4)(.3) + (.7^5) = 83.7\%$.
- Para 101 classificadores temos uma acurácia de 99,9%.

Realidade???

O que fazer na prática?

- Mesmos algoritmos com parâmetros diferentes (e.g., k-NN, redes neurais etc.); **Árvores são melhores que k-NN???**
- Bases de dados com diferentes atributos;
- Subconjuntos da mesma base de dados (*bagging*);
- Reponderar a base de treinamento (*boosting*);

Bagging

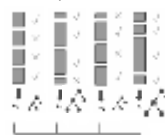
Algoritmo de Bootstrap Aggregation (Brieman, 1996):

- 1) Amostrar, M vezes, N exemplos da base de dados (com reposição);
- 2) Treinar M classificadores (um para cada amostra);
- 3) Combinar os classificadores via voto majoritário.

- Passo 1) insere variância nas bases de treinamento dos componentes, aumentando a estabilidade do *ensemble*.
- Espera-se que em cada amostra existam 63,2% de tuplas não repetidas;
- *Random Forests* são baseadas nessa ideia.

Boosting

- Em vez de reamostrar, repondera exemplos;
- Cada iteração induz um classificador e repondera exemplos;
- Ensemble é baseado no voto ponderado (acurácia) dos componentes.



- Cada retângulo representa um exemplo;
- Tamanho da árvore reflete a acurácia e indica seu peso no ensemble.

Agenda

- Conceitos preliminares e classificador 1R (aquecimento)
- Classificadores Bayesianos
- Árvores de classificação
- k-Nearest Neighbors (k-NN)
- Classifier ensembles
- **Random forests**
- Noções sobre SVMs
- Redes Neurais
- Avaliação de classificadores

Random forests

Indução do classificador:

Considere que temos N exemplos de treinamento.

Para cada uma das t iterações faça:

1. Amostrar N exemplos com reposição (bagging).
2. Induzir uma árvore repetindo recursivamente os seguintes passos em cada novo nó da árvore, até que o critério de parada seja satisfeito:
 - a) Selecionar m atributos (e.g., $m \leq \sqrt{p}$) aleatoriamente.
 - b) Selecionar o melhor atributo/corte (entre os m candidatos).
 - c) Criar dois nós filhos usando o ponto de corte do passo anterior.
3. Armazenar a árvore obtida.

Classificação

Para cada uma das t árvores de classificação:

Predizer o rótulo de classe do exemplo do conjunto-alvo.

Retornar a classe predita com maior frequência.



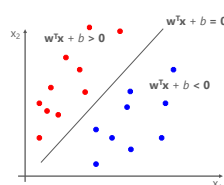
Leo Breiman

Agenda

- Conceitos preliminares e classificador 1R (aquecimento)
- Classificadores Bayesianos
- Árvores de classificação
- k-Nearest Neighbors (k-NN)
- Classifier ensembles
- Random forests
- **Noções sobre SVMs**
- Redes Neurais
- Avaliação de classificadores

Classificação com máxima margem

- Classificação binária pode ser vista como uma tarefa de separar classes num espaço de características:



$$f(\mathbf{x}) = \text{sign}(\mathbf{w}^T \mathbf{x} + b)$$

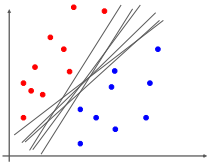
Exemplo:

$$\mathbf{w}^T = [-1 \ 1], \mathbf{x} = [x_1 \ x_2]^T, b=0.$$

$$f(\mathbf{x}) = \text{sign}(x_2 - x_1)$$

Slide de Raymond Mooney

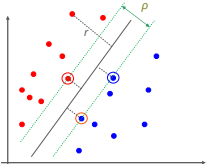
Qual é o separador ótimo?



Slide de Raymond Mooney

Otimizar a margem de classificação

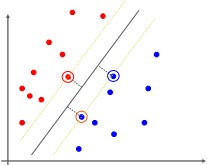
- Distância de \mathbf{x}_i para o separador é r_i .
- Exemplos mais próximos ao hiperplano são os **vetores de suporte**.
- **Margem** p do separador é a distância entre os vetores de suporte.




Slide de Raymond Mooney

Otimizar a margem de classificação

- Maximizar a margem é intuitivo e está de acordo com a teoria do aprendizado PAC (*Probably Approximately Correct*, 1984).
- Implica que apenas os **vetores de suporte** importam; demaiss podem ser ignorados.



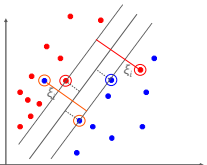


Leslie Valiant
(Turing Award)

72

Classificação com margem flexível

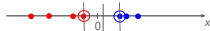
- E se o conjunto de treinamento não é linearmente separável?
- Variáveis *frouxas* (*slack*), ξ_i , podem ser adicionadas para permitir erros de classificação em exemplos ruidosos ou difíceis, resultando numa margem flexível:



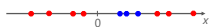
73

SVM não linear

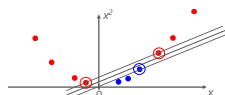
Para dados que são linearmente separáveis podemos usar uma SVM linear (equivalente a um perceptron simples):



E se o problema é muito difícil?



Mapear dados para um espaço de dimensionalidade maior:



Slide de Raymond Mooney

77

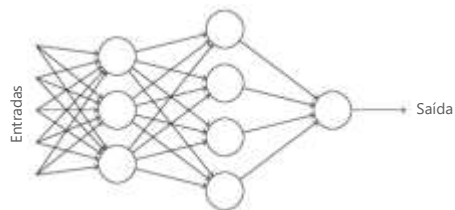
Agenda

- Conceitos preliminares e classificador 1R (aquecimento)
- Classificadores Bayesianos
- Árvores de classificação
- k-Nearest Neighbors (k-NN)
- Classifier ensembles
- Random forests
- Noções sobre SVMs
- **Redes Neurais**
- Avaliação de classificadores

78

Redes Neurais

Ideia geral: combinar os dados de entrada para gerar a(s) saída(s) esperada(s)



<http://neuralnetworksanddeeplearning.com>

80

Redes Neurais – Exemplo (Detecção de dígitos)

Problema: inferir qual dígito está na imagem (28x28 = 784 pixels)



<http://neuralnetworksanddeeplearning.com>

81

Redes Neurais – Como obter bons pesos?

1. Não sabemos bons pesos, então iniciamos todos **aleatoriamente**
2. Fornecemos dados para a rede e conseguimos computar o erro da resposta

$$MSE(\mathbf{W}, \mathbf{b}) = \frac{1}{2n} \sum_{\mathbf{x}} \|f^*(\mathbf{x}) - \text{out}(\mathbf{x})\|^2$$

Podemos aprender com os erros da rede?



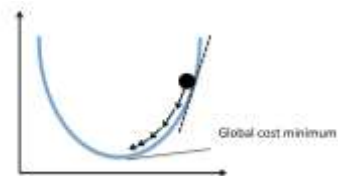
Nazare, T. S. "Unusual Event Detection in Surveillance Videos"

Redes Neurais – Aprendendo com (Minimizando o) Erro

Podemos usar o conceito de gradiente:

- Direção na qual a função mais cresce

Pensando na função de erro: **"andar no sentido contrário ao do gradiente"**



Redes Neurais – Minimizando o Erro (Backpropagation)

Temos o erro da saídas, mas como cada peso é responsável por esse erro???

Como a saída depende dos pesos podemos inferir como cada peso ajuda no erro

Podemos modificar (atualizar) os pesos para melhorar a rede



Agenda

- Conceitos preliminares e classificador 1R (aquecimento)
- Classificadores Bayesianos
- Árvores de classificação
- k-Nearest Neighbors (k-NN)
- Classifier ensembles
- Random forests
- Noções sobre SVMs
- Redes Neurais
- Avaliação de classificadores

Avaliação de classificadores – *matriz de confusão*

		Actual Value (as confirmed by experiment)	
		positives	negatives
Predicted Value (predicted by the test)	positives	TP True Positive	FP False Positive
	negatives	FN False Negative	TN True Negative

Métricas:

- Acurácia:

$$\frac{TP + TN}{TP + TN + FP + FN}$$

- AUC:

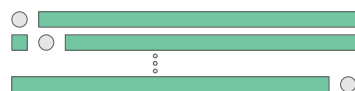
- Dados desbalanceados

Avaliação de classificadores – *cross-validation*

- Considerando uma medida de avaliação (acurácia, precisão, gini etc.), o procedimento padrão é fazer validação cruzada (*k-fold cross-validation*):



- Leave-one out (LOO):

Avaliação de classificadores – *cross-validation*

- **Custo computacional** da validação cruzada com k pastas:

- k treinamentos do classificador em $N(k-1)/k$ exemplos.
- k validações em n/k exemplos.

- **Output:** média das avaliações obtidas nas k validações.

Qual **classificador** uso em produção, para classificar dados não vistos na base de treinamento?

- Classificador induzido com a maior quantidade de dados disponível (e.g., para NB toda a base de treinamento).
- k classificadores (ensemble).



Nota: não há consenso sobre o uso dos termos **teste** e **validação** (ambos podem significar a mesma coisa).

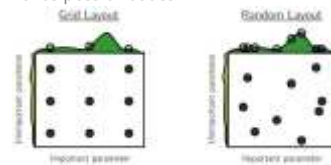
Grid search e Randomized search

Como encontrar os melhores **hiper parâmetros** dos modelos?

- Qual o melhor valor de k para o meu k -NN?
- Quantas árvores usar no meu Random Forest?
- Quantas camadas minha rede neural deve ter?



Podemos testar várias possibilidades



Dicas

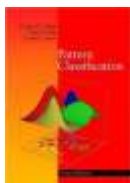
Algoritmos simples frequentemente funcionam muito bem na prática. Além disso

- Menor tempo de construção do modelo;
- Combinação (*ensembles*) de algoritmos simples;
- *Baselines*.

Sugestão

- Usar um único atributo (melhor discriminador – 1R);
- Usar todos os atributos, assumindo independência condicional;
- Árvores de Decisão (interpretabilidade) e *Random Forests*;
- Regressão Logística (LASSO);
- K-NN;
- Ensembles, SVMs e redes neurais.

Sucesso de cada algoritmo depende do domínio de aplicação: ver *No Free Lunch Theorems*.



Diretoria

 Engenharia de Dados

Obrigado!

Módulo 3 – Regressão

Atlas de Ciência de Dados | Aprendizado de Máquina



Agenda

- Regressão linear simples
- Métodos de estimação
- Regressão linear múltipla
- Regularização (Ridge, Lasso e Elastic-Net)
- k-Nearest Neighbors
- Árvore de regressão
- Redes neurais
- SVM regressor
- Softwares
- Quiz

Motivação

Dinheiro traz felicidade?

Country	GDP per capita (US\$)	Life satisfaction
Hungary	12240,0	4,9
Korea	27195,0	5,8
France	37675,0	6,5
Australia	50962,0	7,3
United States	55805,0	7,2

O que acham?
Podemos usar uma função?

99

Motivação

Dinheiro traz felicidade?

Country	GDP per capita (US\$)	Life satisfaction
Hungary	12240,0	4,9
Korea	27195,0	5,8
France	37675,0	6,5
Australia	50962,0	7,3
United States	55805,0	7,2

Diversas possibilidades
Qual é a melhor e porque?

99

Motivação

Dinheiro traz felicidade?

Country	GDP per capita (US\$)	Life satisfaction
Hungary	12240,0	4,9
Korea	27195,0	5,8
France	37675,0	6,5
Australia	50962,0	7,3
United States	55805,0	7,2

Melhor ajuste

100

Modelo de Regressão Linear

- **Objetivo:** “prever” o valor de y (qualidade de vida) usando dados observados de x (renda per capita).
- **Abordagem:** podemos usar uma reta (ou hiperplano, para mais dimensões). Assim:
$$y = f(x)$$

Qual é a cara dessa função?

Modelo de Regressão Linear

Reta

- a : deslocamento
- b : inclinação

$y = a + bx$

Modelo de Regressão Linear

Reta

$y = a + bx$

a fixo; b variável

a variável; b fixo

Como encontrar a melhor reta?

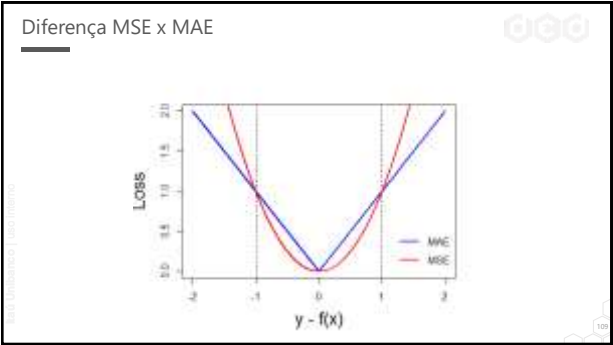
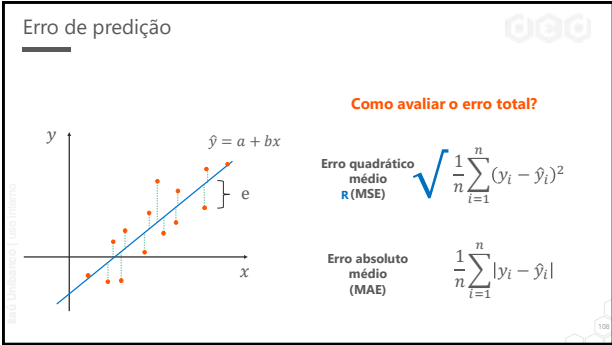
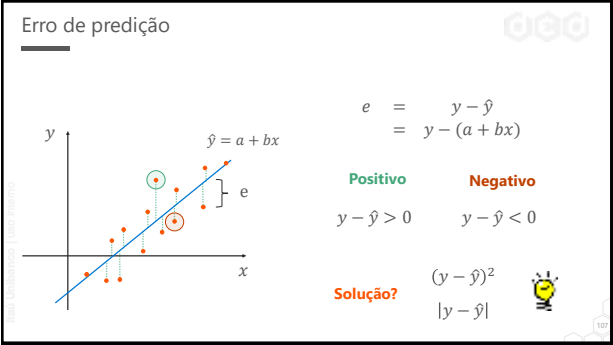
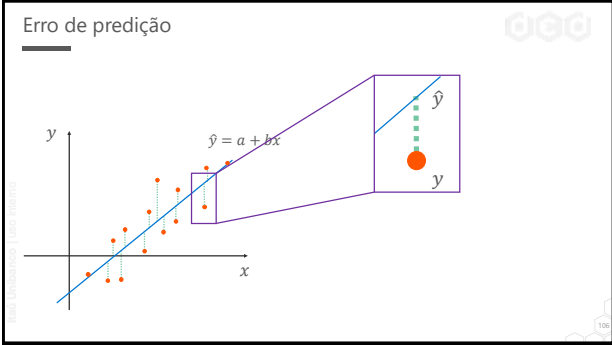
Dinheiro traz felicidade?

Country	GDP per capita (US\$)	Life satisfaction
Hungary	12240,0	4,9
Korea	27195,0	5,8
France	37675,0	6,5
Australia	50962,0	7,3
United	55805,0	7,2

Como sabemos o quão boa é essa reta?

Agenda

- Regressão linear simples
- Métodos de estimação
- Regressão linear múltipla
- Regularização (Ridge, Lasso e Elastic-Net)
- k-Nearest Neighbors
- Árvore de regressão
- Redes neurais
- SVM regressor
- Softwares
- Quiz



Método simplista

- Busca por tentativa e erro.
- Para um número n de possibilidades, fazer:
 1. Simular (chutar) valores de a e b
 2. Calcular $\hat{y} = a + bx$
 3. Avaliar o erro de acordo com uma métrica pré determinada (MSE, MAE, etc.)
 4. Selecionar a e b que apresentam o menor erro

Qual a eficiência do método? É rápido?

Método teórico

- **Objetivo:** encontrar os valores de a e b que minimizam o erro (MSE ou MAE, etc).
- **Abordagem:**
 - Mínimos quadrados
 - Máxima verossimilhança
 - Gradiente descendente

Mínimos quadrados

- Achar a e b tal que a soma dos erros quadráticos (RSS) é mínima!

$$RSS(a, b) = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$
$$= \sum_{i=1}^n (y_i - (a + bx_i))^2$$

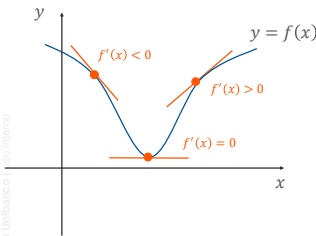


\hat{a} e \hat{b} ?

E agora, como minimizar?



Como minimizar?



Derivadas parciais

$$f'(x) = \frac{df(x)}{dx}$$

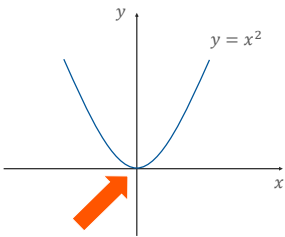
$$f''(x) = \frac{d^2 f(x)}{d^2 x}$$

Ponto de mínimo

$$\begin{cases} f'(x) = 0 \\ f''(x) > 0 \end{cases}$$

Como minimizar?

Exemplo:



Derivadas parciais

$$f'(x) = \frac{df(x)}{dx} \quad f'(x) = 2x$$
$$f''(x) = \frac{d^2f(x)}{d^2x} \quad f''(x) = 2$$

Ponto de mínimo

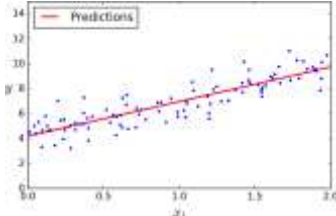
$$\begin{cases} f'(x) = 0 & 2x = 0 \rightarrow x = 0 \\ f''(x) > 0 & 2 > 0 \end{cases}$$

Mínimos quadrados

- Achar a e b tal que a soma dos erros quadráticos (RSS) é mínima!

$$RSS(a, b) = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$
$$\hat{a} = \bar{y} - \hat{b}\bar{x}$$
$$\hat{b} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

Mínimos quadrados



$\hat{a} = 4.21$
 $\hat{b} = 2.77$

Mínimos quadrados

- Representação em forma matricial:

$$RSS(a, b) = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$
$$= \sum_{i=1}^n (y_i - (a + bx_i))^2$$
$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \quad X = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix} \quad \beta = \begin{pmatrix} a \\ b \end{pmatrix}$$

Minimos quadrados

Qual a desvantagem?

