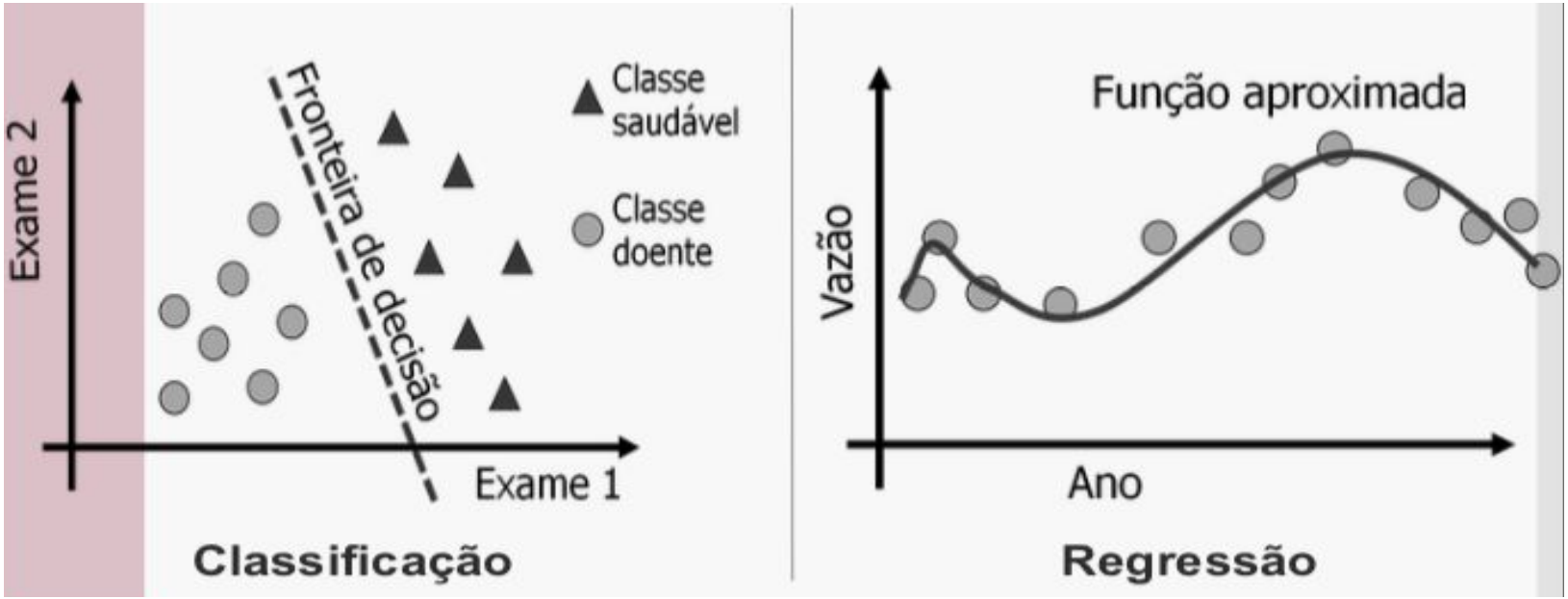
Aprendizado de Máquina

Classificação – Parte I

Modelos Preditivos

- Algoritmo de AM preditivo: função que, dado um conjunto de exemplos rotulados, constrói um estimador
- Classificação
  - Rótulos nominais (conjunto discreto e não ordenado)
  - Ex: (doente, saudável), (fraude, não fraude), (carro, moto, pedestre)
  - Estimador = classificador
- Regressão
  - Rótulos contínuos (conjunto infinito ordenado de valores)
  - Ex: peso, altura, temperatura, valor
  - Estimador = regressor

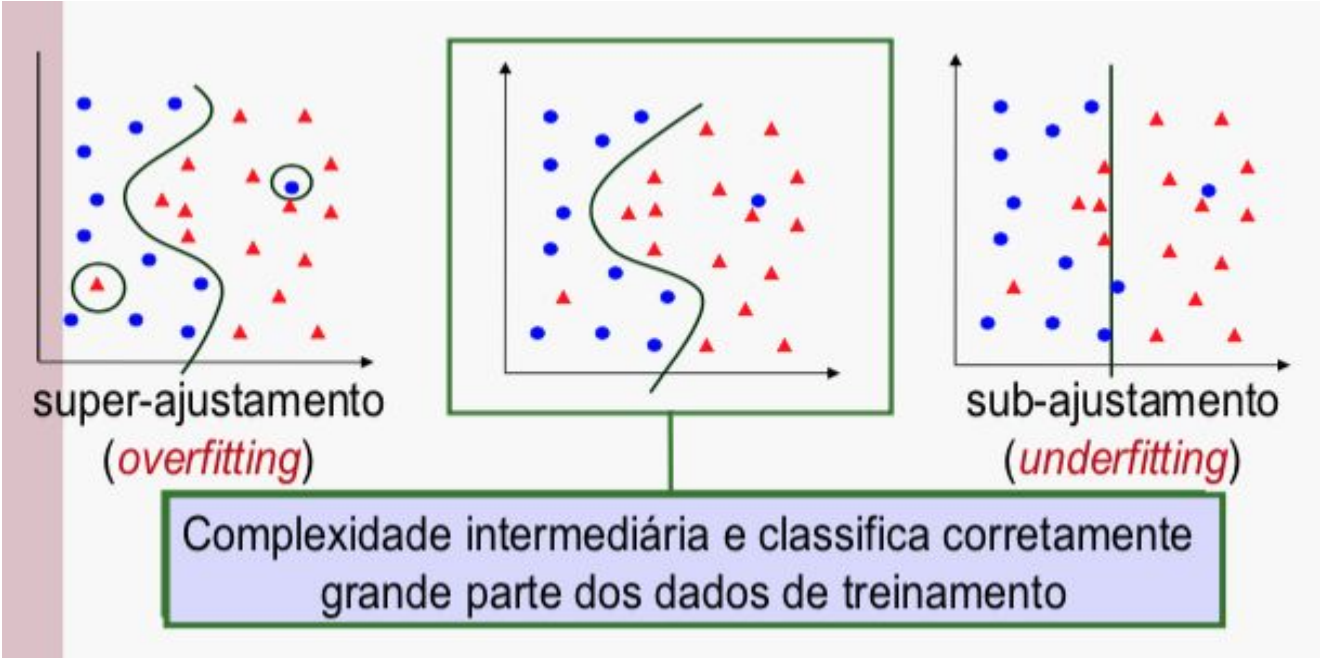

Exame 2
Fronteira de decisão
Classe saudável
Classe doente
Exame 1
Classificação
Função aproximada
Vazão
Ano
Regressão

Classificação

- Meta: encontrar fronteira de decisão que separe classes
- Diferentes algoritmos de AM podem encontrar diferentes fronteiras
- Mesmo algoritmo pode também encontrar fronteiras diferentes
  - Diferenças nos dados de treinamento
  - Variações na ordem de apresentação dos exemplos
  - Processos estocásticos internos

Classificação - Exemplos

- Diagnóstico de doenças
- Distribuição de espécies
- Categorização de textos
- Classificação de imagens
- Detecção de fraudes
- Filtro de SPAM
- Liberação de crédito
- Identificação de perfis de personalidade


super-ajustamento
(overfitting)
sub-ajustamento
(underfitting)
Complexidade intermediária e classifica corretamente
grande parte dos dados de treinamento