• Representação em forma matricial:

$$RSS(a,b) = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$
$$= \sum_{i=1}^n (y_i - (a + bx_i))^2$$

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

$$X\beta = \begin{pmatrix} a & bx_1 \\ a & bx_2 \\ \vdots & \vdots \\ a & bx_n \end{pmatrix}$$

$$RSS(\beta) = (y - X\beta)^T (y - X\beta)$$

$$\hat{\beta} = (X'X)^{-1}X'y$$

Exercício

• Calcular a e b para os seguintes dados usando o método de mínimos quadrados:

id	y	x
1	22	6
2	20	5
3	20	5
4	24	7
5	28	9
6	24	7
7	22	6
8	20	5
9	20	5
10	34	12

$$\hat{b} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$
$$\hat{a} = \bar{y} - \hat{b}\bar{x}$$
$$\hat{a} \text{ e } \hat{b}?$$

Exercício

id	y	x	$x_i - \bar{x}$	$y_i - \bar{y}$	$(x_i - \bar{x})(y_i - \bar{y})$	$(x_i - \bar{x})^2$
1	22	6	-0,7	-1,4	0,98	0,49
2	20	5	-1,7	-3,4	5,78	2,89
3	20	5	-1,7	-3,4	5,78	2,89
4	24	7	0,3	0,6	0,18	0,09
5	28	9	2,3	4,6	10,58	5,29
6	24	7	0,3	0,6	0,18	0,09
7	22	6	-0,7	-1,4	0,98	0,49
8	20	5	-1,7	-3,4	5,78	2,89
9	20	5	-1,7	-3,4	5,78	2,89
10	34	12	5,3	10,6	56,18	28,09

$$\hat{b} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$
$$\hat{b} = \frac{92,2}{46,1} = 2$$
$$\hat{a} = \bar{y} - \hat{b}\bar{x}$$
$$\hat{a} = 23,4 - 2 \cdot 6,7 = 10$$

$$\bar{y} = 23,4$$
$$\bar{x} = 6,7$$
$$\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) = 92,2$$
$$\sum_{i=1}^n (x_i - \bar{x})^2 = 46,1$$

Máxima verossimilhança

• Estimar os parâmetros fazendo suposições sobre a distribuição de probabilidades dos erros.

• Recapitulando:




$$y = a + bx + \varepsilon$$
$$\varepsilon \sim N(0, \sigma^2)$$
$$y \sim N(a + bx, \sigma^2)$$

Máxima verossimilhança

$$y \sim N(a + bx, \sigma^2)$$
$$p_Y(y_i | x_i) = (2\pi\sigma^2)^{-\frac{1}{2}} \exp\left(-\frac{1}{2} \frac{(y_i - (a + bx_i))^2}{\sigma^2}\right)$$

• **Objetivo:** maximizar a log-verossimilhança! ?

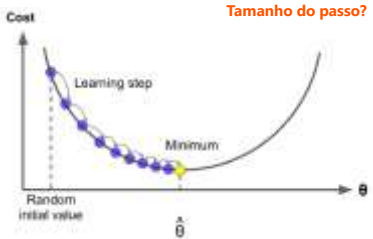
$$\log\left(\prod_{i=1}^n p_Y(y_i | x_i)\right) \quad \blacksquare \quad \hat{\beta} = (X'X)^{-1}X'y$$



Gradiente descendente

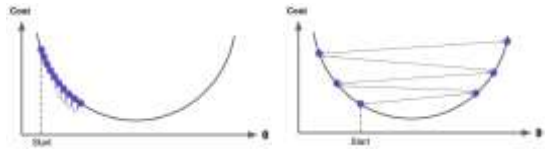
- Ajustar os parâmetros do modelo iterativamente (passos) com a finalidade de minimizar uma função de custo.
- **Ideia:** caminhar no sentido contrário do gradiente (derivada) da função de custo (MSE, MAE, etc) para encontrar o conjunto de parâmetros tal que o gradiente é nulo (ponto de mínimo).

Gradiente descendente

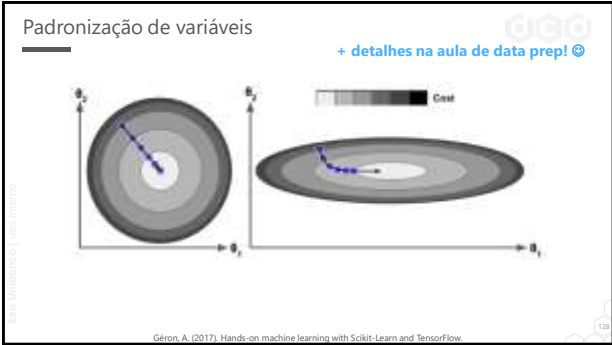
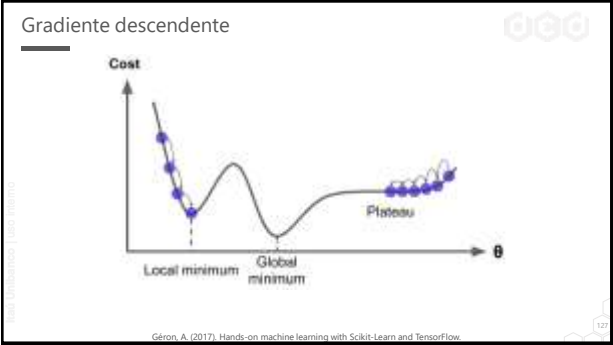
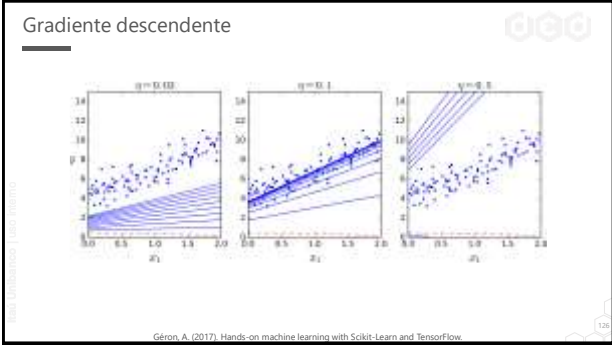


Géron, A. (2017). Hands-on machine learning with Scikit-Learn and TensorFlow

Gradiente descendente



Géron, A. (2017). Hands-on machine learning with Scikit-Learn and TensorFlow



Gradiente descendente

Qual a desvantagem?

- Aprendizado sequencial:

1. Iniciar o algoritmo com pesos aleatórios $\beta = (a, b)$.
2. Para um número n de iterações, fazer:
 - a) Calcular o gradiente (derivada) da função de custo. Para o MSE:
$$\nabla_{\beta} MSE(\beta) = \frac{2}{n} X^T (X\beta - y)$$
 - b) Atualizar os pesos usando o gradiente:
$$\beta^* = \beta - \eta \nabla_{\beta} MSE(\beta)$$

Parar quando atingir o número de iterações ou quando não houver muita alteração nos valores.

Exercício

• Calcular a e b para os seguintes dados usando o método do gradiente descendente:

$$a_n = a_{n-1} - \eta * \frac{dRSS}{da}$$

$$b_n = b_{n-1} - \eta * \frac{dRSS}{db}$$

$$\eta = 0,001$$

id	y	x
1	22	6
2	20	5
3	20	5
4	24	7
5	28	9
6	24	7
7	22	6
8	20	5
9	20	5
10	34	12

$$\frac{1}{2}RSS = \frac{1}{2} \sum_{i=1}^n (y_i - (a + bx_i))^2$$
$$\frac{dRSS}{da} = \sum_{i=1}^n -(y_i - (a + bx_i))$$
$$\frac{dRSS}{db} = \sum_{i=1}^n -x_i(y_i - (a + bx_i))$$

\hat{y}_i

Exercício

id	y	x	a + bx	1/2(y _i - \hat{y}_i) ²	-(y _i - \hat{y}_i)	-x _i (y _i - \hat{y}_i)
1	22	6	15	25	-7	-42
2	20	5	14	18	-6	-30
3	20	5	14	18	-6	-30
4	24	7	16	32	-8	-56
5	28	9	18	50	-10	-90
6	24	7	16	32	-8	-56
7	22	6	15	25	-7	-42
8	20	5	14	18	-6	-30
9	20	5	14	18	-6	-30
10	34	12	21	85	-13	-156
Σ				319,5	-77,0	-562,0

$$a_1 = a_0 - 0,001 * (-77,0) = 9,08$$

$$b_1 = b_0 - 0,001 * (-562,0) = 1,56$$

Exercício

$$a_1 = 9,08$$

$$b_1 = 1,56$$

id	y	x	a + bx	1/2(y _i - \hat{y}_i) ²	-(y _i - \hat{y}_i)	-x _i (y _i - \hat{y}_i)
1	22	6	18,45	6,30	-3,55	-21,31
2	20	5	16,89	4,85	-3,11	-15,57
3	20	5	16,89	4,85	-3,11	-15,57
4	24	7	20,01	7,96	-3,99	-27,92
5	28	9	23,14	11,83	-4,87	-43,79
6	24	7	20,01	7,96	-3,99	-27,92
7	22	6	18,45	6,30	-3,55	-21,31
8	20	5	16,89	4,85	-3,11	-15,57
9	20	5	16,89	4,85	-3,11	-15,57
10	34	12	27,82	19,09	-6,18	-74,15
Σ				78,83	-38,58	-278,65

$$a_2 = a_1 - 0,001 * (-38,58) = 9,12$$

$$b_2 = b_1 - 0,001 * (-278,65) = 1,84$$

Exercício

$$a_2 = 9,12$$

$$b_2 = 1,84$$

id	y	x	a + bx	1/2(y _i - \hat{y}_i) ²	-(y _i - \hat{y}_i)	-x _i (y _i - \hat{y}_i)
1	22	6	20,16	1,69	-1,84	-11,04
2	20	5	18,32	1,41	-1,68	-8,41
3	20	5	18,32	1,41	-1,68	-8,41
4	24	7	22,00	2,00	-2,00	-14,00
5	28	9	25,68	2,69	-2,32	-20,87
6	24	7	22,00	2,00	-2,00	-14,00
7	22	6	20,16	1,69	-1,84	-11,04
8	20	5	18,32	1,41	-1,68	-8,41
9	20	5	18,32	1,41	-1,68	-8,41
10	34	12	31,20	3,91	-2,80	-33,56
Σ				19,64	-19,52	-138,13

$$a_3 = a_2 - 0,001 * (-19,52) = 9,14$$

$$b_3 = b_2 - 0,001 * (-138,13) = 1,98$$

Agenda

- Regressão linear simples
- Métodos de estimação
- **Regressão linear múltipla**
- Regularização (Ridge, Lasso e Elastic-Net)
- k-Nearest Neighbors
- Árvore de regressão
- Redes neurais
- SVM regressor
- Softwares
- Quiz



134

Regressão linear múltipla



• Relembrando:

$$y = a + bx + \varepsilon$$

- **Novidade:** suponha agora que temos outras variáveis (x_1, \dots, x_p) para prever y . Podemos reescrever o modelo:

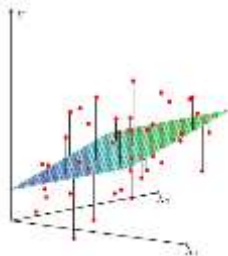
$$y = a + b_1x_1 + b_2x_2 + \dots + b_px_p + \varepsilon$$

- x_p pode ser numérica, categórica ou uma transformação de uma variável ($x_2 = x_1^2$).

Regra: Não compartilhar | Não modificar

135

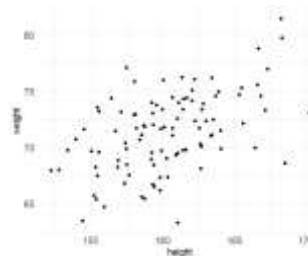
Regressão linear múltipla



Friedman, J., Hastie, T., & Tibshirani, R. (2001). The elements of statistical learning.

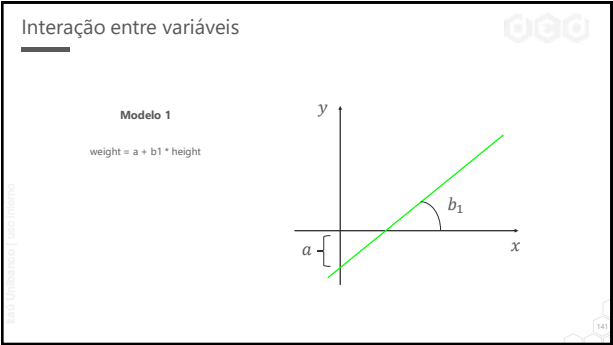
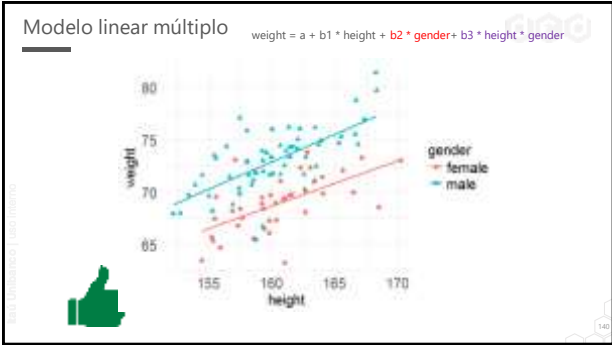
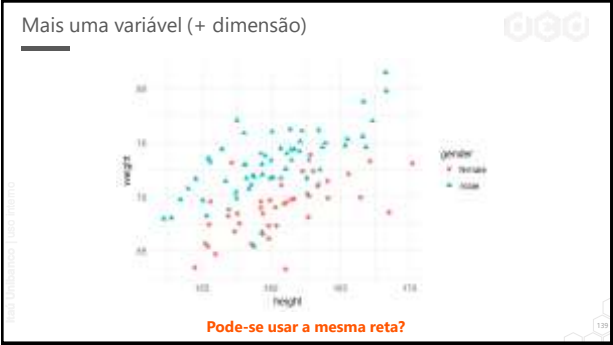
136

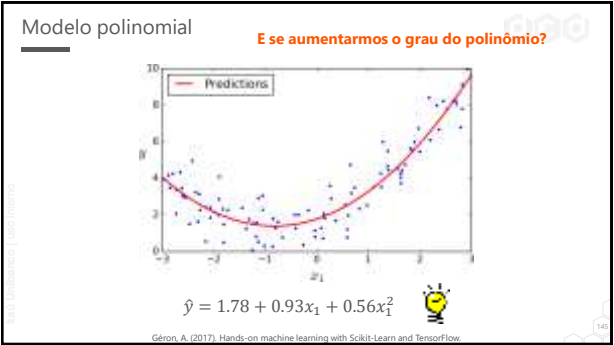
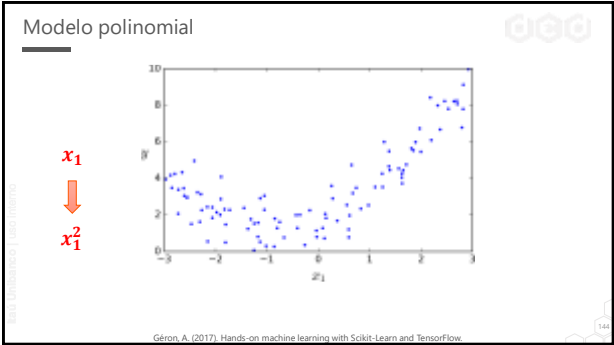
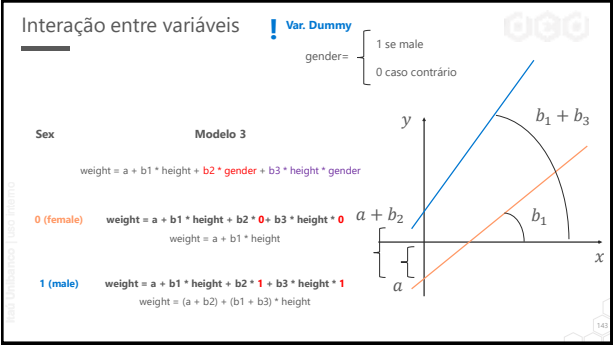
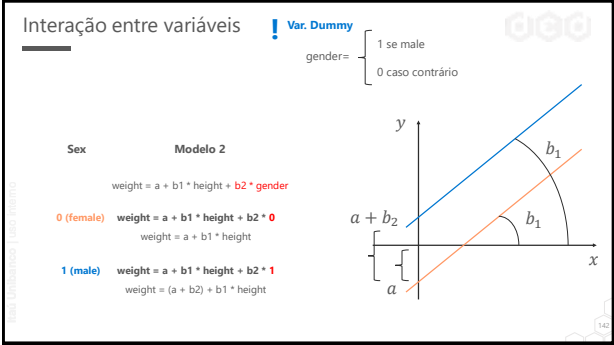
Exemplo



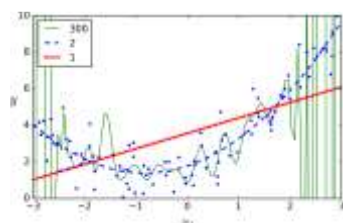
Regra: Não compartilhar | Não modificar

137





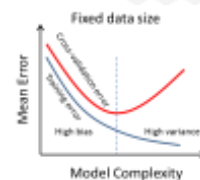
Modelo polinomial



Géron, A. (2017). Hands-on machine learning with Scikit-Learn and TensorFlow.

Cuidado com o superajuste (overfitting)

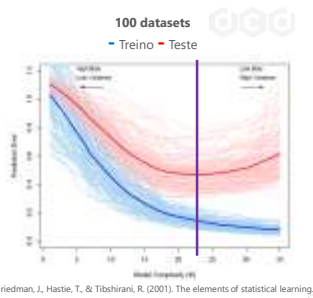
- Podemos definir o grau do polinômio utilizando validação cruzada.



Trade-off: viés x variância

Podemos decompor o erro de generalização do modelo em 3 tipos de erros:

- Viés:** suposições erradas
- Variância:** sensibilidade excessiva do modelo devido à pequena variação dos dados
- Erro irredutível:** ruído dos próprios dados

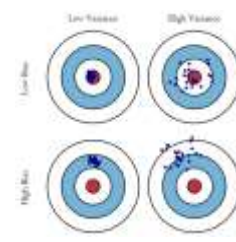


Friedman, J., Hastie, T., & Tibshirani, R. (2001). The elements of statistical learning.

Trade-off: viés x variância

Podemos decompor o erro de generalização do modelo em 3 tipos de erros:

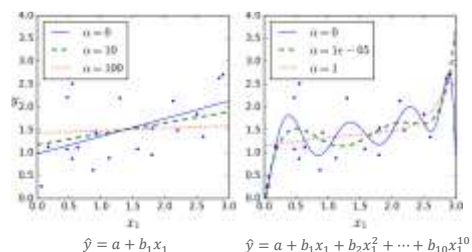
- Viés:** suposições erradas
- Variância:** sensibilidade excessiva do modelo devido à pequena variação dos dados
- Erro irredutível:** ruído dos próprios dados



<http://scott.fortmann-roe.com/docs/BiasVariance.html>

Ridge regression

Vantagens e desvantagens?