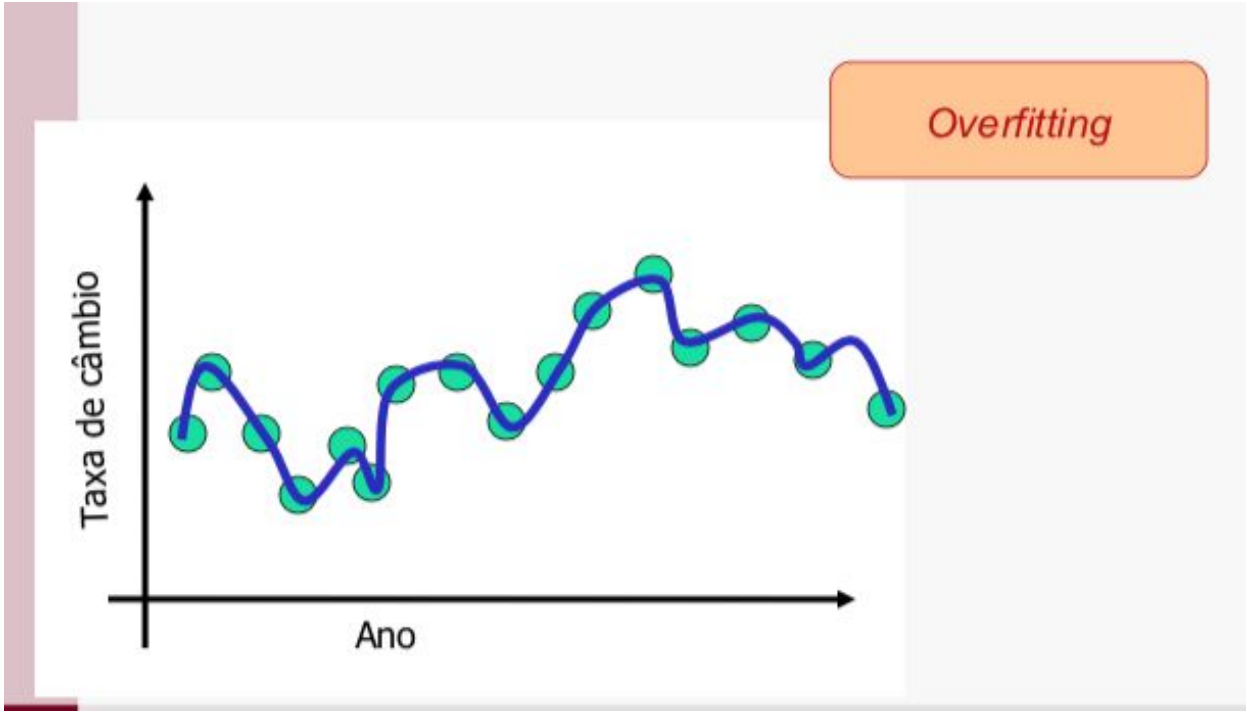
Classificação

- Técnicas de AM:
- Árvores de Decisão (C4.5)
- Conjuntos de regras
- Redes Neurais Artificiais
- Máquinas de Vetores de Suporte
- K-vizinhos mais próximos
- Regressão Logística
- Redes Bayesianas.
- Deep Learning - várias

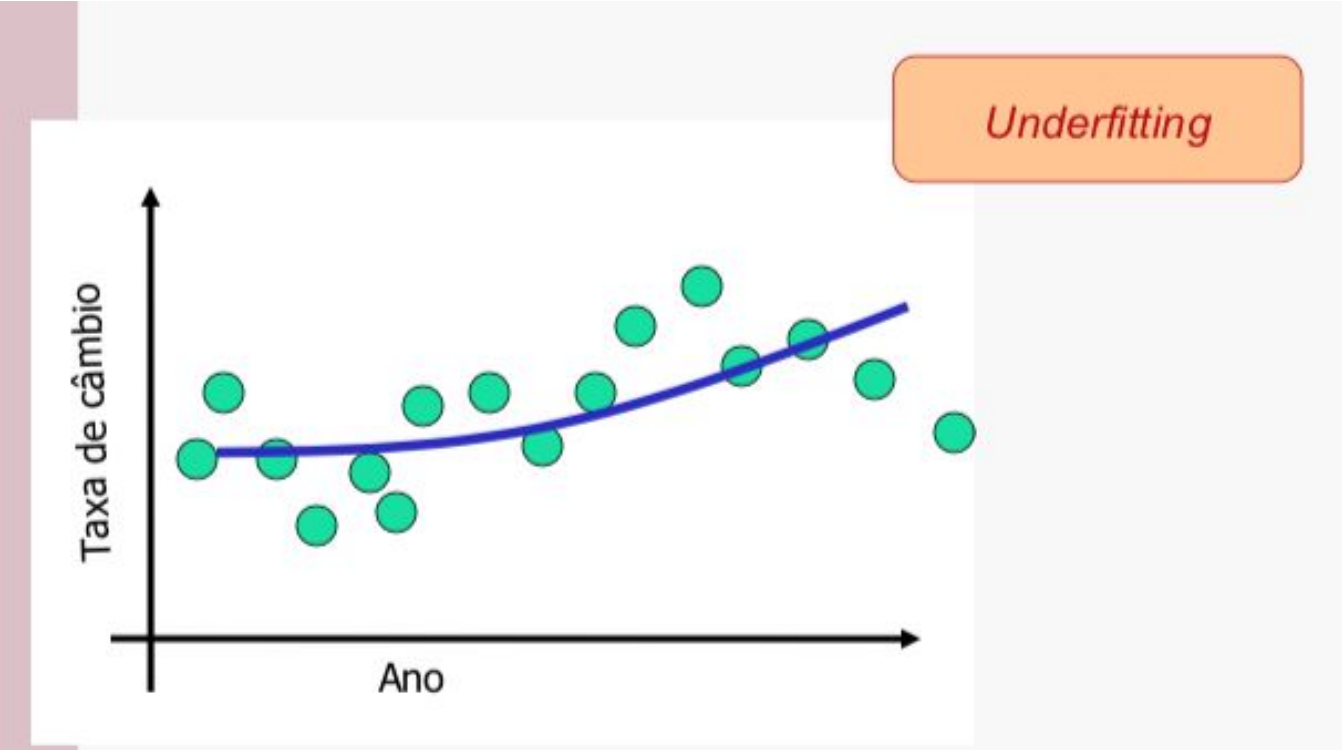
Regressão

- Meta: aprender função (curva aproximada) que relacione entradas a valores contínuos de saídas
- Também há diferentes algoritmos de AM para definir essas curvas
- Exemplos:
  - Prever valor de mercado de um imóvel
  - Prever o lucro de um empréstimo bancário
  - Prever criminalidade em uma região