Géron, A. (2017). Hands-on machine learning with Scikit-Learn and TensorFlow.

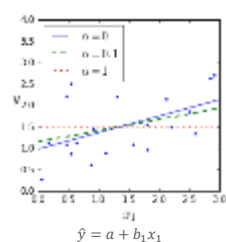
Lasso regression

- Método alternativo para fazer regularização!
- Ao invés de usar a norma l_2 , usamos a norma l_1 ($\|x\|_1 = \sum_{i=1}^n |x_i|$).

- **Abordagem:** adicionar o termo $\alpha \sum_{i=1}^n |\beta_i|$ à função de custo:

$$J(\beta) = \text{MSE}(\beta) + \alpha \sum_{i=1}^n |\beta_i|$$

Lasso regression

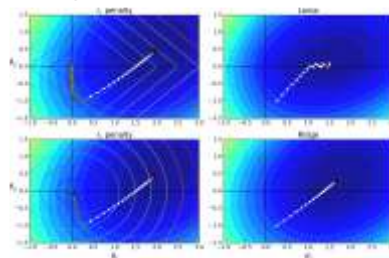


- $\hat{y} = 0,97 + 0,38x_1$
- - $\hat{y} = 1,14 + 0,26x_1$
- $\hat{y} = 1,50 + 0,00x_1$

Géron, A. (2017). Hands-on machine learning with Scikit-Learn and TensorFlow.

Ridge vs Lasso regression

Qual a diferença?



Géron, A. (2017). Hands-on machine learning with Scikit-Learn and TensorFlow.

Rigde + Lasso = Elastic-Net

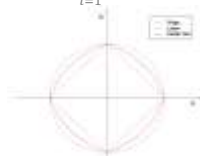
Como determinar os melhores r e α ?

- **Abordagem:** combinar os dois termos de regularização para obter uma nova função de custo:

$$J(\beta) = MSE(\beta) + r\alpha \sum_{i=1}^n |\beta_i| + \frac{1-r}{2}\alpha \sum_{i=1}^n \beta_i^2$$

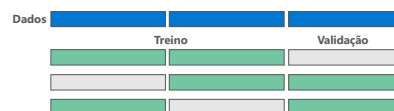
$r = 0$ → Ridge

$r = 1$ → Lasso

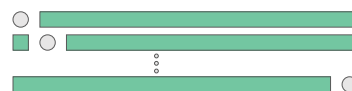


Validação cruzada (... lembrando)

k-fold cross-validation



Leave-one out (LOO)



Onde usar?

- Propensão do uso do LIS
- Régua de cobrança
- Previsão de investimentos de um cliente fora do banco
- Gasto com seguros e PIC
- etc

Agenda

- Regressão linear simples
- Métodos de estimação
- Regressão linear múltipla
- Regularização (Ridge, Lasso e Elastic-Net)
- **k-Nearest Neighbors**
- Árvore de regressão
- Redes neurais
- SVM regressor
- Softwares
- Quiz

k-Nearest Neighbors

Dado x , $t = f(x)$?

Problema: $f(\cdot)$ é desconhecida.

Solução k-NN: Estimar t por meio da média dos vizinhos mais próximos (em relação aos valores de x apenas).

Medidas de distância

Manhattan

$$d_1 = |x_1 - x_2| + |y_1 - y_2|$$

Euclidiana

$$d_2 = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Medidas de distância

Onde se situam os pontos equidistantes de um vetor?

Euclidiana

Mahalanobis

Suprema

Manhattan

Exemplo

Indivíduo	Idade	Empréstimo	Preço	Distância
1	25	40	135	104,56
2	35	60	256	83,02
3	45	80	231	62,07
4	20	20	267	125,17
5	35	120	139	25,55
6	52	18	150	124,06
7	23	95	127	53,24
8	40	62	216	80,40
9	60	100	139	43,68
10	48	220	250	78,00
11	33	150	264	17,00

k = ?

k = 1 264

k = 2 $(264 + 139)/2 = 201,5$

k = 3 $(264 + 139 + 43,68)/3 = 148,89$

12	48	142	?
----	----	-----	---

k-Nearest Neighbors

Algoritmo:

Dado um exemplo x_q cujo valor da variável dependente (y) se deseja estimar, e considerando que x_1, x_2, \dots, x_k representam os k exemplos mais próximos de x_q , retornar:

$$y = f(x_q) = \frac{\sum_{i=1}^k f(x_i)}{k}$$

- Predição por meio da média da vizinhança

- **K-NN ponderado pelo inverso da distância:**

$$y = f(x_q) = \frac{\sum_{i=1}^k w_i f(x_i)}{\sum_{i=1}^k w_i}$$

$$w_i = \frac{1}{d(x_q, x_i)^2}$$

Parâmetros?

- Normalização / padronização ?
- Medida de distância ?
- Número de vizinhos, $k = ?$
- Pesos dos atributos ?
- Preprocessamento via *clustering* minimiza custo de inferência, bem como custo de otimização de parâmetros.

Validação Cruzada

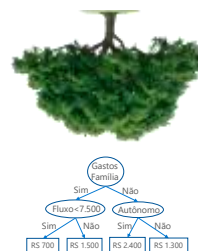
- k-NN é muito usado na prática por conta da simplicidade de implementação computacional, mas custo de inferência pode ser fator impeditivo.

Agenda

- Regressão linear simples
- Métodos de estimação
- Regressão linear múltipla
- Regularização (Ridge, Lasso e Elastic-Net)
- k-Nearest Neighbors
- **Árvore de regressão**
- Redes neurais
- SVM regressor
- Softwares
- Quiz

Árvores de regressão

- Procedimento idêntico àquele para construir árvores de classificação (mais detalhes no módulo de classificação), exceto pela função objetivo de otimização:
 - Trocar entropia/gini pelo erro quadrático;
- Particionamento recursivo dos dados;
- Considerar todos os valores de todas as variáveis na construção do modelo;
- Selecionar par de variável-valor que produz o melhor ajuste à variável dependente;
- Se $X < V_1$ então enviar a tupla para o ramo esquerdo, caso contrário pro direito;
- Repetir o processo em cada um dos nós, gerando uma árvore.



Complexidade

Profundidade da árvore



Como otimizar a profundidade?

Problema?

Vantagens

- Modelo não paramétrico (não presume f.d.p!);
- Realiza seleção de atributos automaticamente;
- Lida com atributos binários, categóricos, ordinais e contínuos;
- Descobrir regras que mostram interações entre variáveis;
- Pouco sensível a *outliers* na variável dependente;
- Útil para análise exploratória de dados;
- Serve para categorizar variáveis – e.g., idade (jovem, adulto, velho) em função da correlação com a variável dependente (e.g., renda).

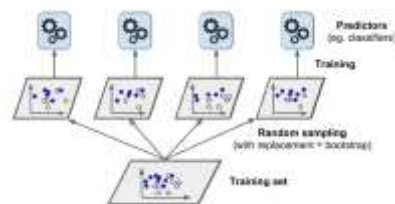
Desvantagens

- Modelo obtido é uma "step function";
 - Melhorado com regressores lineares nos nós folha.
- Problemas complexos requerem árvores complexas;
 - Perde-se a interpretabilidade.
- Modelos instáveis
 - Flutuações amostrais podem resultar em árvores bem diferentes (lembre de bias x variância);

- Problemas difíceis? Plante uma floresta!
- *Random Forests* ou *Boosting*.



Bagging (... relembando)



Geron, A. (2017). Hands-on machine learning with Scikit-Learn and TensorFlow

Algoritmos de Árvore

Random Forest

Dados

Amostra com reposição (bootstrap)

Árvore 1

Árvore 2

Árvore N

Previsão 1

Previsão 2

Previsão N

Em cada nó, o melhor "split" é selecionado em um conjunto aleatório de variáveis

$$\text{Previsão Final} = \frac{\sum_{i=1}^N \text{Previsão}_i}{n}$$

Algoritmos de Árvore

Extremely Randomized Trees

! pode-se utilizar bootstrap

Árvore 1

Árvore 2

Árvore N

Previsão 1

Previsão 2

Previsão N

Em cada nó, seleciona-se o melhor "split" aleatório em um conjunto aleatório de variáveis

$$\text{Previsão Final} = \frac{\sum_{i=1}^N \text{Previsão}_i}{n}$$

Boosting

Géron, A. (2017). Hands-on machine learning with Scikit-Learn and TensorFlow.

Algoritmos de Árvore

Gradient Boosting

Cada árvore depende da anterior, pois ela dá mais peso nas observações mal ajustadas pela sua antecessora

Árvore 1

Árvore 2

Árvore N

Previsão 1

Previsão 2

Previsão N

Erro

Erro

$$\text{Previsão Final} = w_1 \text{Previsão}_1 + \dots + w_N \text{Previsão}_N$$

Agenda

- Regressão linear simples
- Métodos de estimação
- Regressão linear múltipla
- Regularização (Ridge, Lasso e Elastic-Net)
- k-Nearest Neighbors
- Árvore de regressão
- **Redes neurais**
- SVM regressor
- Softwares
- Quiz

182

Multilayer perceptrons (MLP)

$a_i = g(w_{io}^{(3)} + \sum w_{ij}^{(3)} x_j)$

1) Identidade (linear)
5) MSE

Toda função contínua pode ser aproximada, com um erro arbitrariamente pequeno, por uma MLP

Pontos de Decisão

- 1 Função de ativação
- 2 Número de neurônios em cada camada
- 3 Número de camadas
- 4 Método de otimização
- 5 Loss

183

Modelo linear? Caso particular!

Modelo linear é a menor deep learning do mundo!

Modelo linear

$$y = a + b_1x_1 + b_2x_2 + b_3x_3$$

MLP c/ 1 camada e 1 neurônio

Inputs: 1, x_1 , x_2 , x_3

Pesos: a , b_1 , b_2 , b_3

Ativação: $f(x) = x$

Output: $a + b_1x_1 + b_2x_2 + b_3x_3$

184

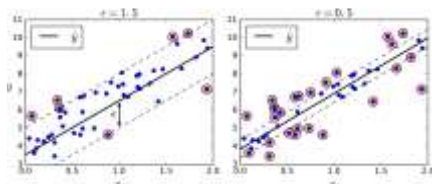
Agenda

- Regressão linear simples
- Métodos de estimação
- Regressão linear múltipla
- Regularização (Ridge, Lasso e Elastic-Net)
- k-Nearest Neighbors
- Árvore de regressão
- Redes neurais
- **SVM regressor**
- Softwares
- Quiz

185

SVM Regressor

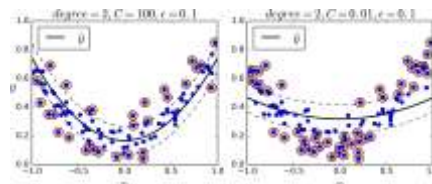
- **Objetivo:** ao invés de encontrar a maior "rua" que separa as classes (classificação), considerar "rua" (margem) que engloba a maior quantidade de exemplos possível!



Géron, A. (2017). Hands-on machine learning with Scikit-Learn and TensorFlow.

SVM Regressor

- **Objetivo:** ao invés de encontrar a maior "rua" que separa as classes (classificação), considerar "rua" (margem) que engloba a maior quantidade de exemplos possível!



Géron, A. (2017). Hands-on machine learning with Scikit-Learn and TensorFlow.

Agenda

- Regressão linear simples
- Métodos de estimação
- Regressão linear múltipla
- Regularização (Ridge, Lasso e Elastic-Net)
- k-Nearest Neighbors
- Árvore de regressão
- Redes neurais
- SVM regressor
- **Softwares**
- Quiz

Softwares



Agenda

- Regressão linear simples
- Métodos de estimação
- Regressão linear múltipla
- Regularização (Ridge, Lasso e Elastic-Net)
- k-Nearest Neighbors
- Árvore de regressão
- Redes neurais
- SVM regressor
- Softwares
- **Quiz**

Quiz

- Questão 1) Quais das seguintes métricas de avaliação podem ser usadas para modelar uma variável resposta contínua?

- a) AUC-ROC
- b) Acurácia
- c) Logloss
- d) Erro quadrático médio ✓

Quiz

- Questão 2) Regularização via LASSO pode ser usada para selecionar variáveis no modelo de regressão linear?

- a) Verdadeiro ✓
- b) Falso

Quiz

- Questão 3) O que acontece quando aplicamos uma penalização alta no modelo Ridge regression?

- a) Alguns coeficientes se tornarão nulos (zero absoluto)
- b) Alguns coeficientes tenderão a zero, mas não serão nulos ✓
- c) Letras a e b são verdadeiras
- d) Nenhuma das anteriores é verdadeira

Quiz

• Questão 4) Quais das seguintes sentenças sobre pontos atípicos (outliers) em regressão linear são verdadeiras?

- a) Regressão linear é sensível à outliers ✓
- b) Regressão linear não é sensível à outlier
- c) Não sei
- d) Nenhuma das anteriores é verdadeira

+ E se der ruim?

• **Hipótese:** temos um modelo com capacidade de predição baixa.

• O que fazer?



Avaliar a capacidade de ordenação!

Correlação de Spearman



Obrigado!

Módulo 4 – Agrupamento de Dados

Atlas de Ciência de Dados | Aprendizado de Máquina



Créditos

O material a seguir consiste de adaptações e extensões dos originais:

- Material do curso: SCC5895 – Análise de Agrupamento de Dados – USP/São Carlos
Prof. Dr. Eduardo Raul Hruschka

CC BY-SA 4.0 licensed presentation

199

Agenda

- Motivação e conceitos
- Definições preliminares
- Algoritmos Hierárquicos
- Algoritmos Particionais
- Algoritmos Baseados em Densidade
- Avaliação de agrupamentos
- Extra: Bissect k-Means, EM, ...

CC BY-SA 4.0 licensed presentation

199

Agenda

- Motivação e conceitos
- Definições preliminares
- Algoritmos Hierárquicos
- Algoritmos Particionais
- Algoritmos Baseados em Densidade
- Avaliação de agrupamentos
- Extra: Bissect k-Means, EM, ...

CC BY-SA 4.0 licensed presentation

200

Motivação e potenciais aplicações

Humanos se interessam por categorizações:

- Música: erudita, popular, religiosa etc.



- Filmes: animação, comédia, drama etc.

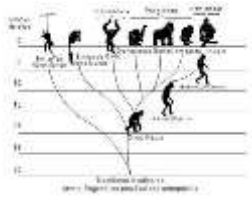


Motivação e potenciais aplicações

Diversas ciências se baseiam na *organização* de objetos de acordo com suas similaridades.

Biologia:

- Reino: Animalia
- Ramo: Chordata
- Classe: Mammalia
- Ordem: Primatas
- Família: *Hominidae*
- Gênero: *Homo* (homem moderno)
- Espécie: *Homo sapiens*



O que é um agrupamento natural entre os seguintes objetos?

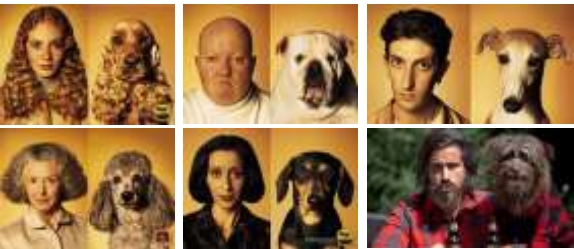


Grupo/cluster é um conceito subjetivo



Subjetividade: Semelhança entre objetos

Quais atributos devemos considerar (e como) para avaliar similaridade?



Literatura Estabelecida

Apesar de todas as dificuldades, a literatura sobre Análise de Agrupamento de Dados é rica e muito bem estabelecida; Trabalhos importantes datam da década de 50.

- P. ex., vide A. K. Jain, **Data Clustering: 50 Years Beyond K-Means**, Pattern Recognition Letters, 2010

"according to JSTOR [jst, 2009], data clustering first appeared in the title of a 1954 article dealing with anthropological data"

Há medidas de dis(similaridade) bem estudadas e fundamentadas para diversos tipos de dados e domínios de aplicação:

Dados Numéricos, Categóricos/Nominais, Binários, ...



Frequência com que se usa clustering?

Web of Science: **+ 12.000 artigos** usando o termo *cluster analysis* no (título, palavras chaves, resumo) oriundos de mais de **3.000 journals** diferentes.

(Xu & Wunsch, *Clustering*, IEEE Press, 2009)



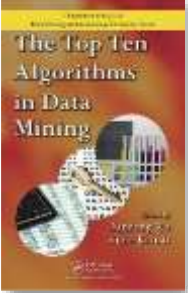


Algorithm	Frequency
Decision Tree/Random Forest	10.0%
K-Means	8.5%
Clustering	7.5%
Statistical Regression	6.5%
Naive Bayes	5.5%
Support Vector Machines	4.5%
Neural Networks	3.5%
Bayesian Networks	2.5%
Association Rules	1.5%
Genetic Algorithms	1.5%
Evolutionary Algorithms	1.5%
Particle Swarm Optimization	1.5%
Ant Colony Optimization	1.5%
Tabu Search	1.5%
Simulated Annealing	1.5%
Genetic Programming	1.5%
Neural Networks	1.5%
Bayesian Networks	1.5%
Association Rules	1.5%
Genetic Algorithms	1.5%
Evolutionary Algorithms	1.5%
Particle Swarm Optimization	1.5%
Ant Colony Optimization	1.5%
Tabu Search	1.5%
Simulated Annealing	1.5%
Genetic Programming	1.5%

IEEE ICDM and ACM SIGKDD Poll

2 Algoritmos (K-Means e EM) listados entre os **Top 10 Most Influential Algorithms in DM**

- Wu, X. and Kumar, V. (Editors), **The Top Ten Algorithms in Data Mining**, CRC Press, 2009
- X. Wu et al., "Top 10 Algorithms in Data Mining", Knowledge and Info. Systems, vol. 14, pp. 1-37, 2008



Agenda

- Motivação e conceitos
- Definições preliminares
- Algoritmos Hierárquicos
- Algoritmos Particionais
- Algoritmos Baseados em Densidade
- Avaliação de agrupamentos
- Extra: Bisect k-Means, EM, ...

Definição

Considerando um conjunto de N objetos a serem agrupados $X = \{x_1, x_2, \dots, x_N\}$, uma **partição** (rígida) é uma coleção de k grupos não sobrepostos $P = \{C_1, C_2, \dots, C_k\}$ tal que:

$$C_1 \cup C_2 \cup \dots \cup C_k = X$$
$$C_i \cap C_j = \emptyset \text{ para } i \neq j$$

Exemplo: $P = \{ (x_1), (x_2, x_3, x_4, x_5), (x_6, x_7) \}$

Definição

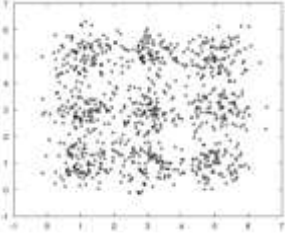
Uma **Matriz de Partição** é uma matriz com k linhas (no. de grupos) e N colunas (no. de objetos) na qual cada elemento μ_{ij} indica o *grau de pertinência* do j -ésimo objeto (x_j) ao i -ésimo grupo (C_i):

$$\mathbf{U}(\mathbf{X}) = \begin{bmatrix} \mu_{11} & \mu_{12} & \cdots & \mu_{1N} \\ \mu_{21} & \mu_{22} & \cdots & \mu_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \mu_{k1} & \mu_{k2} & \cdots & \mu_{kN} \end{bmatrix} \quad \mathbf{U}(\mathbf{X}) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

Se essa matriz for **binária**, ou seja, $\mu_{ij} \in \{0,1\}$ e, ainda, se a restrição $\sum_i (\mu_{ij}) = 1 \forall j$ for respeitada, então denomina-se de matriz de partição rígida ou sem sobreposição.

Clusters? Clusters!

Humanos reconhecem clusters no plano quando os veem, sem saber explicar exatamente porquê (Jain & Dubes, 1988)



9-Gauss Dataset: http://www.icmc.usp.br/~campello/Sub_Pages/JH.htm

Particionamento combinatório

Problema: Presumindo que k seja conhecido, o no. de possíveis formas de agrupar N objetos em k clusters é dado por (Liu, 1968):

$$NM(N, k) = \frac{1}{k!} \sum_{i=0}^k (-1)^i \binom{k}{i} (k-i)^N$$

- Por exemplo, $NM(100, 5) \approx 56.6 \times 10^{67}$. Em um computador com capacidade de avaliar 10^9 partições/s, levaria $\approx 1.8 \times 10^{50}$ séculos para processar todas as avaliações.
- Como k é, em geral, desconhecido, problema é ainda maior.
- Em problemas NP-Hard, precisamos de formulações alternativas.

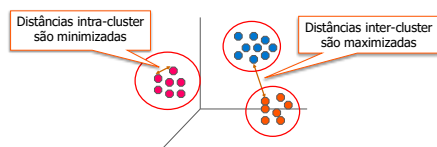
Agrupe os MM's ?!



Definição

"Finding groups of objects such that the objects in a group are similar (or related) to one another and different from (or unrelated to) the objects in other groups." (Tan et al., 2006)

- Uma visão matemática/geométrica:



Agrupamento de E-mails (mineração de texto)



Conceitos Básicos

Algumas Definições (Everitt, 1974):

- Um cluster (grupo) é um conjunto de entidades semelhantes e entidades pertencentes a diferentes clusters não são semelhantes;
- Um grupo é uma aglomeração de pontos no espaço tal que a distância entre quaisquer dois pontos no grupo é menor do que a distância entre qualquer ponto no grupo a qualquer ponto fora deste;
- Grupos podem ser descritos como regiões conectadas de um espaço multidimensional contendo uma densidade de pontos relativamente alta, separada de outras tais regiões por uma região contendo uma densidade relativamente baixa de pontos;

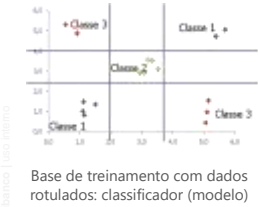
Lembre que algoritmos induzem os clusters

- Os clusters a serem induzidos dependem de uma série de fatores, além dos dados propriamente ditos:
 - medidas de dis(similaridade), índices de avaliação, parâmetros definidos pelo usuário etc.
 - fortemente dependente do domínio / problema
- Na perspectiva de Aprendizado de Máquina (AM) há uma relação com o conceito de bias indutivo:
 - projetista define o que o computador pode aprender
 - existem centenas de algoritmos...

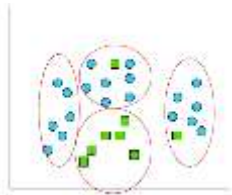
O que não é análise de agrupamento?

- Classificação Supervisionada
 - Disponibilidade de rótulos de classes
- Segmentação Simples
 - Dividir estudantes em diferentes grupos alfabeticamente
- Resultados de uma consulta (query)
 - Grupos são resultado de uma especificação externa

Classificação x Agrupamento



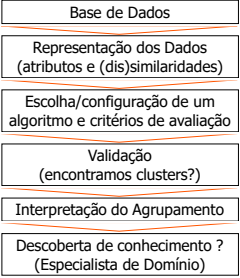
Classificação x Agrupamento



Classificação:
Aprender um método para prever as categorias (classes) de padrões não vistos a partir de exemplos pré-rotulados (classificados)

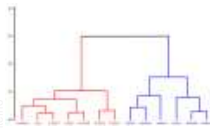
Agrupamento de Dados (Clustering):
Encontrar os rótulos das categorias (grupos ou **clusters**) e possivelmente o número de categorias diretamente a partir dos dados

Ciclo de modelagem em agrupamento



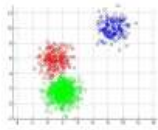
Nota: em qualquer etapa podemos (ter de) retornar a uma etapa anterior.

Tipos de algoritmos de agrupamento



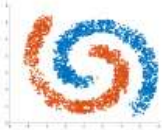
Hierárquico:

- Single Linkage
- Completed Linkage
- Average Linkage
- ...



Particionais:

- K-Means
- K-Median
- K-Medoid
- ...

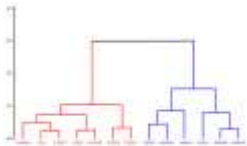


Densidade:

- DBScan
- ...

Agenda

- Motivação e conceitos
- Definições preliminares
- Algoritmos Hierárquicos
- Algoritmos Particionais
- Algoritmos Baseados em Densidade
- Avaliação de agrupamentos
- Extra: Bisset k-Means, EM, ...



Hierarquia: Conceitos Básicos

Hierarquias são comumente usadas para organizar informação

Web-Site Directory - Free, organized by subject. [Suggest new site](#)

Business & Economy
Bus. Finance, Marketing, Adm.

Regional
Australasia, Europe, US, Canada

Computers & Internet
Internet, WWW, Software, Games

Society & Culture
Fashion, Entertainment, Religion

Business & Economy

- HR
- Finance
- Marketing
- Jobs
- Business Application...
- Marketing Research...
- Marketing Approach...
- Human Resources...

Hierarquia: Conceitos Básicos

Exemplo: árvores filogenéticas em biologia

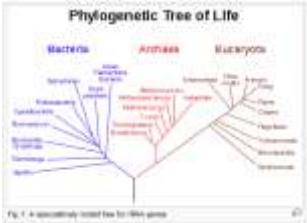
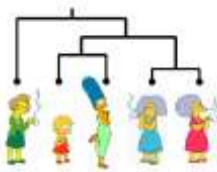


Fig. 1. A phylogenetic tree for life.

http://en.wikipedia.org/wiki/Phylogenetic_tree

Métodos Clássicos para Agrupamento Hierárquico



Bottom-Up (aglomerativos):

- Iniciar colocando cada objeto em um *cluster*
- Encontrar o melhor par de *clusters* para unir
- Unir o par de *clusters* escolhido
- Repetir até que todos os objetos estejam reunidos em um só *cluster*

Top-Down (divisivos):

- Iniciar com todos objetos em um único *cluster*
- Sub-dividir o *cluster* em dois novos *clusters*
- Aplicar o algoritmo recursivamente em ambos, até que cada objeto forme um *cluster* por si só

Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

Métodos Clássicos para Agrupamento Hierárquico

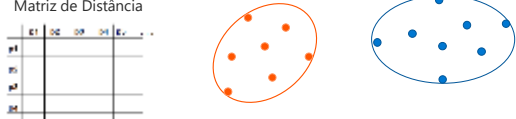
Algoritmos hierárquicos podem operar somente sobre uma matriz de distâncias.



$D(\{1,2\},3) = 1$ $D(\{1,2\},4) = 8$

Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

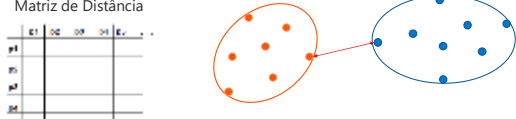
Como definir Inter-Cluster (Dis)similaridade



- MIN
- MAX
- Group Average
- ...

Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

Como definir Inter-Cluster (Dis)similaridade



- MIN
- MAX
- Group Average
- ...

Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

Single Linkage (Florek, 1951)

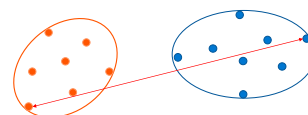
- Dissimilaridade entre clusters é dada pela **menor** dissimilaridade entre 2 objetos (um de cada cluster)
- Originalmente baseado em Grafos: **menor** aresta entre dois vértices de subconjuntos distintos



Como definir Inter-Cluster (Dis)similaridade

Matriz de Distância

	e1	e2	e3	e4	e5
e1					
e2					
e3					
e4					
e5					



- MIN
- MAX
- Group Average
- ...

Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

Complete Linkage (Sorensen, 1948)

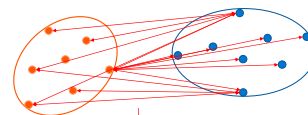
- Dissimilaridade entre clusters é dada pela **maior** dissimilaridade entre 2 objetos (um de cada cluster)
- Originalmente baseado em Grafos: menor aresta entre dois vértices de subconjuntos distintos



Como definir Inter-Cluster (Dis)similaridade

Matriz de Distância

	e1	e2	e3	e4	e5
e1					
e2					
e3					
e4					
e5					



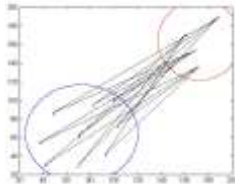
Todas as distâncias para-a-par

- MIN
- MAX
- Group Average
- ...

Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

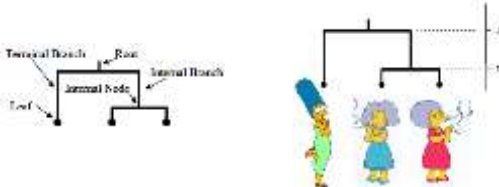
Complete Linkage (Sokal R and Michener C, 1958)

- Dissimilaridade entre clusters é dada pela **distância média** entre cada par de objetos (um de cada cluster)
- Também conhecido como UPGMA – Unweighted Pair Group Method using Arithmetic averages

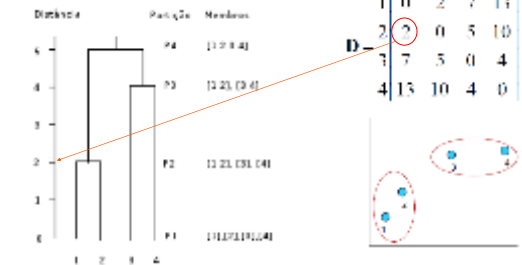


Dendrograma = Hierarquia + Dissimilaridade entre Clusters

A dissimilaridade entre dois clusters (possivelmente **singletons**) é representada como a altura do nó interno mais baixo compartilhado



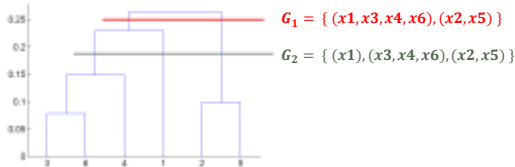
Dendrograma



Dendrograma -> Grupos

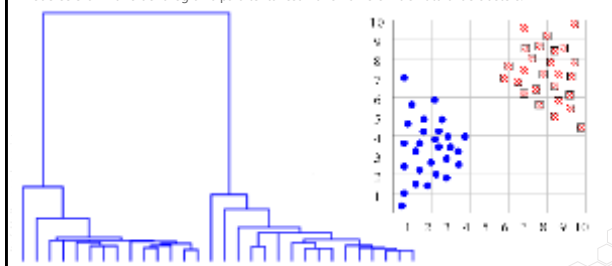
Partições são obtidas via **cortes** no dendrograma

- cortes horizontais
- no. de grupos da partição = no. de interseções



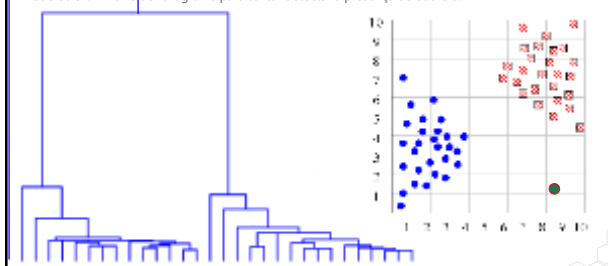
Dendrograma -> Grupos

Pode-se examinar o dendrograma para tentar estimar o número mais natural de clusters.

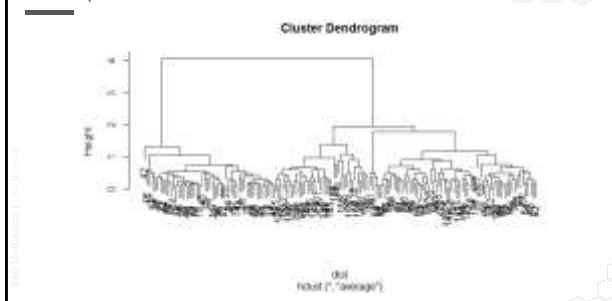


Dendrograma -> Outlier

Pode-se examinar o dendrograma para tentar detectar a presença de outliers.

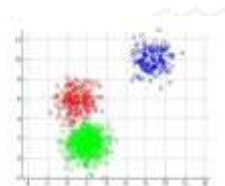


Hierárquico: R



Agenda

- Motivação e conceitos
- Definições preliminares
- Algoritmos Hierárquicos
- Algoritmos Particionais
- Algoritmos Baseados em Densidade
- Avaliação de agrupamentos
- Extra: Bisset k-Means, EM, ...



Métodos Particionais

Métodos *particionais* sem sobreposição referem-se a algoritmos de agrupamento que buscam (explícita ou implicitamente) por uma matriz de partição rígida de um conjunto de objetos **X**

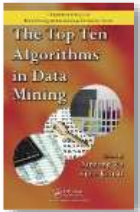
Encontrar uma Matriz de Partição U(X):

Equivale a particionar o conjunto $X = \{x_1, x_2, \dots, x_N\}$ de N objetos em uma coleção $C = \{C_1, C_2, \dots, C_k\}$ de k grupos disjuntos C_i tal que $C_1 \cup C_2 \cup \dots \cup C_k = X$, $C_i \neq \emptyset$ e $C_i \cap C_j = \emptyset$ para $i \neq j$

K-Means

Começaremos nosso estudo com um dos algoritmos mais clássicos da área de mineração de dados em geral

- algoritmo das k-médias ou k-means
- listado entre os Top 10 Most Influential Algorithms in DM
- Wu, X. and Kumar, V. (Editors), **The Top Ten Algorithms in Data Mining**, CRC Press, 2009
- X. Wu et al., "**Top 10 Algorithms in Data Mining**", Knowledge and Info. Systems, vol. 14, pp. 1-37, 2008



K-Means

Referência Mais Aceita como Original:

J. B. MacQueen, Some methods of classification and analysis of multivariate observations, In Proceedings 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, California, USA, 1967, 281–297

Porém...

"K-means has a rich and diverse history as it was independently discovered in different scientific fields by Steinhaus (1956), Lloyd (proposed in 1957, published in 1982), Ball & Hall (1965) and MacQueen (1967)" [Jain, Data Clustering: 50 Years Beyond K-Means, Patt. Rec. Lett., 2010]

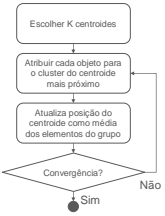
... e tem sido assunto por mais de meio século !