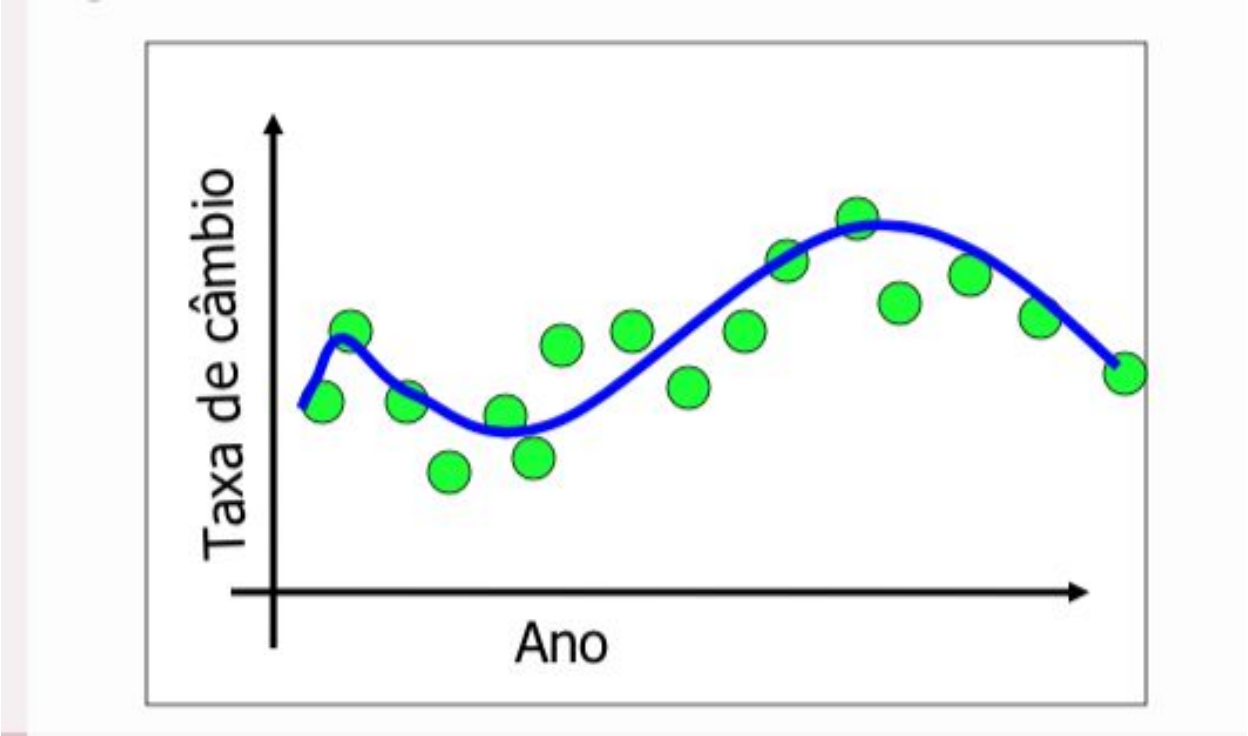
Regressão



Regressão



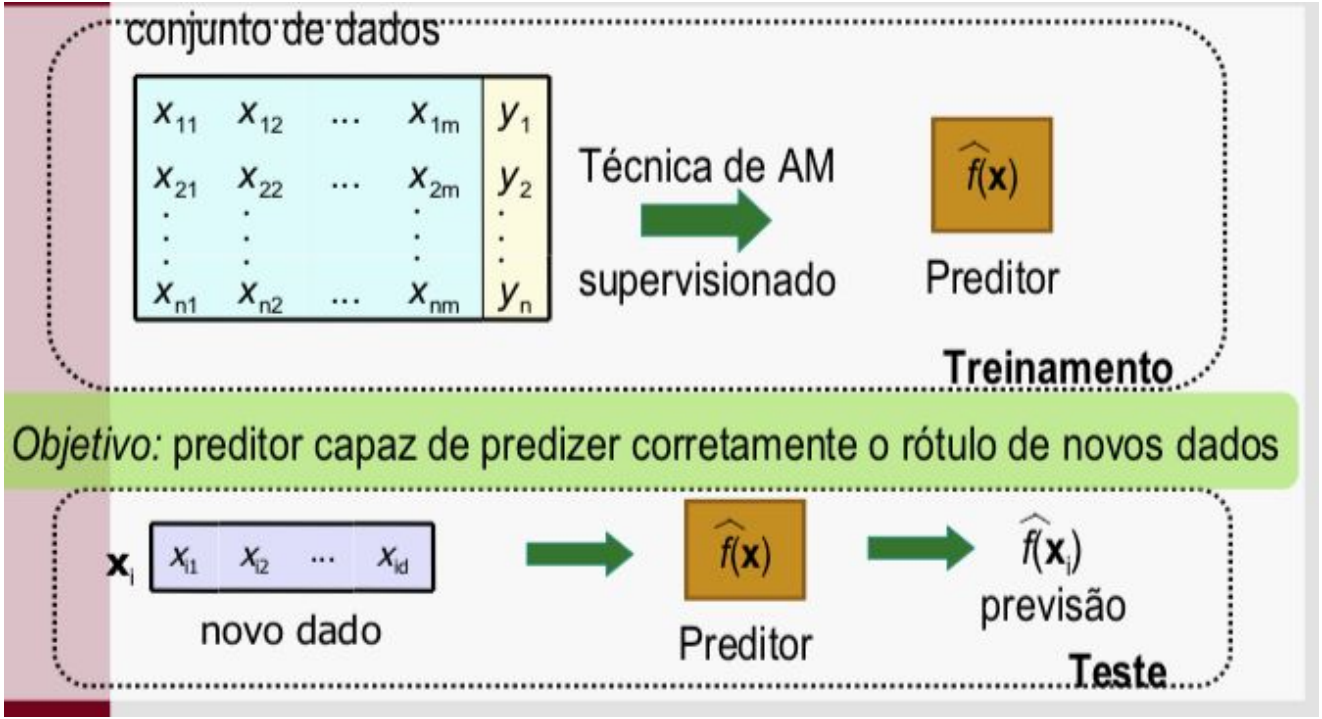
Regressão



Regressão

- Técnicas de AM:
  - Árvores de Regressão
  - Redes Neurais Artificiais
  - Máquinas de Vetores de Suporte
  - Regressão Linear
  - Etc.

Modelos Preditivos



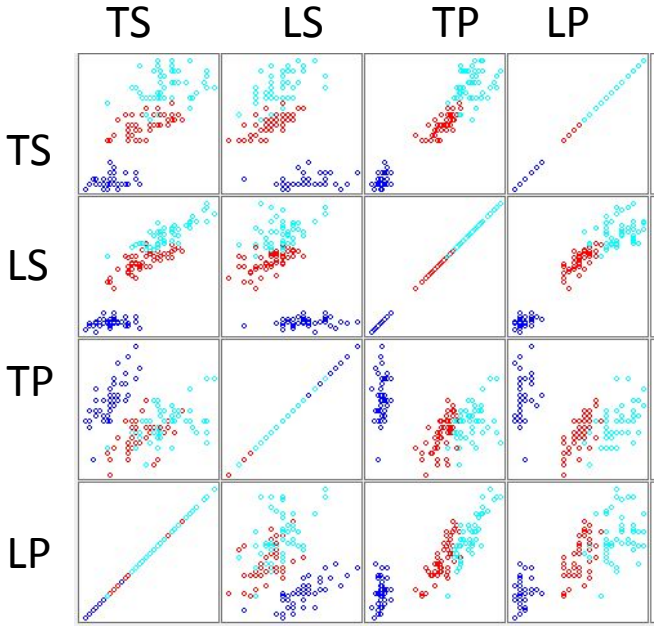
Métodos baseados em distâncias

- Técnicas de AM que consideram proximidade entre os dados para realizar predições

Hipótese: dados similares tendem a estar concentrados em uma mesma região do espaço de entradas

E dados que não são similares estarão distantes entre si

Métodos baseados em distâncias



Proximidade

- Medida de proximidade entre pares de objetos pode ser de:

Similaridade

- Mede o quanto dois objetos são parecidos
- Quando mais parecidos ⇒ maior o valor
- Geralmente valor ∈ [0, 1]

Dissimilaridade

- Mede o quanto dois objetos são diferentes
- Quanto mais diferentes ⇒ maior o valor
- Geralmente valor ∈ [0, X]

Escolha da medida deve considerar tipos e escalas dos atributos, além de propriedades dos dados que se deseja focar

Similaridade vs Dissimilaridade

Similaridade e dissimilaridade

- Normalmente as medidas satisfazem algumas propriedades, tais como:
  - Os objetos não são diferentes de si próprios
    - $d(x_i, x_i) = 0$
    - Em similaridade, objetos são similares a si próprios
      - $s(x_i, x_i) = 1$
  - Simetria
    - $d(x_i, x_j) = d(x_j, x_i)$
  - Positividade
    - $d(x_i, x_j) \geq 0$

Todas medidas de distância (medem dissimilaridade) satisfazem essas propriedades

Medidas de similaridade costumam ter definições menos rigorosas em relação às propriedades que devem satisfazer

Similaridade e dissimilaridade

- Outras propriedades possíveis:
  - $d(x_i, x_j) = 0 \; se \; e \; somente \; se \; x_i = x_j$
- Desigualdade triangular
  - $d(x_i, x_l) \leq d(x_i, x_j) + d(x_j, x_l)$

Medidas de distância que satisfazem essas propriedades também são denominadas métricas

Medidas para atributos quantitativos: dissimilaridade

- Para atributos racionais, mais usadas são distâncias baseadas na métrica de *Minkowski*
  - Distância *Lp, 1 ≤ p < ∞*

$$d(X,Y) = \sqrt[p]{\sum_{i=1}^{d} |x_i - y_i|^p}$$

- Menores valores de p ⇒ estimativas mais robustas
  - Menos sensíveis a outliers
- São sensíveis a variações de escala dos atributos
  - Normalmente solucionado por normalização

Distância Mahattan

- Distância de Manhattan:
  - *Minkowski com p = 1*
  - Também chamada distância bloco-cidade
    - Equivalente a Hamming para atributos binários

$$d(X, Y) = \sum_{i=1}^{d} |x_i - y_i|$$

Distância Euclidiana

- Distância Euclidiana:
  - Minkowski com p = 2
  - Medida de distância mais popular

$$d(X,Y) = \sqrt[2]{\sum_{i=1}^{d}(x_i - y_i)^2}$$

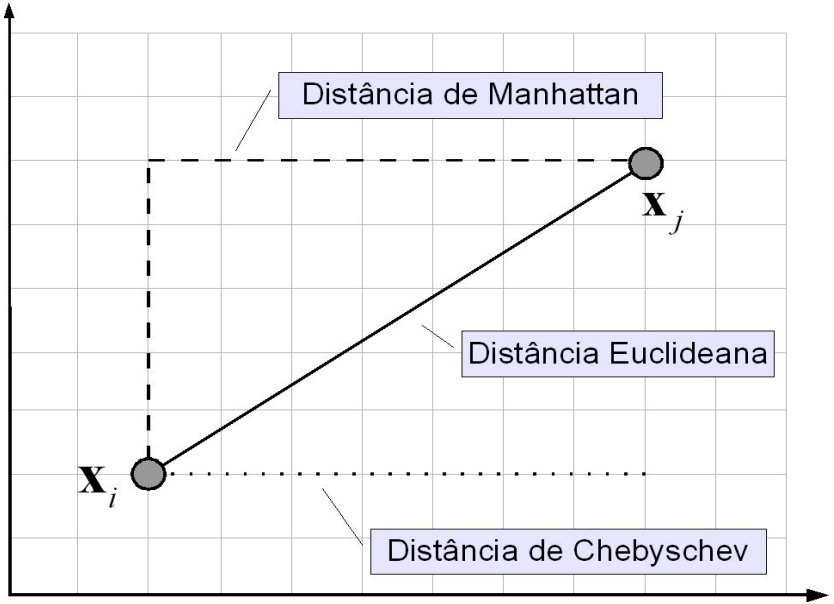
Distância Supremum

- Distância Supremum:
  - Minkowski com p = ∞
  - Também chamada distância de Chebyschev
    - Diferença absoluta máxima entre quaisquer atributos