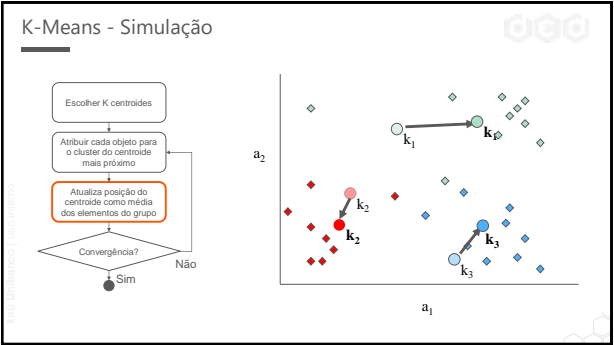
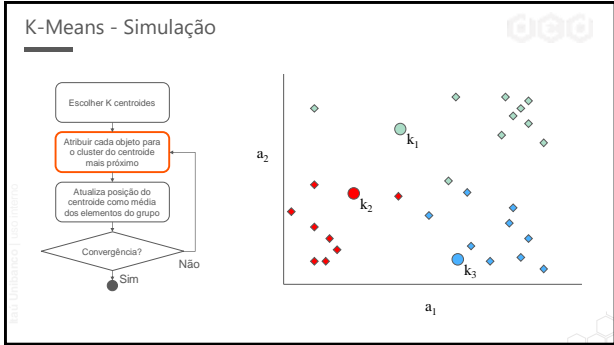
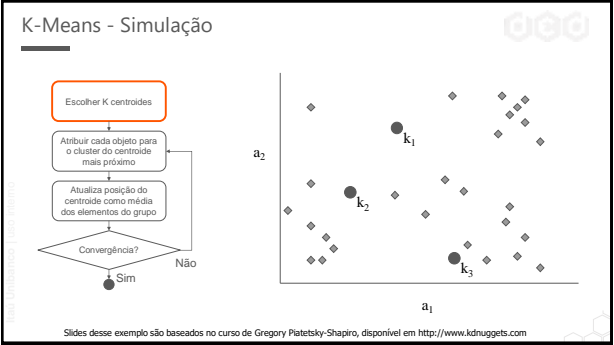
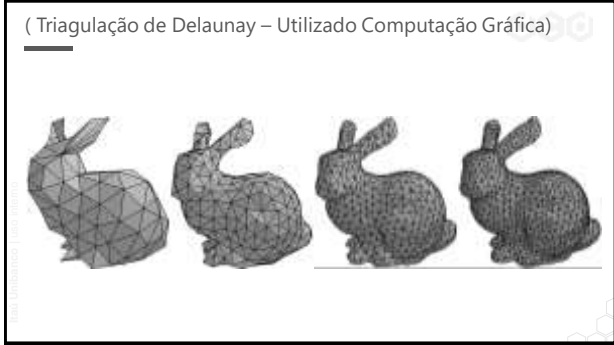
Douglas Steinley, K-Means Clustering: A Half-Century Synthesis, British Journal of Mathematical and Statistical Psychology, Vol. 59, 2006

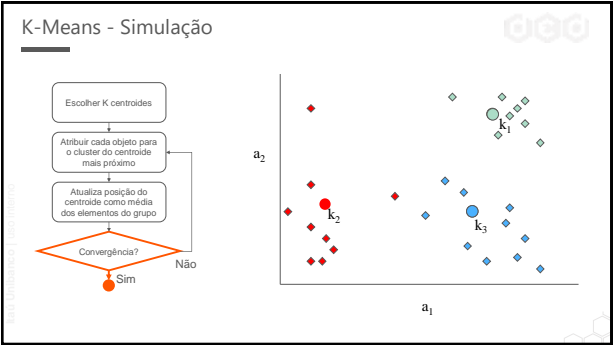
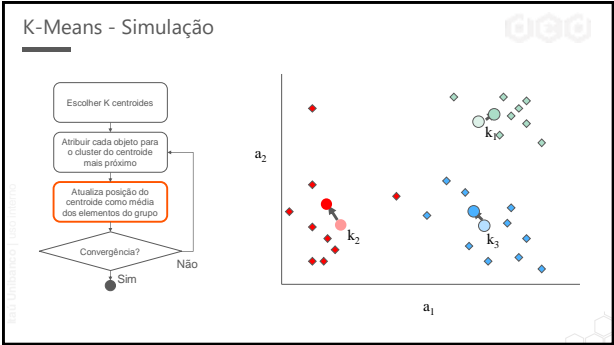
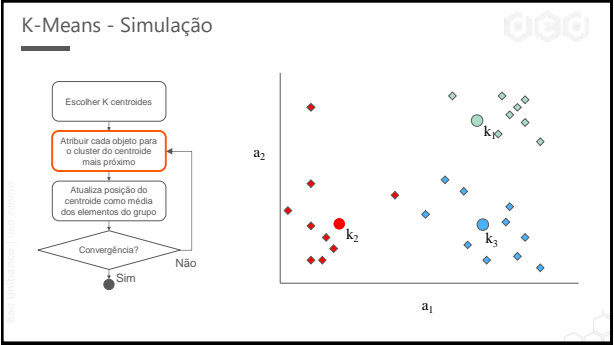
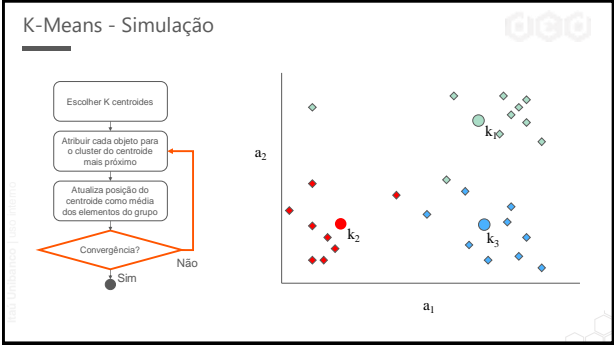
K-Means

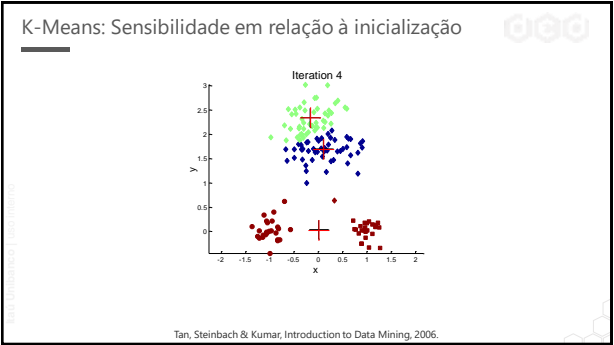
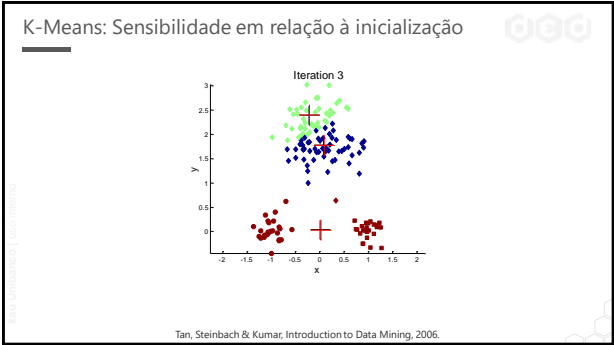
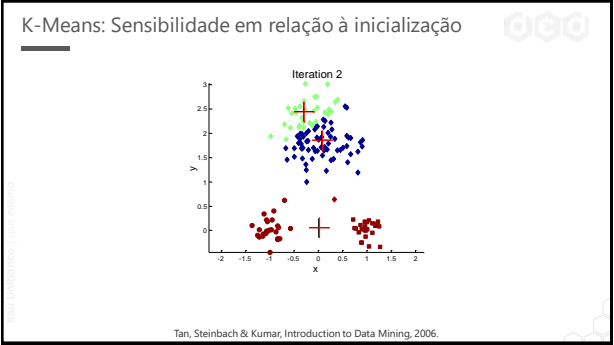
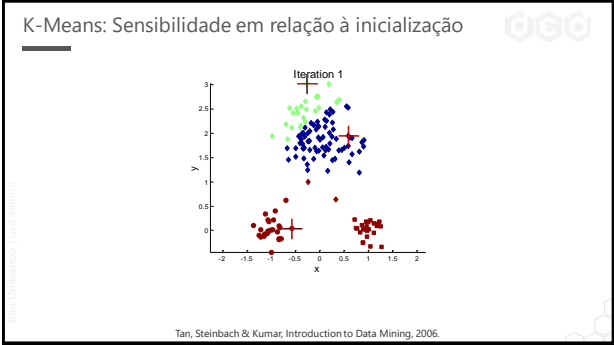
Objetiva particionar N observações dentre k grupos em que cada observação pertence ao grupo mais próximo da média. Isso resulta em uma divisão do espaço de dados em um Diagrama de Voronoi.

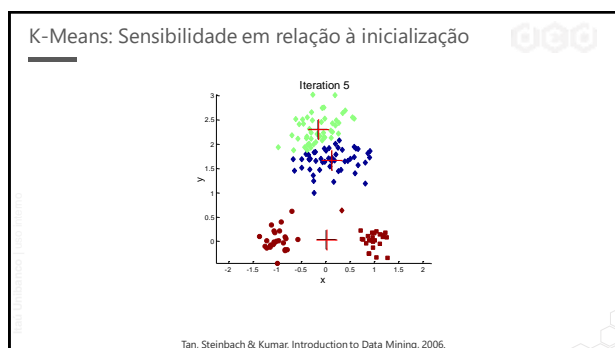
Calculado por meio da triangulação de Delaunay











K-Means: Sensibilidade em relação à inicialização

Premissa:
Uma boa seleção de k protótipos iniciais em uma base de dados com k grupos naturais é tal que cada protótipo é um objeto de um grupo diferente.

- No entanto, a chance de se selecionar um protótipo de cada grupo é pequena, especialmente para k grande.
- Consideremos grupos balanceados, com uma mesma quantidade $g = N / k$ de objetos cada. A probabilidade de selecionar um protótipo de cada grupo diferente é:

$$P = \frac{\text{no. de maneiras de selecionar 1 objeto de cada grupo (N / k objetos)}}{\text{no. de maneiras de selecionar k dentre N objetos}} = \frac{k!}{k^k}$$

Para $k = 10$ temos $P = 0.00036 \rightarrow 2.778$ inicializações.

K-Means: Sensibilidade em relação à inicialização

Múltiplas Execuções (inicializações aleatórias):

- Funciona bem em muitos problemas;
- Pode demandar muitas execuções (especialmente com k alto).

Agrupamento Hierárquico:

- agrupa-se uma amostra dos dados para tomar os centros da partição com k grupos.

Seleção "informada" em uma amostra dos dados:

- Tomar o 1º protótipo como um objeto aleatório ou como o centro dos dados (*grand mean*);
- Sucessivamente escolhe-se o próximo protótipo como o objeto mais distante dos protótipos correntes.

Busca Guiada:

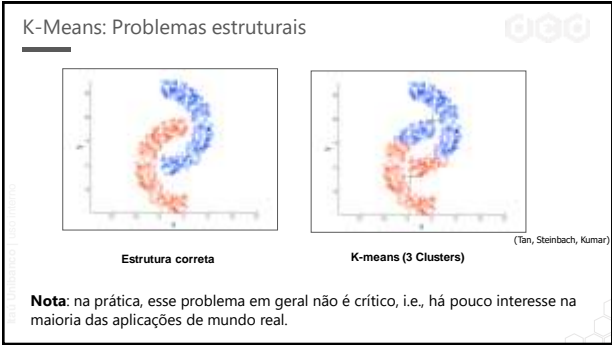
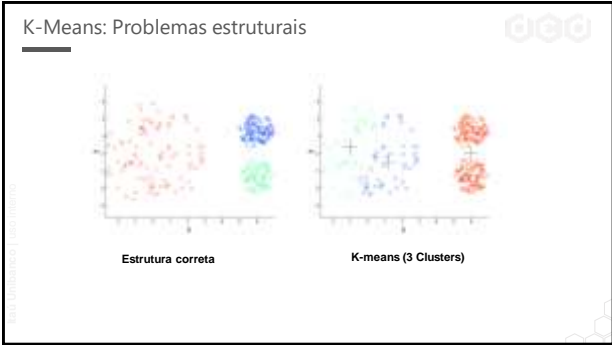
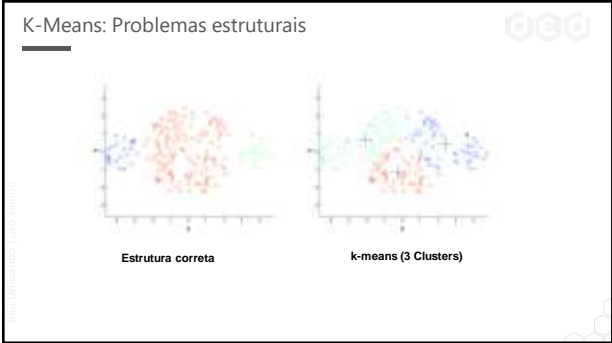
- X-means, k -means evolutivo, ...



K-Means: Problemas estruturais

Algoritmo *k*-means funciona bem se:

- Clusters são (hiper)esféricos e bem separados
- Clusters de volumes aproximadamente iguais
- Cluster com quantidades de pontos semelhantes
- Formas Globulares



K-Means: Custo Computacional

Complexidade (assintótica) de tempo:

$$O(i \cdot K \cdot N \cdot n)$$

- O que isso significa?

O que dizer sobre a constante de tempo?

→ Computar Distância Euclidiana via aproximações sucessivas (Newton-Raphson) custa caro.

Se também tenho problema de espaço em memória...

→ Solução aproximada (*sampling*).

→ Paralelizar (mesmo computador) ou distribuir (e.g., map-reduce) o processamento.

TA BARATO RÁPIDO



PRA CARAMBA

Resumo das (des)vantagens do k-means

Vantagens

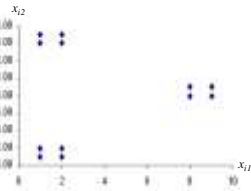
- Simples e intuitivo
- Complexidade **linear** em todas as variáveis críticas
- Eficaz em muitos cenários de aplicação
- Resultados de interpretação simples

Desvantagens

- $k = ?$
- Sensível à inicialização dos protótipos (mínimos locais de J)
- Limita-se a encontrar clusters volumétricos / globulares
- Cada item deve pertencer a um único cluster (**partição rígida**)
- Limitado a atributos numéricos
- Sensível a outliers

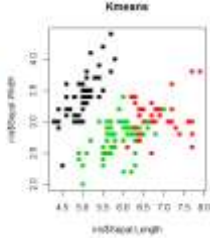
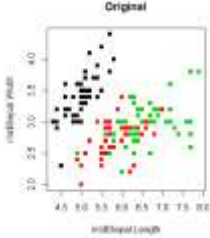
K-Means - Homework

Objeto x _i	x _{1j}	x _{2j}
1	1	2
2	2	1
3	1	1
4	2	2
5	8	9
6	9	8
7	9	9
8	8	8
9	1	15
10	2	15
11	1	14
12	2	14

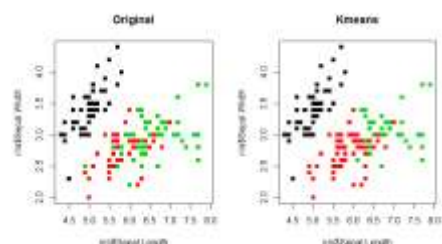


Executar k-means com $k = 3$ nos dados acima a partir dos protótipos [6 6], [4 6] e [5 10]. Quais foram as partições e os centróides obtidos?

K-Means: R



K-Means Inicialização hierárquico: R



K-Medianas

K-medianas: Substituir as médias pelas medianas

- Média de 1, 3, 5, 7, 9 é 5
- Média de 1, 3, 5, 7, 1009 é 205
- Mediana de 1, 3, 5, 7, 1009 é 5

Vantagem: menos sensível a outliers

Desvantagem: implementação mais complexa
cálculo da mediana em cada atributo...

K-Medóides

K-medóides: Substituir cada centróide por um objeto representativo do cluster, denominado **medóide**

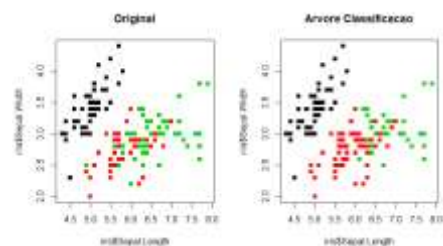
- Medóide = objeto mais próximo aos demais objetos do cluster mais próximo em média (empates resolvidos aleatoriamente)

Vantagens:

- menos sensível a outliers
- permite cálculo relacional (apenas matriz de distâncias)
 - logo, pode ser aplicado a bases com atributos categóricos
- convergência assegurada com qualquer medida de (dis)similaridade

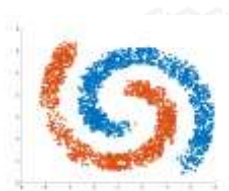
Desvantagem: Complexidade quadrática com no. de objetos (N)

K-Means Implantação: R



Agenda

- Motivação e conceitos
- Definições preliminares
- Algoritmos Hierárquicos
- Algoritmos Particionais
- Algoritmos Baseados em Densidade
- Avaliação de agrupamentos
- Extra: Bisect k-Means, EM, ...



Algoritmos Baseados em Densidade

Paradigma de Agrupamento por Densidade

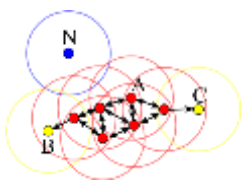
- Clusters como regiões de alta concentração de objetos separadas por regiões de baixa concentração de objetos
- Paradigma alternativo àquele baseado em protótipos: K-means e variantes, EM, etc

Existem vários algoritmos, veremos a seguir um dos mais conhecidos: **DBSCAN**



DBScan: definições

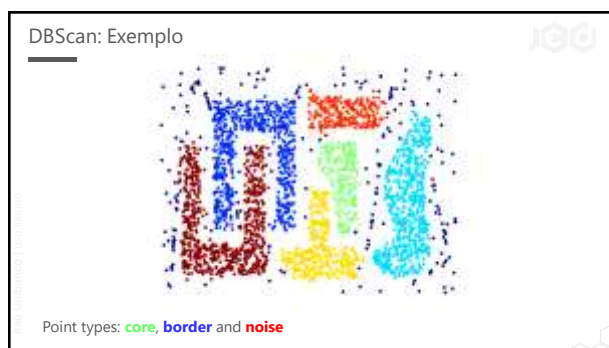
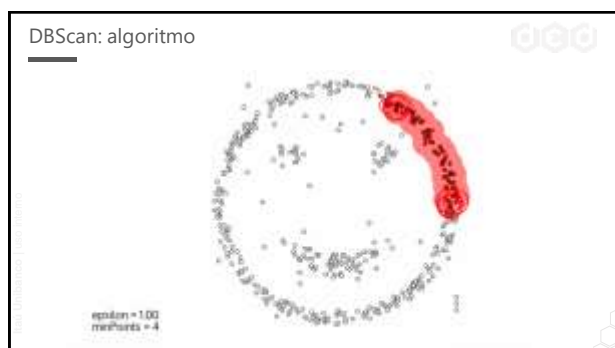
- A point is a **core** point if it has at least a specified number of points (MinPts) within the radius Eps (including the point itself)
 - These are points that are in the interior of a cluster
- A **border** point has fewer than MinPts within Eps, but is in the neighborhood (within the radius) of at least 1 core point
- A **noise** point is neither a core point nor a border point



DBScan: algoritmo

Algoritmo Conceitual:

1. Percorra a BD e rotule os objetos como core, border ou noise
2. Elimine aqueles objetos rotulados como **noise**
3. Insira uma aresta entre cada par de objetos **core** vizinhos
 - 2 objetos são vizinhos se um estiver dentro do raio Eps do outro
4. Faça cada componente conexo resultante ser um cluster
5. Atribua cada **border** ao cluster de um de seus core associados
 - Resolva empates se houver objetos core associados de diferentes clusters



Resumo das (des)vantagens do DBScan

Vantagens	Desvantagens
<ul style="list-style-type: none"> • Não necessita do número de clusters a priori • Consegue encontrar clusters com formatos arbitrários • Tem uma definição de ruído e é robusto a outliers • Necessita de apenas dois parâmetros: <ul style="list-style-type: none"> • Raio • Número de vizinhos para virar core (minpts) 	<ul style="list-style-type: none"> • Extremamente sensível aos parâmetros Raio e minPts • Depende da distância utilizada para determinar se um ponto está ou não presente dentro do raio. (tipicamente se utiliza euclidiana) • Não consegue clusterizar dados com grupos com grandes diferenças de densidades • Se a escala dos dados não for conhecida, determinar o raio pode ser difícil

Agenda

<ul style="list-style-type: none"> • Motivação e conceitos • Definições preliminares • Algoritmos Hierárquicos • Algoritmos Particionais • Algoritmos Baseados em Densidade • Avaliação de agrupamentos 	<ul style="list-style-type: none"> • Extra: Bisset k-Means, EM, ...
---	--

Validação

*"The validation of clustering structures is the most difficult and frustrating part of cluster analysis. Without a strong effort in this direction, cluster analysis will remain a **black art** accessible only to those true believers who have experience and great courage."*

(Jain and Dubes, *Algorithms for Clustering Data*, 1988)



Validação

- Refere-se, de forma ampla, aos diferentes procedimentos para avaliar de maneira objetiva e quantitativa os resultados de análise de agrupamento.
- Cada um desses procedimentos pode nos ajudar a responder uma ou mais questões do tipo:
 - Encontramos grupos de fato?
 - grupos são pouco usuais ou facilmente encontrados ao acaso?
 - Qual a qualidade (relativa ou absoluta) dos grupos encontrados?
 - Qual é o número natural / mais apropriado de grupos?

Índices de validação

A maneira quantitativa para validação é alcançada através de algum tipo de **índice**. Há 3 tipos de **índices/critérios de validade**:

- **Internos:** Avalia o grau de compatibilidade entre a estrutura de grupos sob avaliação e os dados, usando apenas os próprios dados.
- **Relativos:** Avaliam qual dentre duas ou mais estruturas de grupos é melhor sob algum aspecto. Tipicamente são critérios internos capazes de quantificar a qualidade relativa.
- **Externos:** Avalia o grau de correspondência entre a estrutura de grupos (partição ou hierarquia) sob avaliação e informação a priori na forma de uma solução de agrupamento esperada ou conhecida.

Índices de validação: Internos

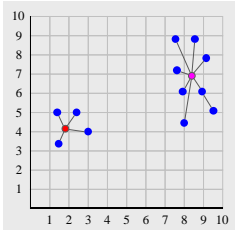
A maneira quantitativa para validação é alcançada através de algum tipo de **índice**. Há 3 tipos de **índices/critérios de validade**:

- **Internos:** Avalia o grau de compatibilidade entre a estrutura de grupos sob avaliação e os dados, usando apenas os próprios dados.
- **Relativos:** Avaliam qual dentre duas ou mais estruturas de grupos é melhor sob algum aspecto. Tipicamente são critérios internos capazes de quantificar a qualidade relativa.
- **Externos:** Avalia o grau de correspondência entre a estrutura de grupos (partição ou hierarquia) sob avaliação e informação a priori na forma de uma solução de agrupamento esperada ou conhecida.

Índices de validação: Internos – Erro Quadrático

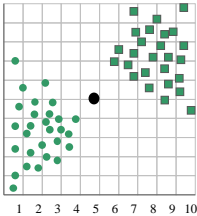
$$J = \sum_{i=1}^k \sum_{\mathbf{x}_j \in C_i} d(\mathbf{x}_j, \bar{\mathbf{x}}_i)^2$$

Função Objetivo



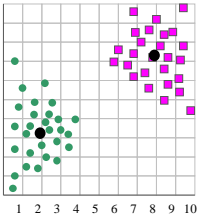
Exemplo de Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community, SBBD 2003, Manaus.

Índices de validação: Internos – Erro Quadrático



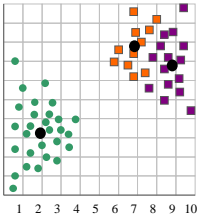
Para $k = 1$, o valor da função objetivo é 873

Índices de validação: Internos – Erro Quadrático



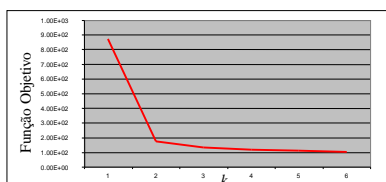
Para $k = 2$, o valor da função objetivo é 173

Índices de validação: Internos – Erro Quadrático



Para $k = 3$, o valor da função objetivo é 134

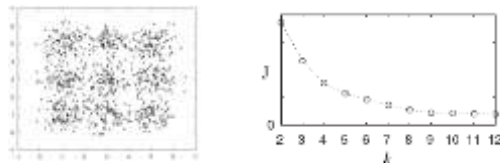
Índices de validação: Internos – Erro Quadrático



Podemos então repetir este procedimento e plotar os valores da função objetivo J para $k = 1, \dots, 6, \dots$ e tentar identificar um "joelho"

Índices de validação: Internos – Erro Quadrático

Infelizmente os resultados não são sempre tão claros quanto no exemplo anterior:



Outras alternativas para lidar com o problema de se estimar o número de clusters?

- Índices de validade relativos...

Índices de validação: Relativo

A maneira quantitativa para validação é alcançada através de algum tipo de **índice**. Há 3 tipos de **índices/critérios de validade**:

- **Internos**: Avalia o grau de compatibilidade entre a estrutura de grupos sob avaliação e os dados, usando apenas os próprios dados.
- **Relativos**: Avaliam qual dentre duas ou mais estruturas de grupos é melhor sob algum aspecto. Tipicamente são critérios internos capazes de quantificar a qualidade relativa.
- **Externos**: Avalia o grau de correspondência entre a estrutura de grupos (partição ou hierarquia) sob avaliação e informação a priori na forma de uma solução de agrupamento esperada ou conhecida.

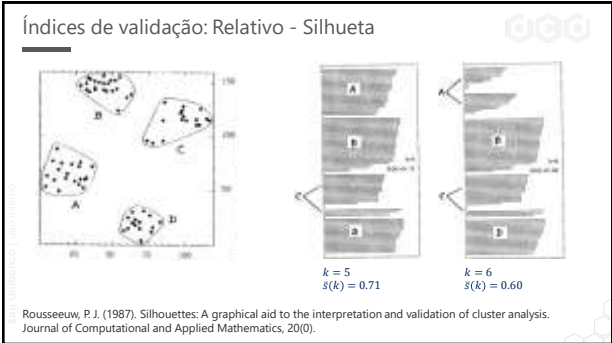
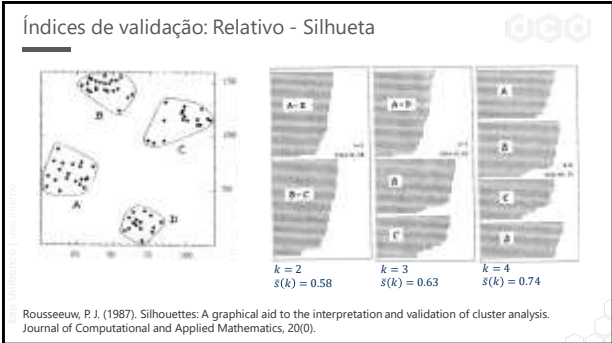
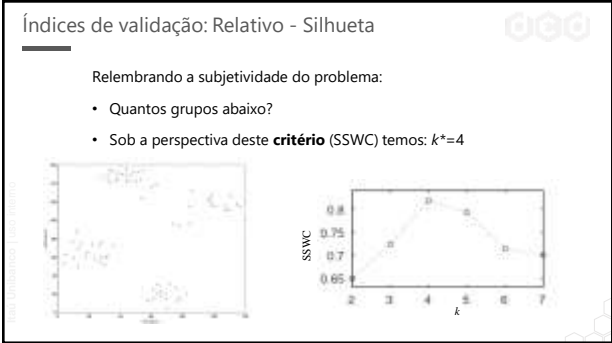
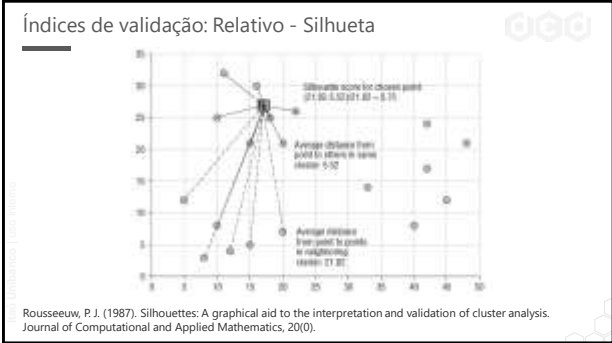
Índices de validação: Relativo - Silhueta

Métrica utilizada para mensurar a qualidade do agrupamento priorizando grupos densos/concisos e separados.

$$s_{xj} = \frac{b_{p,j} - a_{p,j}}{\max\{b_{p,j}, a_{p,j}\}}$$

Considere que o j -ésimo objeto, x_j , de um data set pertence a um cluster $p \in \{1, \dots, k\}$

- $a_{p,j}$ - distância média do objeto j a todos os outros objetos pertencentes ao mesmo cluster p
- $d_{q,j}$ - distância média do objeto j a todos os outros objetos pertencentes a clusters distintos q em que $q \neq p$
- $b_{p,j}$ - o menor valor de $d_{q,j}$ computado para $q = 1, \dots, k$ em que $q \neq p$. Representa a distância entre o objeto x_j e o cluster vizinho mais próximo



Índices de validação: Relativo – Silhueta Simplificada

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$
$$SWC = \frac{1}{N} \sum_{i=1}^N s(i)$$

Silhueta Simplificada: $a(i)$ e $b(i)$ são calculados como a distância do i -ésimo objeto ao centróide do cluster em questão - $O(N)$.

E. R. Hruschka, L. N. de Castro and R. J. G. B. Campello, "Evolutionary algorithms for clustering gene-expression data," Data Mining, 2004. ICDM '04. Fourth IEEE International Conference on, 2004, pp. 403-406. doi: 10.1109/ICDM.2004.10073
URL: <http://ieeexplore.ieee.org/Stamp/Stamp.asp?document=3810231&number=20463>

Índices de validação: Externos

A maneira quantitativa para validação é alcançada através de algum tipo de **índice**. Há 3 tipos de **índices/critérios de validade**:

- **Internos:** Avalia o grau de compatibilidade entre a estrutura de grupos sob avaliação e os dados, usando apenas os próprios dados.
- **Relativos:** Avaliam qual dentre duas ou mais estruturas de grupos é melhor sob algum aspecto. Tipicamente são critérios internos capazes de quantificar a qualidade relativa.
- **Externos:** Avalia o grau de correspondência entre a estrutura de grupos (partição ou hierarquia) sob avaliação e informação a priori na forma de uma solução de agrupamento esperada ou conhecida.

Índices de validação: Externos

Estudaremos os índices mais usados (Rand e Jaccard). Adotaremos a seguinte terminologia:

- grupos da **partição de referência** (*golden truth*) → "**classes**"
- grupos da **partição sob avaliação** → **clusters (grupos)**

Podemos então definir as grandezas de interesse:

a: No. de pares da mesma classe e do mesmo cluster

b: No. de pares da mesma classe e de clusters distintos

c: No. de pares de classes distintas e do mesmo cluster

d: No. de pares de classes e clusters distintos

Índices de validação: Externos – Rand Index

$$RI = \frac{a+d}{a+b+c+d}$$



Referências

1. Jain, A. K. and Dubes, R. C., Algorithms for Clustering Data, Prentice Hall, 1988
2. Everitt, B. S., Landau, S., and Leese, M., Cluster Analysis, Arnold, 4th Edition, 2001.
3. Tan, P.-N., Steinbach, M., and Kumar, V., Introduction to Data Mining, Addison-Wesley, 2006
4. Kaufman, L., Rousseeuw, P. J., Finding Groups in Data – An Introduction to Cluster Analysis, Wiley, 2005.
5. Wu, X. and Kumar, V., The Top Ten Algorithms in Data Mining, Chapman & Hall/CRC, 2009
6. D. Steinley, K-Means Clustering: A Half-Century Synthesis, British J. of Mathematical and Stat. Psychology, V. 59, 2006

Agenda

- Motivação e conceitos
- Definições preliminares
- Algoritmos Hierárquicos
- Algoritmos Particionais
- Algoritmos Baseados em Densidade
- Avaliação de agrupamentos
- Extra: Bisect k-Means, EM, ...

Agenda

- Motivação e conceitos
- Definições preliminares
- K-means
- Estimando o número de clusters a partir dos dados
- Bisecting K-means
- K-medoids
- EM para misturas de Gaussianas
- Avaliação de agrupamentos

Bisecting k -means (particional-hierárquico):

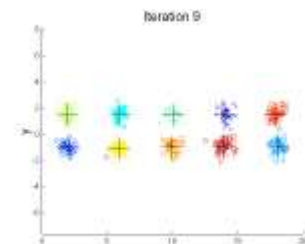
Rekursivamente particiona a base de dados em dois grupos, gerando uma "árvore de partições". Lembrar que:

$$P = \frac{\text{no. de maneiras de selecionar 1 objeto de cada grupo (N / k objetos)}}{\text{no. de maneiras de selecionar k dentre N objetos}} = \frac{k!}{k^N}$$

1. Initialize the list of clusters to contain the cluster containing all points.
2. repeat:
 - a. Select a cluster from the list of clusters.
 - b. for $i = 1$ to number of iterations do:
 1. Bisect the selected cluster using basic K-means
 - c. end for
 - d. Add the two clusters from the bisection with the lowest SSE to the list of clusters.
3. until: Until the list of clusters contains K clusters.

$$SSE(C_i) = \sum_{x_j \in C_i} d(x_j, \bar{x}_i)^2 \rightarrow \text{Sum of Squared Errors (para } C_i)$$

Exemplo:



Tan, Steinbach & Kumar, Introduction to Data Mining, 2006.

Notas sobre Bisecting k -means:

- Note que fazendo $k = N$ (no. total de objetos) no passo 8 do algoritmo, obtemos uma hierarquia completa.
- No passo 3, a seleção do grupo a ser bi-seccionado pode ser feita de diferentes maneiras, por exemplo usando outro critério de avaliação de qualidade dos grupos, para eleger o "pior":
 - Diâmetro máximo (sensível a *outliers*).
 - SSE normalizado pelo no. de objetos do grupo (mais robusto).

Agenda

- Motivação e conceitos
- Definições preliminares
- K-means
- Estimando o número de clusters a partir dos dados
- Bisecting K-means
- K-medoids
- EM para misturas de Gaussianas
- Avaliação de agrupamentos



k-medoids

- Substituir centróide por um objeto representativo (*medoid*);
 - Medoid é o objeto mais próximo aos demais objetos do grupo - mais próximo em média (empates resolvidos aleatoriamente);
- Menos sensível a *outliers*;
- Permite cálculo relacional (requer apenas matriz de distâncias);
- Pode ser aplicado a bases com atributos categóricos;
- Converge com qualquer medida de (dis)similaridade
- Complexidade quadrática com n°. de objetos (N)

Agenda

- Motivação e conceitos
- Definições preliminares
- K-means
- Estimando o número de clusters a partir dos dados
- Bisecting K-means
- K-medoids
- EM para misturas de Gaussianas
- Avaliação de agrupamentos



EM para mistura de Gaussianas

- O Algoritmo **EM** (*Expectation Maximization*) é um procedimento genérico para a modelagem probabilística de um conjunto de dados;
- Basicamente, **EM** otimiza os parâmetros de uma função de distribuição de probabilidades (p.d.f.) de forma que esta represente os dados da forma mais verossímil possível;
- Modelo mais utilizado: **Mistura de Gaussianas**

GMM (Gaussian Mixture Model)

Um GMM é representado pela *p.d.f.*:

$$p(\mathbf{x}) = \sum_{i=1}^k \pi_i \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$$

Diagram illustrating the components of the Gaussian Mixture Model (GMM) equation:

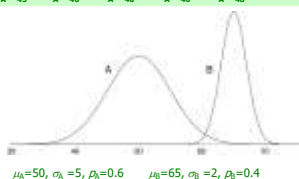
- Número de componentes** (Number of components) points to k .
- Coefficientes da mistura** (Mixture coefficients) points to π_i .
- Gaussiana com a mesma dimensão dos objetos** (Gaussian with the same dimension as the objects) points to $\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$.
- Centro da i-ésima Gaussiana (vetor da mesma dimensão de \mathbf{x})** (Center of the i -th Gaussian (vector of the same dimension as \mathbf{x})) points to $\boldsymbol{\mu}_i$.
- Matriz de covariância da i-ésima Gaussiana** (Covariance matrix of the i -th Gaussian) points to $\boldsymbol{\Sigma}_i$.

Exemplo

Objetos:

A	51	B	62	B	64	A	48	A	39	A	51
A	43	A	47	A	51	B	64	B	62	A	48
B	62	A	52	A	52	A	51	B	64	B	64
B	64	B	64	B	62	B	63	A	52	A	42
A	45	A	51	A	49	A	43	B	63	A	48
A	42	B	65	A	48	B	65	B	64	A	41
A	46	A	48	B	62	B	65	A	48		
A	45	A	49	A	43	B	65	B	64		
A	45	A	46	A	40	A	46	A	48		

Modelo:



Witten & Frank, Data Mining: Practical Machine Learning Tools and Techniques (Chapter 6)

Dado $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ de N observações *i.i.d* temos:

$$p(\mathbf{X}) = p(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N) = \prod_{j=1}^N p(\mathbf{x}_j) = \prod_{j=1}^N \sum_{i=1}^k \pi_i \mathcal{N}(\mathbf{x}_j | \mathbf{v}_i, \Sigma_i)$$

Por conveniência matemática, utiliza-se da **log-verossimilhança**:

$$\ln(p(\mathbf{X} | \boldsymbol{\pi}, \boldsymbol{\Sigma}, \mathbf{v})) = \sum_{j=1}^N \ln \left(\sum_{i=1}^k \pi_i \mathcal{N}(\mathbf{x}_j | \mathbf{v}_i, \Sigma_i) \right)$$

➤ Maximizar a verossimilhança pode ser visto como maximizar a compatibilidade entre as N observações e o modelo.

- EM (Dempster et al., 1977) é um algoritmo de otimização que visa maximizar a (log) verossimilhança em dois passos:

- **Passo E (Expectation)**

- Avalia as probabilidades a posteriori μ_{ij} ($i = 1, \dots, k; j = 1, \dots, N$) a partir das N observações $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ e do modelo corrente, dado pelos parâmetros $\boldsymbol{\Sigma} = \{\boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_k\}$, $\mathbf{v} = \{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ e $\boldsymbol{\pi} = \{\pi_1, \dots, \pi_k\}$.

- **Passo M (Maximization)**

- Ajusta os parâmetros do modelo visando maximizar a log-verossimilhança.