$$d(x_i, x_j) = \max |x_i - y_i|$$

Métricas de Minskowski

- Interpretação das métricas de Minkowski



Medidas para atributos quantitativos: similaridade

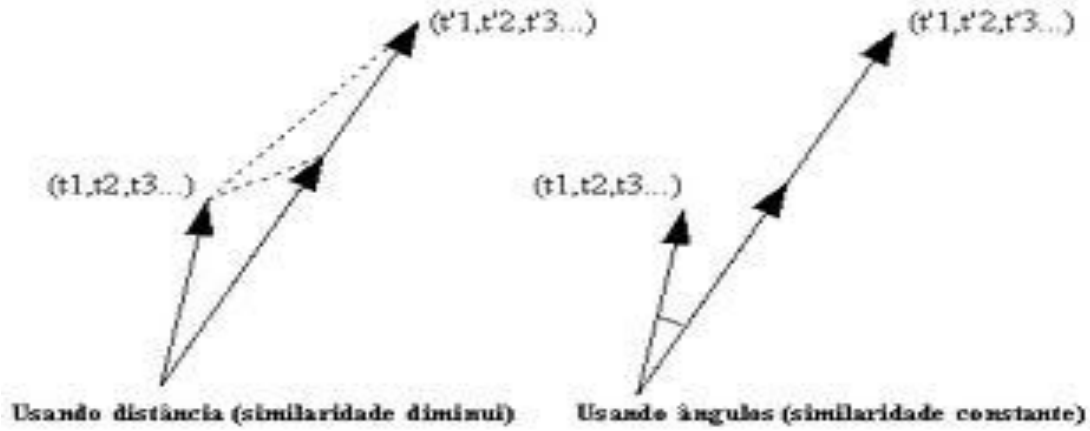
- Duas medidas comuns para avaliar similaridade:
  - Separação angular (cosseno)
    - Muito usada em Mineração de textos
      - Atributos assimétricos
        - Grande dimensionalidade
        - Esparsividade
  - Correlação de Pearson

Variam no intervalo [-1, 1], em que magnitude indica força da correlação e sinal indica a direção
Valores próximos de -1 e +1 indicam similaridade (correlação)
+1 ⇒ correlação positiva; -1 ⇒ correlação negativa

Cosseno

- Equação:

$$\cos(x_i, x_j) = \frac{\sum_{i=1}^{d} x_i y_i}{\sqrt{\sum (x_i)^2 \sum (y_i)^2}}$$



Verifica se dois vetores estão na mesma direção

Correlação de Pearson

- Equação:

$$pearson(x_i, x_j) = \frac{covariância(x_i, x_j)}{variância(x_i)\ variância(x_j)}$$

- É considerada uma medida de forma
  - Insensível a diferenças na magnitude dos atributos
  - Avaliar quando apenas o padrão de variação dos valores dos atributos dos objetos é importante

Medidas de similaridade

- Interpretação geométrica:
  - Correlação = 1 ⇒ vetores dos objetos são paralelos e apontam no mesmo sentido (ângulo 0°)
  - Correlação = -1 ⇒ vetores dos objetos são paralelos e apontam em sentidos opostos (ângulo 180°)
  - Correlação = 0 ⇒ vetores dos objetos são ortogonais (ângulo 90°)

Normalmente se usa o valor absoluto deles como similaridade

- Correlação de Pearson é sensível a outliers

Similaridade entre vetores binários

- Frequentemente, objetos têm apenas atributos com valores binários

- Similaridades podem ser computadas usando:
  - M01 = número de atributos em que x = 0 e y = 1
  - M10 = número de atributos em que x = 1 e y = 0
  - M00 = número de atributos em que x = 0 e y = 0
  - M11 = número de atributos em que x = 1 e y = 1

Similaridade entre vetores binários

- Coeficiente de Casamento Simples
  - CCS = num. de coinc. / num. de atributos
  - CCS = (M11 + M00) / (M01 + M10 + M11 + M00)

- Coeficiente Jaccard
  - J = num. coinc. 11 / num. em que ambos ≠ 0
  - J = (M11) / (M01 + M10 + M11)

Exemplo

- Calcular disssimilaridade entre p e q usando coeficientes:
  - Casamento Simples
  - Jaccard

p	= 1 0 0 1 1 0 1 0 1 1 1 0
q	= 0 1 0 0 1 1 0 0 1 0 1 1

Exemplo

- Calcular disssimilaridade entre p e q usando coeficientes:
  - Casamento Simples
  - Jaccard

p	= 1 0 0 1 1 0 1 0 1 1 1 0
q	= 0 1 0 0 1 1 0 0 1 0 1 1

- CCS = (M11 + M00) / (M01 + M10 + M11 + M00)
- J = (M11) / (M01 + M10 + M11)

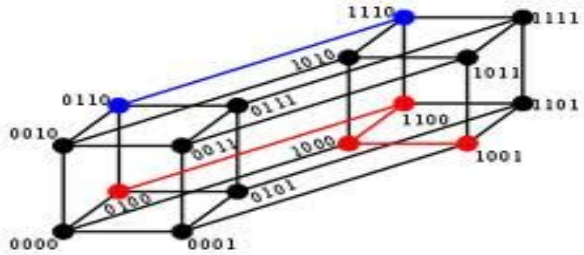
$M_{00} = 2$
$M_{01} = 3$
$M_{10} = 4$
$M_{11} = 3$

CCS = 5 / 12 = 0,41
J = 3 / 9 = 0,33

L_1 = 7 (distância)

Medidas para atributos qualitativos

- Medidas obtidas pela soma das contribuições individuais
- Ex. Distância de Hamming
  - Conta número de atributos categóricos com valores diferentes nos dois objetos
  - Varia em [0, d]
    - Valor 0 significa maior similaridade



Medidas para atributos heterogêneos

- Muitos conjuntos de dados apresentam atributos qualitativos e quantitativos
- Coeficiente geral de similaridade
  - s_{ij} = contribuição do *k*-ésimo atributo para a similaridade
  - w_{ijk} = 0 ou 1 (se comparação para o atributo é válida ou não)

$$s(x_i, x_j) = \frac{\sum_{k=1}^{d} w_{ijk} s_{ij}}{\sum w_{ijk}}$$

Algoritmos

Algoritmo dos vizinhos mais próximos

- Algoritmo de AM mais simples
  - Intuição: Objetos relacionados ao mesmo conceito são semelhantes entre si

Se anda como um cachorro e late como um cachorro, então provavelmente é um ...

Algoritmo dos vizinhos mais próximos

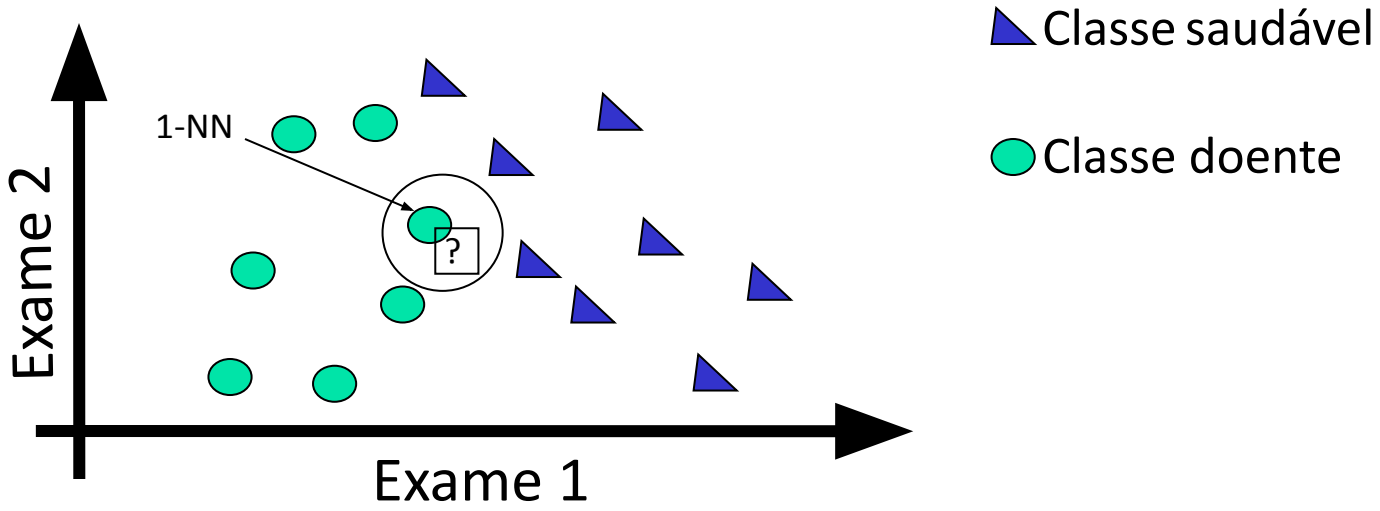
- Rotula novos objetos com base nos exemplos do conjunto de treinamento mais próximos a ele
  - É um algoritmo preguiçoso (lazy)
    - Não aprende modelo compacto, memoriza objetos de treinamento
    - Adia computação para a fase de classificação
  - Baseado em informações locais
  - Pode ser utilizado em classificação e regressão, sem necessidades de alterações significativas
  - Há variações de acordo com o número de vizinhos mais próximos adotado