Passos E e M

E: computar μ_{ij} ($i = 1, \dots, k; j = 1, \dots, N$)

$$\mu_{ij} = \frac{\pi_i \mathcal{N}(\mathbf{x}_j | \mathbf{v}_i, \Sigma_i)}{\sum_{i=1}^k \pi_i \mathcal{N}(\mathbf{x}_j | \mathbf{v}_i, \Sigma_i)}$$

$$\mathcal{N}(\mathbf{x}_j | \mathbf{v}_i, \Sigma_i) = \frac{1}{(2\pi)^{n/2} \det(\Sigma_i)^{1/2}} \exp \left\{ -\frac{1}{2} (\mathbf{x}_j - \mathbf{v}_i)^T \Sigma_i^{-1} (\mathbf{x}_j - \mathbf{v}_i) \right\}$$

$$\text{M: computar } \left\{ \begin{array}{l} \mathbf{v}_i = \frac{1}{N_i} \sum_{j=1}^N \mu_{ij} \mathbf{x}_j \quad \rightarrow \text{centróide ponderado} \\ \boldsymbol{\Sigma}_i = \frac{1}{N_i} \sum_{j=1}^N \mu_{ij} (\mathbf{x}_j - \mathbf{v}_i) (\mathbf{x}_j - \mathbf{v}_i)^T \quad \rightarrow \text{covariância ponderada} \\ \pi_i = \frac{N_i}{N} \quad \rightarrow \text{Coeficientes = prob. a priori do } i\text{-ésimo componente} \\ N_i = \sum_{j=1}^N \mu_{ij} \quad \rightarrow \text{Nº efetivo de pontos atribuídos ao } i\text{-ésimo grupo} \end{array} \right.$$

Algoritmo EM

1. Inicialização (via k -means)

- protótipos v_i = centróides finais do k -means
- covariâncias Σ_i = matrizes de covariância dos grupos
- probabilidades μ_{ij} (para N_i e π_i) = matriz de partição final

2. Passo E

3. Passo M

4. Avaliação do Critério de Parada (função de log-verossimilhança)

5. Interrupção ou Retorno ao Passo 2

EM x k -means

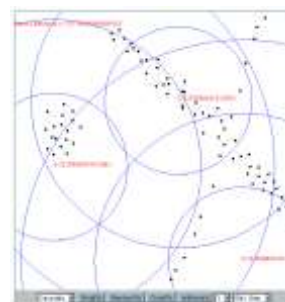
- EM produz informação muito mais rica sobre os dados (probabilidades associadas a cada objeto e cluster);
- Probabilidades produzidas por EM podem facilmente ser convertidas em uma partição rígida;
 - Essa partição é capaz de representar clusters alongados, elipsoidais, com atributos correlacionados;
- No entanto, todas as vantagens acima vêm com um elevado custo computacional associado:
 - Cálculo das Normais Multi-Dimensionais demanda as inversas das matrizes de covariância $\Sigma_i^{-1} \sim O(n^3)$;
 - k -means é um caso particular de EM. Ambos estão sujeitos a mínimos locais.

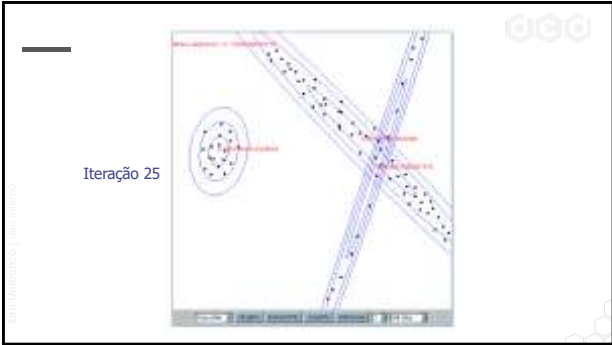
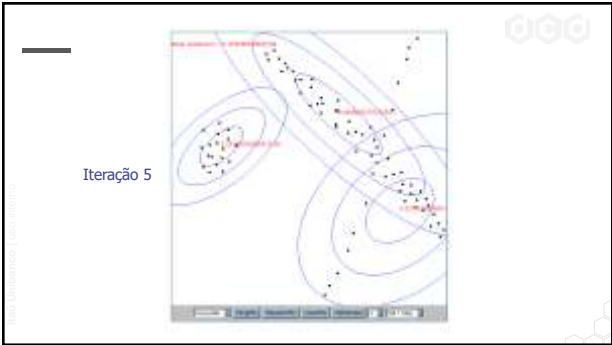
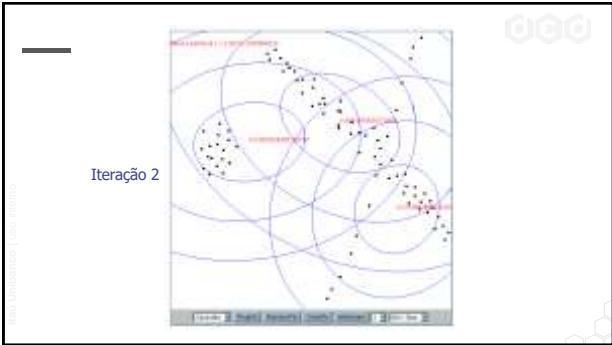
Rodando o EM (exemplo)



Fonte do exemplo: Keogh, E. A Gentle Introduction to Machine Learning and Data Mining for the Database Community. SBBD 2003, Marau.

Iteração 1



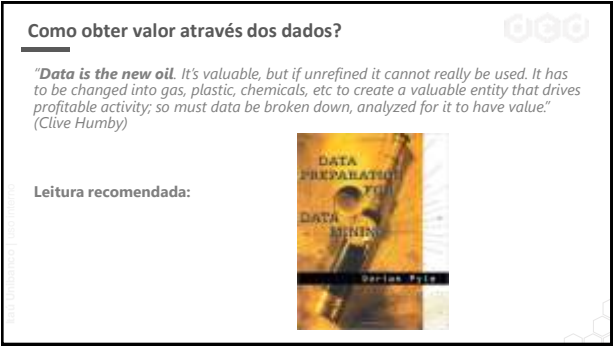
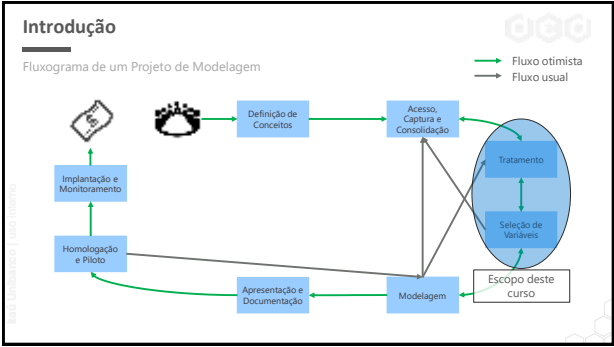
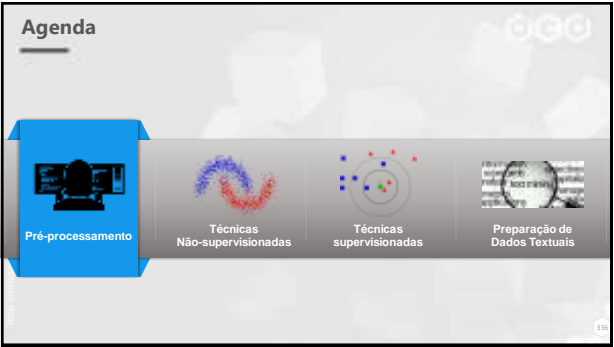


Exercício

Execute iterações do EM ($n = 1$, $N = 10$), com $k = 2$. Tome protótipos iniciais arbitrários e os demais parâmetros inicializados a partir destes, de maneira análoga à inicialização via k -means.

Ilustre o resultado obtido de forma gráfica.

Objeto	x
1	-1.31
2	-0.43
3	0.34
4	3.57
5	2.76
6	0.30
7	9.06
8	4.45
9	2.87
10	4.42



Pré Processamento

Dados reais em geral são:

- **incompletos**: faltam valores e/ou atributos (salário = " ").
 - **ruidosos**: dados com erros e/ou outliers (salário = -150).
 - **inconsistentes**: apresentam discrepâncias em códigos e nomes (1→A, 2→B, 3→C).
- Problemas causados por humanos, softwares, problemas de hardware, dados de diferentes fontes etc.
- Lembrar dos princípios gerais GIGO.
- Preparar os dados pode consumir mais de 80% do esforço de modelagem.

NYU - Data Mining: Concepts and Techniques

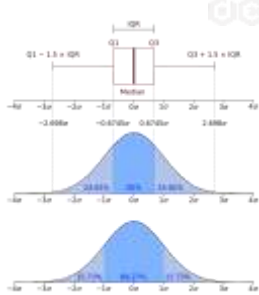
Atividades Comuns

- Limpeza de dados: lidar com valores ausentes, dados ruidosos, outliers, inconsistências etc.
- Integração de dados de múltiplos arquivos e bases.
- Transformação de dados: normalização e agregação.
- Redução de dados: menos dados com mesmos resultados analíticos (seleção de atributos e de amostras).
- **Provocação: como definir o tamanho da amostra?**
- Discretização: compactar informação.
- Nunca deixar de fazer análise exploratória (médias, medianas, variâncias, min, max, gráficos etc).

NYU - Data Mining: Concepts and Techniques

Análise Exploratória

- Média
- Mediana
- Desvio padrão
- Intervalo Inter-quartil
- Proporção
- Preenchimento



Uspensky, N.G. (2001) [1996], "Density of a probability distribution", in: Hazewinkel, Michiel, Encyclopedia of Mathematics, Springer Science+Business Media B.V. / Kluwer Academic Publishers, ISBN 978-1-55608-010-4

Valores Ausentes

- **Ocorrência comum:**
- Mau funcionamento de dispositivos de coleta de dados;
 - Dado omitido pela fonte de informação numa pesquisa;
 - Falha na digitação ou na composição da base;
- **Formas de eliminação de valores ausentes:**
- Eliminar registros/atributos com valores ausentes;
 - Perda de dados pode ser considerável.
 - Preenchimento de valores (imputação)
 - Por uma constante (e.g., média/moda do atributo).
 - Como altera a distribuição?
 - Desconsidera a relação entre atributos da base de dados.
 - Por valores que tentem preservar as relações entre atributos da base de dados.
 - Uso de um algoritmo de aprendizado.

Noção Intuitiva sobre padrões de ausência

- Completamente aleatória (Missing Completely at Random – **MCAR**).
- Aleatória (Missing at Random – **MAR**)
 - Ausência de valor num atributo depende de valores de outro(s) atributo(s).
- Não aleatória (Missing not at Random – **MNAR**).
 - Ausência de um valor num atributo relacionada a uma condição envolvendo o próprio valor do atributo.

Exemplo - pressão arterial de pacientes:
X: Medidas em Janeiro
Y: Medidas em Fevereiro;
Completo (All): todos pacientes
MCAR: pacientes escolhidos ao acaso
MAR: pacientes com pressão < 140 em Jan
MNAR: pacientes com pressão < 140 em Fev

X	Y			
	All	MCAR	MAR	MNAR
169	148	148	148	148
126	123	-	-	-
132	149	-	-	149
160	169	-	169	169
105	138	-	-	-
118	102	-	-	-
113	150	-	-	150
109	96	-	-	-
106	148	-	-	148
176	137	-	137	-
138	155	-	-	155
131	131	-	-	-
130	101	101	-	-
145	155	-	155	155
136	140	-	-	-
146	134	-	134	-
111	129	-	-	-
97	85	85	-	-
124	124	124	-	-
153	112	-	112	-
137	122	122	-	-

Valores Ausentes

Tratamento de dados faltantes: **Imputação da média**

NaN = Not a Number
Indica ausência de dados

Tabela original, com dados faltantes

var1	var2	var3
1	NaN	-4
2	4	NaN
NaN	5	1

Calcula a média dos valores presentes

var1	var2	var3
1	NaN	-4
2	4	NaN
NaN	8	1
Média	1.5	-1.5

Obs: Não considerar os NaNs para o cálculo

Substitui os identificadores NaNs com as médias

var1	var2	var3
1	6	-4
2	4	-1.5
1.5	5	1

Slide de Fernando Beserra

Valores Ausentes

Tratamento de dados faltantes: **Imputação do valor mais frequente**

Tabela original, com dados faltantes

var1	var2	var3
5	NaN	-4
7	4	NaN
NaN	5	1
7	5	1
8	12	0
7	6	2

Calcula o valor mais frequente

	var1	var2	var3
	5	NaN	-4
	7	4	NaN
	NaN	5	1
	7	5	1
	8	12	0
	7	6	2
Valor mais frequente	7	5	1

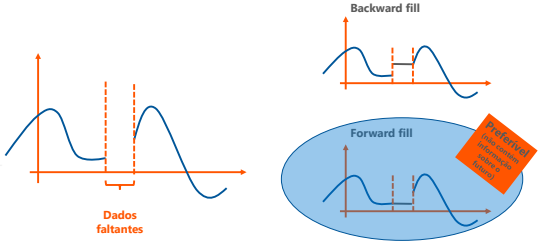
Substitui os identificadores NaNs com os valores obtidos

var1	var2	var3
5	5	-4
7	4	1
7	5	1
7	5	1
8	12	0
7	6	2

Slide de Fernando Beserra

Valores Ausentes

Tratamento de dados faltantes: **Padding de séries temporais**



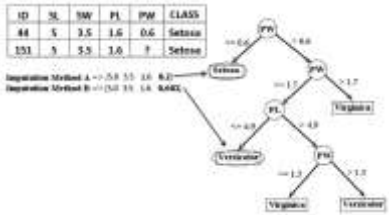
Slide de Fernando Beserra

Duas abordagens para avaliar imputações

- a) **Predição:**
- Comparar valor imputado com valor conhecido;
 - Qual métrica poderia ser usada?
 - Viável em aplicações práticas?
 - Avaliação em dados completos diminui a informação disponível para avaliação da ferramenta de imputação.
- b) **Modelagem:**
- Minimizar a influência na classificação, nas partições etc.
 - Vejamos um exemplo ilustrativo...

Cuidado ao avaliar algoritmos de imputação

- É preciso considerar a tarefa de modelagem (e.g., classificação);
- Imputação causa a falsa sensação de que valor passa a ser conhecido;



Abordagem simples para clustering

- Utilizar distância tolerante a ausentes;
- Exemplo para distância euclidiana:

$D_{ij} / \text{Atrib.}$	A_1	A_2	A_3	A_4
x_1	2	-1	?	0
x_2	7	0	-4	8
x_3	?	3	5	2
x_4	?	10	?	5

Exercício: calcule todos os pares de distâncias.

➢ Caso se opte por imputações, qual deveria ser a abordagem a ser adotada?

Bases Desbalanceadas



Photo by MichaEli

Bases Desbalanceadas

Já tentei:

- Coletar mais dados
- Mudar a métrica (ROC, Precision, Recall, F1-score, ...)
- Diferentes algoritmos

Amostragem:


- Over-sampling: Aumentar número de observações da classe minoritária
 - Amostra com reposição
- Under-sampling: Diminuir número de observações da classe majoritária
 - Todos da classe minoritária + conjunto aleatório da classe majoritária

Bases Desbalanceadas


Outra abordagem:

- Gerar clusters
- Gerar dados sintéticos
 - N. V. Chawla, K. W. Bowyer, L. O. Hall and W. P. Kegelmeyer (2002) "SMOTE: Synthetic Minority Over-sampling Technique", Volume 16, pages 321-357
- Tentar modelos penalizados
 - Ex: Elasticnet
- Técnicas novas
 - Ex: Detecção de Anomalias


Agenda




Pré-processamento



Técnicas Não-supervisionadas

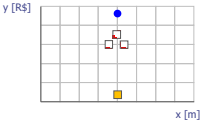


Técnicas supervisionadas

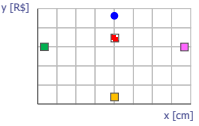


Preparação de Dados Textuais

Preparação para aprendizado não supervisionado



x [m]

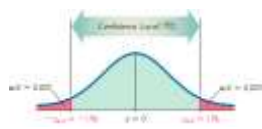


x [cm]

- Pode-se lidar com tais problemas por meio do que usualmente se denomina **normalização/padronização**.
- Vamos rever as formas de normalização mais comuns.

Normalizações comuns

- Reescala Linear [0,1]:
$$l_{ij} = \frac{x_{ij} - \min(a_j)}{\max(a_j) - \min(a_j)}$$
- Padronização por escore z:
$$z_{ij} = \frac{x_{ij} - \mu_{a_j}}{\sigma_{a_j}}$$



Três, Notas de Aula, Copyright © 2004 Pearson Education

Normalizações comuns

- Remoção de média e divisão pelo desvio padrão
- Padronização para um intervalo [0,1]

X	X - μ	$\frac{(X - \mu)}{\sigma}$
3	-4	-1.069
6	-1	-0.26
12	5	1.33

$\mu = \frac{(3+6+12)}{3} = 7$

$\sigma = \sqrt{\frac{1}{3}((3-7)^2 + (6-7)^2 + (12-7)^2)} = 3.74$

X	X - min(X)	$\frac{(X - \min(X))}{\max(X) - \min(X)}$
3	0	0
6	3	0.33
12	9	1

$\min(X) = \min(3,6,12) = 3$

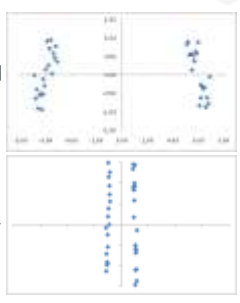
$\max(X) = \max(3,6,12) = 12$

$\max(X) - \min(X) = 9$

Slide de Fernando Beserra

Normalização é sempre apropriada?

➤ escore z (efeito semelhante para linear [0,1])



Normalização é sempre apropriada?

- Comparação das alternativas de padronização

$\frac{X - \mu}{\sigma}$	$\frac{(X - \min(X))}{\max(X) - \min(X)}$	$\frac{X - \bar{X}}{\text{Range}}$
-0.50	0	-0.428
-0.50	0.0003	0
-0.49	0.0009	0.857
1.99	1	1427.1
-0.50	0.0002	-0.142

Escalamento padrão

- Outlier influencia fortemente no posicionamento da média
- Menor poder de discriminação para dados normais (não-outliers)

Escalamento robusto

- Maior poder de discriminação
- Outlier continua destoando dos dados normais

Padronização para um intervalo [0,1]

- Outlier desloca os outros valores para perto de 0
- Perigo: preditores podem passar a ignorar dados normais

Slide de Fernando Beserra

Discussão

- Atributos com escala mais ampla (maior variabilidade) tendem a ter maior peso nos cálculos de distâncias;
- Isso representa uma forma de **pré-ponderação** dos dados;
- Normalização busca eliminar esse efeito, presumindo-o ser artificial;
- Simples consequência do uso de unidades de medida específicas;
- Porém, impõe uma (contra) ponderação aos dados originais;
- Introduz distorções se (ao menos parte das) diferentes variabilidades originais refletiam corretamente a natureza do problema;
- ❑ Agrupamento de dados é considerada uma área muito desafiadora.