Algoritmo 1-vizinho mais próximo

- Variação mais simples: 1-NN
  - 1-Nearest Neighbour
  - Cada objeto representa um ponto no espaço de entradas
  - Definindo métrica, é possível calcular distâncias
  - Métrica mais usual: distância euclidiana
  - Treinamento: memoriza exemplos rotulados do conjunto de treinamento
  - Classificação de novo exemplo: classe do exemplo de treinamento mais próximo

Algoritmo 1-vizinho mais próximo

- Ex. 1-NN



Algoritmo k-vizinhos mais próximos

- Extensão imediata do 1-NN considerando mais vizinhos
  - k vizinhos mais próximos
    - k é parâmetro do algoritmo
  - Cada vizinho vota em uma classe
    - Previsões são então agregadas

Classificação

$f(x_t) \leftarrow \text{moda}(f(x_1), ..., f(x_k))$

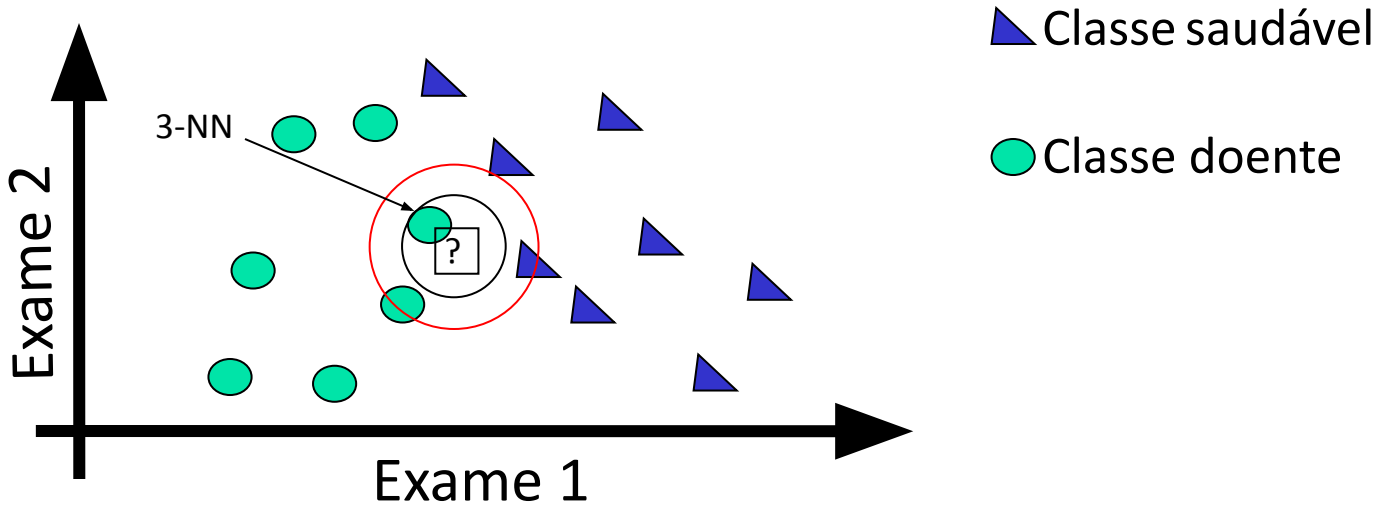
Regressão

$f(x_t) \leftarrow \text{média}(f(x_1), ..., f(x_k))$

ou $f(x_t) \leftarrow \text{mediana}(f(x_1), ..., f(x_k))$

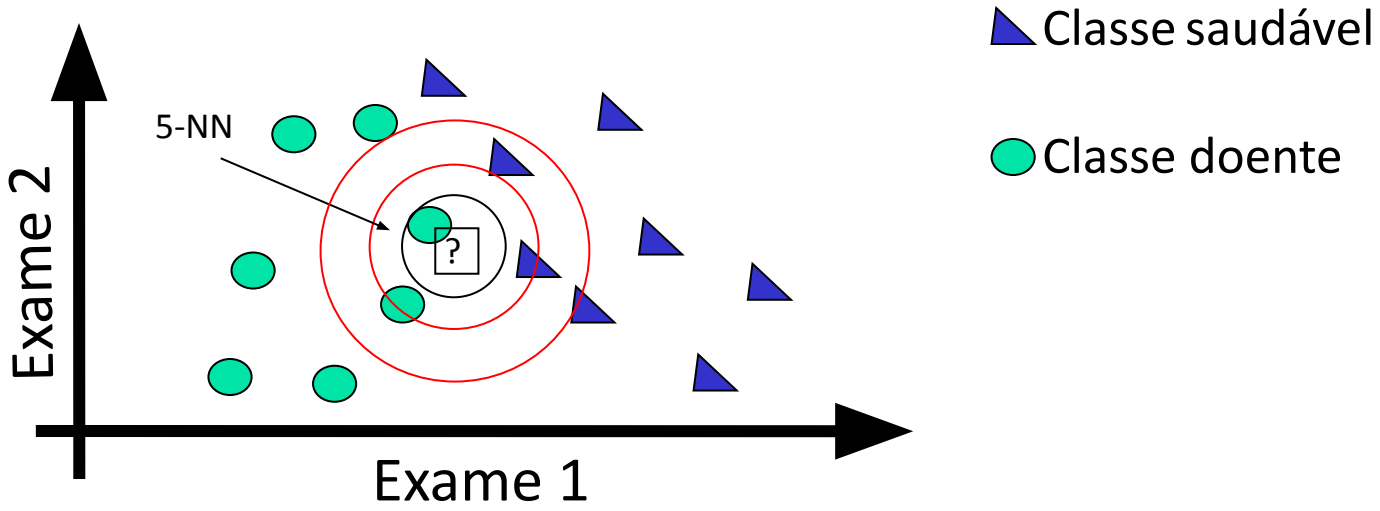
Algoritmo k-vizinho mais próximo

- Ex. 3-NN



Algoritmo k-vizinho mais próximo

- Ex. 3-NN



Quantos vizinhos?

- k muito grande
  - Vizinhos podem ser muito diferentes
  - Predição tendenciosa para classe majoritária
  - Custo computacional mais elevado
- k muito pequeno
  - Não usar informação suficiente
  - Previsão pode ser instável

Frequentemente usa k pequeno e ímpar (3, 5, ...). Valores pares não são usuais em classificação por poderem levar a empates

Exemplo

Nome	Febre	Enjôo	Manchas	Dores	Diagnóstico
João	sim	sim	pequenas	sim	doente
Pedro	não	não	grandes	não	saudável
Maria	sim	sim	pequenas	não	saudável
José	sim	não	grandes	sim	doente
Ana	sim	não	pequenas	sim	saudável
Leila	não	não	grandes	sim	saudável

Exemplo

- Usar k-NN e os exemplos anteriores para definir as classes dos exemplos de teste
  - Usar k = 1, 3 e 5
- Exemplos de teste:
  - (Luis, não, não, pequenas, sim)
  - (Laura, sim, sim, grandes, sim)

Atributo contendo nome não é usado

Exemplo

- Exemplos de teste:
  - (Luis, não, não, pequenas, sim)
    - d(x, x1) = 1 + 1 + 0 + 1 = 3
    - d(x, x2) = 0 + 0 + 1 + 1 = 2
    - d(x, x3) = 1 + 1 + 0 + 1 = 3
    - d(x, x4) = 1 + 0 + 1 + 0 = 2
    - d(x, x5) = 1 + 0 + 0 + 0 = 1
    - d(x, x6) = 0 + 0 + 1 + 0 = 1

Distância de Hamming

k = 1: saudável
k = 3: saudável
k = 5: saudável

Exemplo

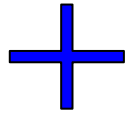
- Exemplos de teste:
  - (Laura, sim, sim, grandes, sim)
    - d(x, x1) = 0 + 0 + 1 + 0 = 1
    - d(x, x2) = 1 + 1 + 0 + 1 = 3
    - d(x, x3) = 0 + 0 + 1 + 1 = 2
    - d(x, x4) = 0 + 1 + 0 + 0 = 1
    - d(x, x5) = 0 + 1 + 1 + 0 = 2
    - d(x, x6) = 1 + 1 + 0 + 0 = 2

Distância de Hamming

k = 1: doente
k = 3: doente
k = 5: saudável

Análise do algoritmo

- Vantagens:
  - O algoritmo de treinamento é simples
    - Armazenar os objetos
  - Constrói aproximações locais da função objetivo
    - Interessante se função objetivo é muito complexa, mas pode ser descrita por aproximações locais de menor complexidade
  - É aplicável mesmo em problemas complexos
  - É um algoritmo naturalmente incremental
    - Novos exemplos ⇒ basta armazená-los na memória



Análise do algoritmo

- Desvantagens:
  - Não obtêm uma representação compacta dos dados
    - Não se tem modelo explícito a partir dos dados
  - Predição pode ser custosa
    - Requer calcular distâncias a todos os objetos de treinamento
  - É afetado pela presença de atributos redundantes e irrelevantes
  - Problemas com dimensionalidade elevada
    - Objetos ficam equidistantes
    - Maldição da dimensionalidade

Avaliação Modelos Preditivos

Avaliação Modelos Preditivos

- Não existe técnica de AM universal, que se saia melhor em qualquer tipo de problema
  - Implica na necessidade de experimentos

- Características do problema e das técnicas pode auxiliar em alguns casos
  - Ex. modelo deve ser interpretável ⇒ técnicas simbólicas, dados possuem alta dimensão ⇒RNA, etc.
  - Mesmo assim diversos algoritmos podem ser candidatos

Avaliação Modelos Preditivos

- Mesmo que um único algoritmo seja escolhido
  - Variações de parâmetros produzem diferentes modelos

- ⇒ Domínio de AM: necessidade de experimentação
  - Experimentos controlados
  - Procedimentos que garantem a corretude e reprodutibilidade dos experimentos

Avaliação Modelos Preditivos

- Diferentes aspectos podem ser considerados:
  - Acurácia do modelo nas previsões
  - Compreensibilidade do conhecimento extraído
  - Tempo de aprendizado
  - Requisitos de armazenamento
  - Etc.

Métricas de Erro

- Desempenho na rotulação de objetos
  - Métricas para classificação:
    - Taxa de erro
    - Acurácia
  - Métricas para regressão:
    - Erro quadrático médio
    - Distância absoluta média

Concentraremos discussões a medidas de desempenho preditivo

Métricas para classificação

- Taxa de erro de um classificador f
  - De classificações incorretas

$$err(f) = (1/n) \sum_{i=1...n} I(yi \neq f(xi))$$

  - Proporção de exemplos classificados incorretamente em um conjunto com n objetos
    - Comparação da classe conhecida com a predita
    - I é função identidade

= 1 se argumento é verdadeiro e 0 em caso contrário

    - Varia entre 0 e 1 e valores próximos de 0 são melhores

Métricas para classificação

- Taxa de acerto ou acurácia de um classificador f
  - Complemento da taxa de erro

$$acc(f) = 1 - err(f) = (1/n) \sum_{i=1...n} I(y_i = f(x_i))$$

  - Proporção de exemplos classificados corretamente em um conjunto com n objetos
    - Varia entre 0 e 1 e valores próximos de 1 são melhores

Métricas para classificação

- Matriz de confusão
  - Alternativa para visualizar desempenho de classificador
  - Predições corretas e incorretas em cada classe

Predito

Real		C_1	C_2	C_3
	C_1	11	1	3
	C_2	1	4	0
	C_3	2	1	6

- Linhas representam classes verdadeiras
- Colunas representam classes preditas
- Elemento m_{ij}: número de exemplos da classe c_i classificados como pertencentes à classe c_j
- Diagonal da matriz: acertos do classificador
- Outros elementos: erros cometidos

Amostragem

- Tem-se usualmente um único conjunto de n objetos
  - Deve ser usado para induzir e avaliar o preditor
  - Desempenho no conjunto de treinamento é otimista
    - Todos algoritmos tentam de alguma forma melhorar seu desempenho no conjunto de treinamento na fase indutiva
    - Avaliar modelo no conjunto de treinamento é conhecido como resubstituição
      - Produz taxa de erro/acerto aparente

Amostragem

- Métodos de amostragem: obter estimativas de desempenho mais confiáveis
  - Definindo subconjuntos disjuntos de:

Treinamento
Dados empregados na indução e no ajuste do modelo

Qualquer ajuste de parâmetros deve ser feito nos dados de treinamento

Teste
Simulam a apresentação de novos exemplos ao preditor
(não vistos em sua indução)

Somente avaliar o modelo obtido

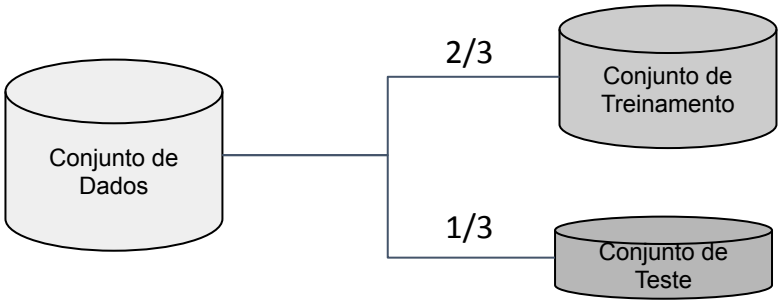
Em algumas situações, dados de treinamento são subdivididos, gerando conjunto de validação dedicado ao ajuste de parâmetros

Amostragem

- Principais métodos de amostragem:
  - Holdout
  - Amostragem aleatória
  - Validação cruzada
  - Leave-one-out
  - Bootstrap

Holdout

- Método mais simples:
  - Divide conjunto de dados em proporção p para treinamento e (1-p) para teste
    - Uma única partição
    - Valores típicos de p: ½, 2/3 ou ¾



Treinamento do modelo

Estimativa de desempenho

Holdout

Objeto	Atributo 1	Atributo 2	Atributo 3	Classe
1	855	5142	2708	Safra 95
2	854	23155	2716	Safra 95
3	885	16586	2670	Safra 95
4	877	16685	2677	Safra 95
5	839	5142	2708	Safra 95
6	854	5005	2685	Safra 95
7	885	19455	2708	Safra 95
8	839	5027	2708	Safra 95
9	877	16823	2677	Safra 95
10	892	19180	2716	Safra 95
11	24628	39437	381	Safra 96
12	43183	39277	328	Safra 96
13	27871	39712	389	Safra 96
14	42329	40307	328	Safra 96
15	41627	40032	335	Safra 96
16	39399	40322	335	Safra 96
17	33677	40375	328	Safra 96
18	33539	40078	335	Safra 96
19	34150	40353	358	Safra 96
20	34485	40742	358	Safra 96

Holdout

Conjunto de treinamento

Objeto	Atributo 1	Atributo 2	Atributo 3	Classe
4	877	16685	2677	Safra 95
6	854	5005	2685	Safra 95
8	839	5027	2708	Safra 95
2	854	23155	2716	Safra 95
10	892	19180	2716	Safra 95
1	855	5142	2708	Safra 95
6	854	5005	2685	Safra 95
18	33539	40078	335	Safra 96
15	41627	40032	335	Safra 96
19	34150	40353	358	Safra 96
12	43183	39277	328	Safra 96
17	33677	40375	328	Safra 96
20	34485	40742	358	Safra 96
11	24628	39437	381	Safra 96

Conjunto de teste

Objeto	Atributo 1	Atributo 2	Atributo 3	Classe
3	885	16586	2670	Safra 95
5	839	5142	2708	Safra 95
9	877	16823	2677	Safra 95
13	27871	39712	389	Safra 96
14	42329	40307	328	Safra 96
16	39399	40322	335	Safra 96

Holdout

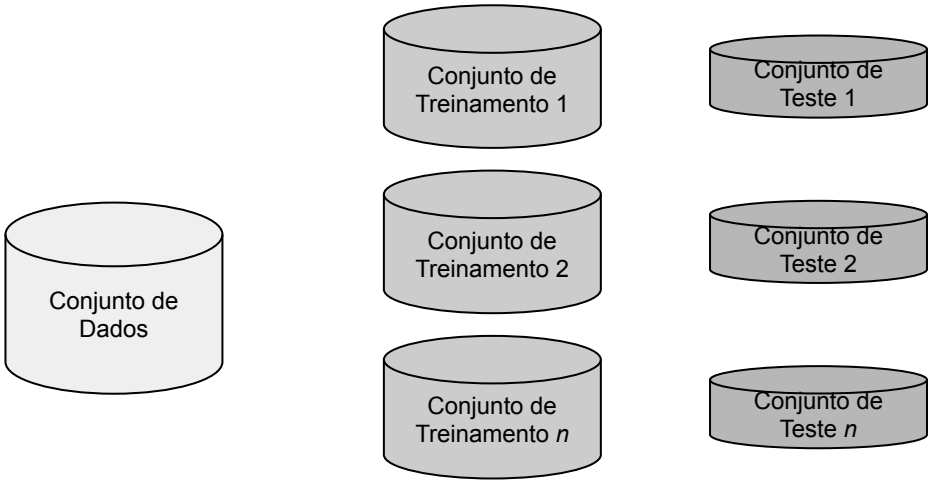
- Indicado para grande quantidade de dados
  - Se pequena quantidade de dados
    - Poucos exemplos são usados no treinamento
    - Modelo pode depender da composição dos conjuntos de treinamento e teste
      - Quanto menor conjunto de treinamento, maior a variância do modelo
      - Quanto menor conjunto de teste, menos confiável a acurácia estimada para ele
- Muito usado para definir subconjuntos de validação