Como lidar com atributos discretos?

	Sexo	País	Estado Civil	Comprar
x ₁	M	França	solteiro	Sim
x ₂	M	China	separado	Sim
x ₃	F	França	solteiro	Sim
x ₄	F	Inglaterra	casado	Sim
x ₅	F	França	solteiro	Não
x ₆	M	Alemanha	viúvo	Não
x ₇	M	Brasil	casado	Não
x ₈	F	Alemanha	casado	Não
x ₉	M	Inglaterra	solteiro	Não
x ₁₀	M	Argentina	casado	Não

Motivação:
 $d(\mathbf{x}_1, \mathbf{x}_6) = ?$
 $d(\mathbf{x}_1, \mathbf{x}_7) = ?$

Atributos binários

- Calcular a distância entre $\mathbf{x}_1 = [1\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0]$ e $\mathbf{x}_2 = [0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0]$
- Usando uma tabela de contingências temos:

Objeto x _i	Objeto x _j			
	1	0	Total	
	1	n ₁₁	n ₁₀	n ₁₁ +n ₁₀
0	n ₀₁	n ₀₀	n ₀₁ +n ₀₀	
Total	n ₁₁ +n ₀₁	n ₁₀ +n ₀₀	n	

$$S_{(\mathbf{x}_i, \mathbf{x}_j)}^{SM} = \frac{n_{11} + n_{00}}{n_{11} + n_{00} + n_{10} + n_{01}} = \frac{n_{11} + n_{00}}{n}$$

Coefficiente de Casamento Simples (Zubin, 1938)

$$1 - S_{(\mathbf{x}_i, \mathbf{x}_j)}^{SM} = \frac{n_{10} + n_{01}}{n} = \frac{d_{Hamming}(\mathbf{x}_i, \mathbf{x}_j)}{n}$$

Atributos assimétricos

- **Atributos simétricos:** valores igualmente importantes.
 - Exemplo típico → Sexo (M ou F)
- **Atributos assimétricos:** valores com importâncias distintas – presença de um efeito é mais importante do que sua ausência.
 - Exemplo: sejam 3 objetos que apresentam (1) ou não (0) dez sintomas para uma determinada doença:
 $\mathbf{x}_1 = [1\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1]$ $S^{SM}(\mathbf{x}_1, \mathbf{x}_2) = 0.5;$
 $\mathbf{x}_2 = [1\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0]$ $S^{SM}(\mathbf{x}_1, \mathbf{x}_3) = 0.5;$
 $\mathbf{x}_3 = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$ ➤ Conclusão?

Atributos assimétricos

➤ Para atributos assimétricos pode-se usar, por exemplo, o *Coefficiente de Jaccard* (1908):

$$S_{(x_i, x_j)}^{Jaccard} = \frac{n_{11}}{n_{11} + n_{10} + n_{01}}$$

- Focada nos *casamentos* do tipo 1-1.
- Despreza *casamentos* do tipo 0-0.
- Existem outras medidas similares na literatura, mas CCS e Jaccard são as mais utilizadas.
- Ver Kaufman & Rousseeuw (2005).

Atributos assimétricos

Exemplo:

$$\begin{aligned} \mathbf{p} &= [1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0] \\ \mathbf{q} &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0] \end{aligned}$$

$n_{01} = 2$ (número de atributos em que $p = 0$ e $q = 1$)
 $n_{10} = 1$ (número de atributos em que $p = 1$ e $q = 0$)
 $n_{00} = 7$ (número de atributos em que $p = 0$ e $q = 0$)
 $n_{11} = 0$ (número de atributos em que $p = 1$ e $q = 1$)

$$\begin{aligned} CCS &= (n_{11} + n_{00}) / (n_{01} + n_{10} + n_{11} + n_{00}) \\ &= (0 + 7) / (2 + 1 + 0 + 7) = 0.7 \end{aligned}$$

$$J = n_{11} / (n_{01} + n_{10} + n_{11}) = 0 / (2 + 1 + 0) = 0.0$$

Atributos ordinais

Ex.: Gravidade de um efeito: {nula, baixa, média, alta}.

- Ordem dos valores é importante.
- Normalizar e então utilizar medidas de (dis)similaridade para valores contínuos (e.g., euclidiana, cosseno etc.):
 - $\{1, 2, 3, 4\} \rightarrow (\text{rank} - 1) / (\text{número de valores} - 1)$:
 - $\{0, 1/3, 2/3, 1\}$
- Abordagem comum.

Atributos de várias naturezas misturados

Método de Gower (1971):

$$s_{(x_i, x_j)} = \frac{1}{n} \sum_{k=1}^n s_{ijk} \longrightarrow d_{(x_i, x_j)} = 1 - s_{(x_i, x_j)}$$

Para atributos nominais / binários:

$$\begin{cases} (x_{ik} = x_{jk}) \Rightarrow s_{ijk} = 1; \\ (x_{ik} \neq x_{jk}) \Rightarrow s_{ijk} = 0; \end{cases}$$

Para atributos ordinais ou contínuos:

$$s_{ijk} = 1 - |x_{ik} - x_{jk}| / R_k \quad R_k = \max_m x_{mk} - \min_m x_{mk}$$

R_k = faixa de observações do k -ésimo atributo (termo de normalização)

Sumário

- Diferentes medidas de dis(similaridade) afetam a formação (indução) dos *clusters*;
 - Como selecionar a medida de (dis)similaridade?
 - Devemos padronizar? Caso afirmativo, como?
- Infelizmente, não há respostas definitivas e globais.
- Análise de agrupamento de dados é, em essência, um processo subjetivo, dependente do problema.
- Lembrem: **análise exploratória de dados!**

Agenda



Preparação para métodos supervisionados

Além das técnicas mencionadas anteriormente é comum realizar seleção de atributos (**feature selection**):

- Subconjunto mínimo de atributos tal que a distribuição de probabilidades para diferentes classes seja parecida à distribuição original (com todos os atributos);
- Facilita interpretação dos modelos obtidos;
- Reduz custo computacional de armazenamento (sistemas produtivos) e de inferência.

- Guyon, I., Elisseeff, A., An Introduction to Variable and Feature Selection, Journal of Machine Learning Research, 2003;
 - Liu, H., Yu, L., Toward Integrating Feature Selection Algorithms for Classification and Clustering, IEEE Transactions on Knowledge and Data Engineering, 17(3), 1-12, 2005.

Complexidade e estratégias

- Otimização combinatória: existem $2^n - 1$ subconjuntos possíveis de " n " atributos;
 - Busca exaustiva é usualmente inviável;
 - Diversas estratégias de busca:
 - Seleção *forward*;
 - Eliminação *backward*;
 - Bidirecional (literatura em inteligência artificial é ampla);




Complexidade e estratégias

- Otimização de atributos;
 - Busca
 - Diversas
 - Seleção
 - Eliminação
 - Bidirecional
- Como parar a busca?
- Como avaliar os subconjuntos de atributos?



Complexidade e estratégias

- Otimização combinatória: existem $2^n - 1$ subconjuntos possíveis de " n " atributos;
 - Busca exaustiva é usualmente inviável;
 - Diversas estratégias de busca:
 - Seleção *forward*;
 - Eliminação *backward*;
 - Bidirecional (literatura em inteligência artificial é ampla);
 - Como parar a busca?
 - Como avaliar os subconjuntos de atributos?



Abordagens

- (i) **Incorporados** (*embedded*): a seleção de atributos é intrínseca ao próprio método (e.g. C4.5, 1R).
- (ii) **Filtragem** (*filters*): selecionar atributos de acordo com características dos dados que presumivelmente influenciam a eficácia do algoritmo de aprendizado. Independem do algoritmo de aprendizado a ser usado.
- (iii) **Empacotamento** (*wrappers*): subconjunto de atributos selecionados é avaliado por meio do próprio algoritmo de aprendizado.
 - Em geral fornecem melhores resultados do que a *filtragem*, mas são computacionalmente mais caros.
 - Atributos selecionados podem não ser apropriados para modelos diferentes daquele usado para avaliar os subconjuntos de atributos.
- (iv) **Métodos Híbridos** (*hybrid approaches*): procuram combinar as vantagens oferecidas pelos modelos (i)-(iii). Filtragem por correlação linear e modelagem não linear (e.g., redes neurais).

Exemplos de Filtros

- Usar o critério do ganho de informação (árvores);
- Agrupar atributos de acordo com medidas de correlação;
- Escore de Fisher (Duda & Hart, Pattern Classification and Scene Analysis, Wiley, 1973):
 - Considerando um problema formado por duas classes (+,-), para cada atributo $j=1,...,n$ calcular:
$$w_j = \frac{(\mu_j^+ - \mu_j^-)^2}{(\sigma_j^+)^2 + (\sigma_j^-)^2}$$
 - Presume-se que a qualidade de cada atributo (w_j) pode ser avaliada individualmente, sem levar em conta as interações entre atributos.

Exemplos de filtro

- Pode-se lidar com múltiplas classes de maneira análoga;
- Considerando cada classe i e atributo j temos:

$$\mu_{ji} = \frac{1}{|C_i|} \sum_{x \in C_i} x_j$$

- A média total para o atributo j é definida como:

$$\mu_j = \frac{1}{m} \sum_x x_j$$

- Usando as duas equações acima pode-se definir a dispersão entre classes para o atributo j como:

$$B_j = \sum_{i=1}^c |C_i| (\mu_{ji} - \mu_j)^2$$

- Em função de B_j podemos usar a seguinte função de escore:

$$B_{dispersão,j} = \frac{B_j}{\sum_{i=1}^c \sigma_{\mu_i}}$$

Chai et al., An Evaluation of Gene Selection Methods for Multi-class Microarray Data Classification, Proc. European Workshop on Data Mining and Text Mining in Bioinformatics, 2004.

Exemplo de wrapper – Naive Bayes

- Atributos irrelevantes e redundantes podem comprometer acurácia de classificação;
- Selecionar atributos com base no desempenho do classificador NB. Pode-se sumarizar o NBW como segue:

- 1) Construir um classificador NB para cada atributo $X_i (i = 1, \dots, n)$. Escolher X_i para o qual o NB apresenta a melhor acurácia e inseri-lo em $A_S = \{\text{atributos selecionados}\}$;
- 2) Para todo $X_i \notin A_S$ construir um NB formado por $\{X_i\} \cup A_S$. Escolher o melhor classificador dentre os disponíveis e verificar se é melhor do que o obtido anteriormente:
 - a) SE sim: atualizar A_S inserindo o atributo adicional, e repetir 2).
 - b) SENÃO: parar e usar o classificador obtido anteriormente.

Complexidade do wrapper

- NB possui complexidade de tempo linear com o número de exemplos e de atributos;
- Constante de tempo do NB também é baixa (computar frequências relativas e/ou densidades);
- Algoritmo NB é facilmente paralelizável;
- O que dizer sobre o NBW?
 - Teoria: $O(2^n)$, n = número de atributos;
 - Busca gulosa *pode* o espaço de busca do problema de otimização combinatoria: $O(n + (n-1) + \dots + 1) = O(n^2)$
 - Por exemplo, para $n=100$ temos: 1.2×10^{30} versus 10^4 avaliações de classificadores diferentes para escolher o melhor.
 - Facilmente paralelizável.

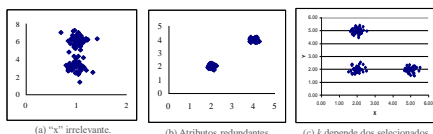
Comparando técnicas

- NÃO se deve selecionar atributos no conjunto completo de dados disponíveis e então rodar a validação cruzada apenas com os atributos selecionados (e.g., via filtros);
- Queremos estimar a capacidade de generalização do modelo: validação cruzada;
- Separar dados em conjuntos de *treinamento* e *teste*;
- Executar validação cruzada no conjunto de *treinamento* (treino+validação) para selecionar atributos e construir modelo;
 - A partir do classificador construído no conjunto de treinamento, avaliá-lo no conjunto de teste;
- Classificador que vai pra produção: construir com todos os dados disponíveis e parâmetros aprendidos na validação cruzada.

Reunanen, Overfitting in Making Comparisons Between Variable Selection Methods, Journal of Machine Learning Research 3, 1371-1382, 2003.

Back to clustering: como selecionar atributos?

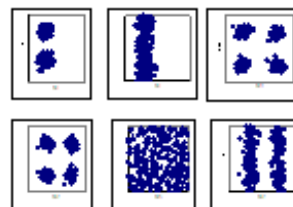
- Informação da classe não está disponível;
- Números de *clusters* e de atributos estão intimamente relacionados;
- Problema se torna muito difícil quando k é desconhecido a priori;
- Vejamos alguns exemplos:



→ O que pode acontecer em bases com mais do que dois atributos?

Back to clustering: como selecionar atributos?

Consideremos 6 atributos (X_1, X_2, \dots, X_6):



Quantos *clusters naturais* existem nesta base de dados?

Possíveis abordagens

- Filtros;
- Métodos baseados em empacotamento:
 - Difícil estabelecer critérios de validade.
 - Difícil comparar partições formadas por diferentes quantidades de grupos e de atributos selecionados.
 - Viável quando se dispõe de partição de referência.
- Métodos híbridos (empacotar + filtrar):
 - Por exemplo, combinar k-means com NBW.
- Problema muito difícil e pouco estudado.

Agenda



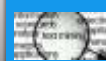
Pré-processamento



Técnicas Não-supervisionadas



Técnicas supervisionadas



Preparação de Dados Textuais

Classificação de textos

Y(

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet.

)=C

➤ Aplicações relacionadas em nosso contexto?

Slide por Dan Jurafsky

Classificação de textos

Y(

I love this movie! It's **sweet**, but with **satirical** humor. The dialogue is **great** and the adventure scenes are **fun**... It manages to be **whimsical** and **romantic** while **laughing** at the conventions of the fairy tale genre. I would **recommend** it to just about anyone. I've seen it **several** times, and I'm always **happy** to see it **again** whenever I have a friend who hasn't seen it yet.

)=C

Slide por Dan Jurafsky

Classificação de textos

Y(

x love xxxxxxxxxxxxxxxx sweet
xxxxxxx satirical xxxxxxxxxxxx
xxxxxxxxxxx great xxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxx fun xxx
xxxxxxxxxxxxxxxxxxx whimsical xxx
romantic xxxx laughing
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxx recommend xxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xx several xxxxxxxxxxxxxxxxxxxx
xxxxx happy xxxxxxxx again
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

)=C

Slide por Dan Jurafsky

Naive Bayes para classificação de textos

• Considerando que cada classe é representada por C_k e que cada documento é representado por um vetor de palavras (\mathbf{d}) queremos computar:

$$P(C_k | \mathbf{d}) = \frac{P(\mathbf{d} | C_k)P(C_k)}{P(\mathbf{d})}$$

• Vejamos como computar cada um dos termos relevantes (numerador).

91

Naive Bayes para classificação de textos

- Para estimar cada termo, computamos frequências relativas:

$$P(c_j) = \frac{doc_count(C=c_j)}{N_{doc}}$$

$$P(w_i | c_j) = \frac{count(w_i, c_j)}{\sum_{w'} count(w', c_j)}$$

- w_i representa cada palavra de um vocabulário V ;
- $P(w_i | c_j)$ é a frequência relativa com a qual w_i aparece em relação a todas as palavras dos documentos do tópico c_j ;
- Usar estimadores de Laplace para evitar termos nulos.

Naive Bayes para classificação de textos

Mais especificamente, inicialmente extraímos um vocabulário do corpus;

- Para calcular $P(c_j)$ fazemos, para cada c_j em C :

$documents_j \leftarrow$ todos os documentos cuja classe é c_j

$$P(c_j) \leftarrow \frac{|documents_j|}{|total\ \#\ documents|}$$

- Calcular $P(w_k | c_j)$:

$Text_{c_j} \leftarrow$ único documento contendo todos os $documents_j$

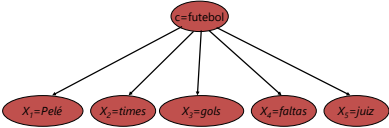
Para cada palavra w_k do vocabulário:

$n_k \leftarrow$ # de ocorrências de w_k no $Text_{c_j}$

$$P(w_k | c_j) \leftarrow \frac{n_k + \alpha}{n + \alpha |Vocabulary|}$$

Naive Bayes para classificação de textos

Representando o classificador na forma de um modelo gráfico para a classe/tópico futebol:



Naive Bayes para classificação de textos

- Note que Naive Bayes fornece uma alternativa para se modelar (aproximadamente) a linguagem natural;
- Considere que queiramos modelar sentenças:

$$P(s|C) = \prod P(palavra|C)$$

Classe +

0.1	I	<u>I</u>	<u>love</u>	<u>this</u>	<u>fun</u>	<u>film</u>
0.1	love	0.1	0.1	.05	0.01	0.1
0.01	this					
0.05	fun					
0.1	film					
...	...					

$$P(s|+) = 0.0000005$$

Slide por Dan Jurafsky

Naive Bayes para classificação de textos

- Qual classe atribui a maior probabilidade à sentença?

Model +		Model -	
0.1	I	0.2	I
0.1	love	0.001	love
0.01	this	0.01	this
0.05	fun	0.005	fun
0.1	film	0.1	film

	I	love	this	fun	film
	0.1	0.1	0.01	0.05	0.1
	0.2	0.001	0.01	0.005	0.1

$P(s|+) > P(s|-)$

Slide por Dan Jurafsky

Detecção de spam

Atributos derivados de observação:

- Mentions Generic Viagra
- Online Pharmacy
- Mentions millions of (dollar) ((dollar) NN,NNN,NNN.NN)
- Phrase: impress ... girl
- From: starts with many numbers
- Subject is all capitals
- One hundred percent guaranteed
- Claims you can be removed from the list
- http://spamassassin.apache.org/tests_3_3_x.html

➤ Interessante, produz bons resultados mas como (eficientemente) manter base de regras atualizada?

Slide por Dan Jurafsky

Tocando em frente

- Aprofundar conhecimento em técnicas específicas
- Aprendizado semi-supervisionado
- Aprendizado ativo
- Aprendizado por reforço
- Fluxos de dados
- Redes complexas
- Modelos gráficos probabilísticos

- Deep learning
- Algoritmos evolutivos
- Sistemas de recomendação
- Processamento paralelo e distribuído
- Bancos de dados (teoria + SQL)
- Softwares: Hadoop, Spark, Hive, R, Python etc.

Slide por Dan Jurafsky





Obrigado
Marco Antonio Lopes
marco-antonio.lopes@itau-unibanco.com.br