Holdout

- Não avalia o quanto o desempenho de uma técnica varia
  - Quanto a diferentes combinações de exemplos de treinamento
  - É possível que uma divisão deixe no subconjunto de teste exemplos “mais fáceis”
  - Para tornar os resultados menos dependentes da partição feita:
    - vários holdout
    - Random subsampling (amostragem aleatória)

Amostragem aleatória

- Repetições de holdout
  - Há sobreposição entre os conjuntos de teste gerados
  - Fornece uma média de desempenho



Média e desvio-padrão de desempenho

Validação cruzada

- Método mais usado: r-fold cross validation
  - Conjunto é dividido em r partes de tamanho aproximadamente igual
  - Objetos de r-1 partes são usados no treinamento e a parte restante é usada para teste
  - Procedimento é repetido r vezes usando cada partição para teste
    - subconjuntos de teste são independentes entre si
  - Desempenho é dado por média
  - Valor típico de r: 10

Validação cruzada

- Variação: *r*-fold cross validation estratificado
  - Manter a distribuição de classes em cada partição
    - Ex: se conjunto de dados original tem 20% na classe c_1 e 80% na classe c_2, cada partição também deve manter essa proporção
  - Distribuição de classes: proporção de exemplos em cada classe
    - Para cada classe c_j, $dist(c_j)$ = número de exemplos que possuem a classe c_j / *número total de exemplos*

$$dist(c_j) = \frac{1}{n}\sum_{i=1}^{n}\left|y_i = c_j\right|$$

Distribuição de classes

- Ex.: conjunto de dados com 100 exemplos
  - 60 são da classe c_1
  - 15 são da classe c_2
  - 25 são da classe c_3
  - A distribuição de classe é $dist(c_1,c_2,c_3) = (0,60, 0,15, 0,25) = (60\%, 15\%, 25\%)$
  - A classe c_1 é a classe majoritária ou prevalente
  - A classe c_2 é a classe minoritária

Cross-validation estratificado

- Exemplo:
  - *r = 5*

Objeto	Atributo 1	Atributo 2	Atributo	Classe
4	877	16685	2677	Safra 95
9	877	16823	2677	Safra 95
18	33539	40078	335	Safra 96
11	24628	39437	381	Safra 96
1	855	5142	2708	Safra 95
3	885	16586	2670	Safra 95
14	42329	40307	328	Safra 96
20	34485	40742	358	Safra 96
7	885	19455	2708	Safra 95
10	892	19180	2716	Safra 95
15	41627	40032	335	Safra 96
19	34150	40353	358	Safra 96
6	854	5005	2685	Safra 95
2	854	23155	2716	Safra 95
17	33677	40375	328	Safra 96
12	43183	39277	328	Safra 96
8	839	5027	2708	Safra 95
5	839	5142	2708	Safra 95
15	41627	40032	335	Safra 96
16	39399	40322	335	Safra 96

Cross-validation estratificado

Conjunto de treinamento

Objeto	Atributo 1	Atributo 2	Atributo	Classe
1	855	5142	2708	Safra 95
3	885	16586	2670	Safra 95
14	42329	40307	328	Safra 96
20	34485	40742	358	Safra 96
7	885	19455	2708	Safra 95
10	892	19180	2716	Safra 95
15	41627	40032	335	Safra 96
19	34150	40353	358	Safra 96
6	854	5005	2685	Safra 95
2	854	23155	2716	Safra 95
17	33677	40375	328	Safra 96
12	43183	39277	328	Safra 96
8	839	5027	2708	Safra 95
5	839	5142	2708	Safra 95
15	41627	40032	335	Safra 96
16	39399	40322	335	Safra 96

Conjunto de teste

Objeto	Atributo 1	Atributo 2	Atributo	Classe
4	877	16685	2677	Safra 95
9	877	16823	2677	Safra 95
18	33539	40078	335	Safra 96
11	24628	39437	381	Safra 96

Cross-validation estratificado

Conjunto de treinamento

Objeto	Atributo 1	Atributo 2	Atributo	Classe
4	877	16685	2677	Safra 95
9	877	16823	2677	Safra 95
18	33539	40078	335	Safra 96
11	24628	39437	381	Safra 96
7	885	19455	2708	Safra 95
10	892	19180	2716	Safra 95
15	41627	40032	335	Safra 96
19	34150	40353	358	Safra 96
6	854	5005	2685	Safra 95
2	854	23155	2716	Safra 95
17	33677	40375	328	Safra 96
12	43183	39277	328	Safra 96
8	839	5027	2708	Safra 95
5	839	5142	2708	Safra 95
15	41627	40032	335	Safra 96
16	39399	40322	335	Safra 96

Conjunto de teste

Objeto	Atributo 1	Atributo 2	Atributo	Classe
1	855	5142	2708	Safra 95
3	885	16586	2670	Safra 95
14	42329	40307	328	Safra 96
20	34485	40742	358	Safra 96

Cross-validation estratificado

Conjunto de treinamento

Objeto	Atributo 1	Atributo 2	Atributo	Classe
4	877	16685	2677	Safra 95
9	877	16823	2677	Safra 95
18	33539	40078	335	Safra 96
11	24628	39437	381	Safra 96
1	855	5142	2708	Safra 95
3	885	16586	2670	Safra 95
14	42329	40307	328	Safra 96
20	34485	40742	358	Safra 96
6	854	5005	2685	Safra 95
2	854	23155	2716	Safra 95
17	33677	40375	328	Safra 96
12	43183	39277	328	Safra 96
8	839	5027	2708	Safra 95
5	839	5142	2708	Safra 95
15	41627	40032	335	Safra 96
16	39399	40322	335	Safra 96

Conjunto de teste

Objeto	Atributo 1	Atributo 2	Atributo	Classe
7	885	19455	2708	Safra 95
10	892	19180	2716	Safra 95
15	41627	40032	335	Safra 96
19	34150	40353	358	Safra 96

Cross-validation estratificado

Conjunto de treinamento

Objeto	Atributo 1	Atributo 2	Atributo	Classe
4	877	16685	2677	Safra 95
9	877	16823	2677	Safra 95
18	33539	40078	335	Safra 96
11	24628	39437	381	Safra 96
1	855	5142	2708	Safra 95
3	885	16586	2670	Safra 95
14	42329	40307	328	Safra 96
20	34485	40742	358	Safra 96
7	885	19455	2708	Safra 95
10	892	19180	2716	Safra 95
15	41627	40032	335	Safra 96
19	34150	40353	358	Safra 96
8	839	5027	2708	Safra 95
5	839	5142	2708	Safra 95
15	41627	40032	335	Safra 96
16	39399	40322	335	Safra 96

Conjunto de teste

Objeto	Atributo 1	Atributo 2	Atributo	Classe
6	854	5005	2685	Safra 95
2	854	23155	2716	Safra 95
17	33677	40375	328	Safra 96
12	43183	39277	328	Safra 96

Cross-validation estratificado

Conjunto de treinamento

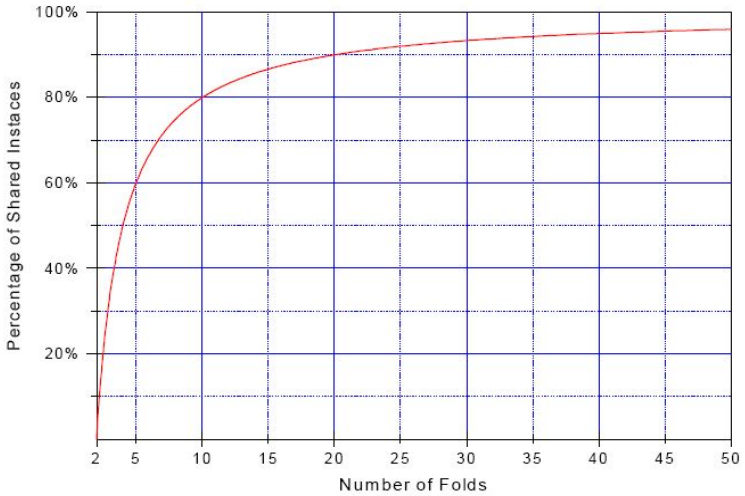
Objeto	Atributo 1	Atributo 2	Atributo	Classe
4	877	16685	2677	Safra 95
9	877	16823	2677	Safra 95
18	33539	40078	335	Safra 96
11	24628	39437	381	Safra 96
1	855	5142	2708	Safra 95
3	885	16586	2670	Safra 95
14	42329	40307	328	Safra 96
20	34485	40742	358	Safra 96
7	885	19455	2708	Safra 95
10	892	19180	2716	Safra 95
15	41627	40032	335	Safra 96
19	34150	40353	358	Safra 96
6	854	5005	2685	Safra 95
2	854	23155	2716	Safra 95
17	33677	40375	328	Safra 96
12	43183	39277	328	Safra 96

Conjunto de teste

Objeto	Atributo 1	Atributo 2	Atributo	Classe
8	839	5027	2708	Safra 95
5	839	5142	2708	Safra 95
15	41627	40032	335	Safra 96
16	39399	40322	335	Safra 96

Validação cruzada

- Crítica: uma parte dos dados é partilhada entre os subconjuntos de treinamento



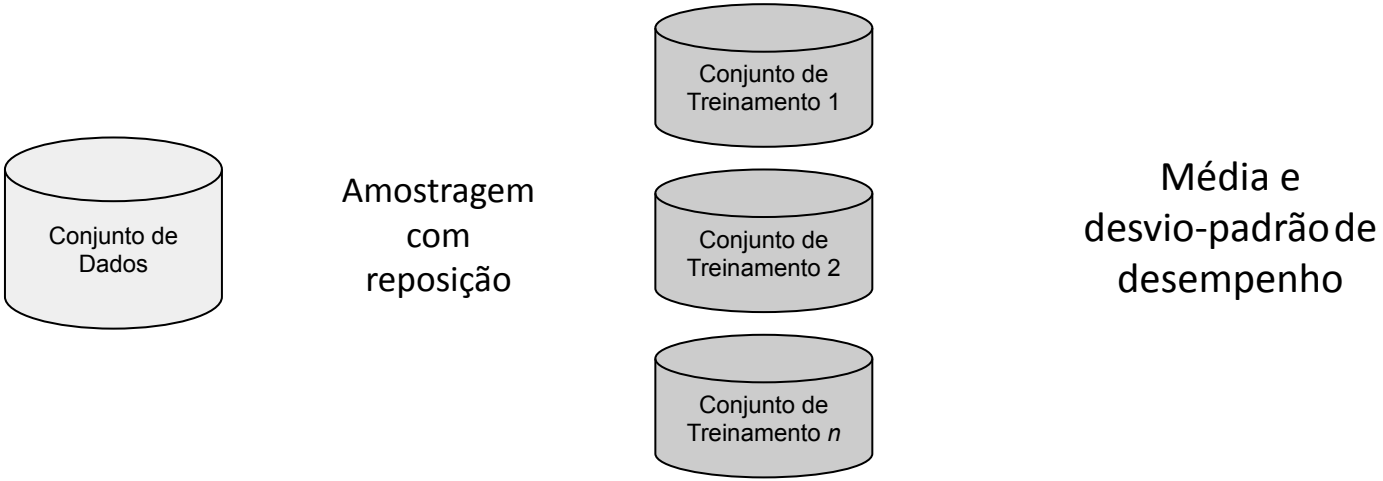
Para $r \geq 2$, uma proporção de $(1 - 2/r)$ dos objetos é compartilhada

Ex. $r = 10 \Rightarrow$ 80% dos objetos são compartilhados

Bootstrap

- Baseado em amostragem com reposição
  - *r* subconjuntos de treinamento são amostrados, com reposição
    - Um exemplo pode estar presente mais de uma vez em um conjunto de treinamento
  - Exemplos não selecionados compõem conjuntos de teste
  - Desempenho: média dos desempenhos nos testes
  - Valor típico para *r*: 100 ou mais
  - É um procedimento custoso aplicado em conjuntos pequenos

Bootstrap


Conjunto de Dados
Amostragem com reposição
Conjunto de Treinamento 1
Conjunto de Treinamento 2
Conjunto de Treinamento *n*
Média e desvio-padrão de desempenho

Amostragem

- Observações:
  - Para médias de desempenho, é importante reportar também os valores de desvio-padrão
    - Alto desvio padrão ⇒ alta variabilidade dos resultados
    - Indicativo de sensibilidade a variações nos dados de treinamento
  - Estimativas mais precisas também podem ser obtidas usando intervalos de confiança

Classificação binária

Seja um problema com duas classes: + e -

Matriz de confusão:

- VP: verdadeiros positivos
  - Número de exemplos da classe +
  - classificados corretamente
- VN: verdadeiros negativos
  - Número de exemplos da classe -
  - classificados corretamente
- FP: falsos positivos
  - Número de exemplos da classe -
  - classificados incorretamente como +
- FP: falsos negativos
  - Número de exemplos da classe +
  - classificados incorretamente como -

Classificação binária

		Classe Predita	
		+	-
Classe Verdadeira	+	VP	FN
	-	FP	VN

Medidas de desempenho

Outras medidas calculadas por matriz de confusão:

- Taxa de erro na classe + (taxa de falsos negativos):
  - Proporção de exemplos da classe + incorretamente classificados
  - *err+(f) = FN/(VP+FN)*
- Taxa de erro na classe - (taxa de falsos positivos):
  - Proporção de exemplos da classe - incorretamente classificados
  - *err-(f) = FP/(VN+FP)*

Medidas de desempenho

Outras medidas calculadas por matriz de confusão:

- Taxa de erro total:
  - Soma da diagonal secundária da matriz dividida pelo número total de exemplos
  - *err(f) = (FP + FN)/n*
- Taxa de acerto ou acurácia total:
  - Soma da diagonal principal dividida pelo número total de exemplos
  - *acc(f) = (VP + VN)/n*

Medidas de desempenho

Outras medidas calculadas por matriz de confusão:

- Precisão
  - Proporção de exemplos + classificados corretamente entre os preditos como +
  - *prec(f) = VP/(VP + FP)*
- Revocação
  - Taxa de acerto na classe positiva (taxa de verdadeiros positivos)
  - *rev(f) = VP/(VP + FN)*

Precisão vs revocação

- Precisão: exatidão do modelo
  - Ex. precisão 1,0 para uma classe *C*: itens rotulados como *C* realmente pertencem a *C*
    - Não fornece informação sobre exemplos de *C* que não foram corretamente classificados
- Revocação: completude do modelo
  - Ex. revocação 1,0 para uma classe *C*: itens da classe *C* foram rotulados como pertencendo a *C*
    - Não fornece informação sobre exemplos que foram classificados incorretamente como *C*

Medidas de desempenho

Outras medidas calculadas por matriz de confusão:

- Precisão e revocação costumam ser discutidas em conjunto, combinadas em uma medida *F*:
  - *F(f) = ((w + 1) rev(f) prec(f))/(rev(f) + w prec(f))*
- Média harmônica da precisão e revocação
  - Usando *w = 1* ⇒ mesmo grau de importância para duas medidas ⇒ *F1*
    - *F1(f) = (2 rev(f) prec(f))/(rev(f) + prec(f))*

Generalizando para mais classes

- Para mais que duas classes:
  - Considera cada uma + e as demais -
  - Ex. *C1*:

	C1	C2	C3
C1	TP	FN	FN
C2	FP	TN	TN
C3	FP	TN	TN

	C1	C2	C3
C1	49	1	0
C2	0	47	3
C3	0	2	48

C1		
	+	-
+	TP	FN
-	FP	TN

C1		
	+	-
+	49	1
-	0	100

erro (+) = (FN/TP+FN) = 0.02

erro (-) = (FP/FP+TN) = 0.00

Generalizando para mais classes

- Para mais que duas classes:
  - Ex. C2:

	C1	C2	C3
C1	TN	FP	TN
C2	FN	TP	FN
C3	TN	FP	TN

	C1	C2	C3
C1	49	1	0
C2	0	47	3
C3	0	2	48

C2		
	+	-
+	TP	FN
-	FP	TN

C2		
	+	-
+	47	3
-	3	97

erro (+) = (FN/TP+FN) = 0.06

erro (-) = (FP/FP+TN) = 0.03

Generalizando para mais classes

- Para mais que duas classes:
  - Ex. C3:

	C1	C2	C3
C1	TN	TN	FP
C2	TN	TN	FP
C3	FN	FN	TP

	C1	C2	C3
C1	49	1	0
C2	0	47	3
C3	0	2	48

C3		
	+	-
+	TP	FN
-	FP	TN

C3		
	+	-
+	48	2
-	3	97

erro (+) = (FN/TP+FN) = 0.04

erro (-) = (FP/FP+TN) = 0.03

Slides construídos com base no material fornecido pela autora do livro 'inteligência artificial: uma abordagem de aprendizado de máquina' (Faceli et al, 2021).