Aprendizagem automática

Naïve Bayes

Luís Rato, Universidade de Évora, 2022/23

Sumário

- Algoritmo
- Características, parâmetros e variações
- Enviesamento e variância
- Ruído

Algoritmo

Aproximação de Bayes

- Objetivo
  - Dados os valores dos atributos $\{a_1, a_2, \ldots, a_n\}$
  - atribuir a **classe c_j mais provável**

$$\arg\max_{c_j \in C} P(c_j | a_1, a_2, \ldots, a_n)$$

- Usando o teorema de Bayes, tem-se

$$\arg\max_{c_j \in C} \frac{P(a_1, a_2, \ldots, a_n | c_j) P(c_j)}{P(a_1, a_2, \ldots, a_n)}$$

constante

$$\arg\max_{c_j \in C} P(a_1, a_2, \ldots, a_n | c_j) P(c_j)$$

teorema de Bayes

$$P(A \mid B) = \frac{P(B \mid A)\, P(A)}{P(B)}$$

Algoritmo Naïve Bayes

- Assume a **independência condicional** dos valores dos atributos dada a classe (este é o pressuposto ingénuo/**Naïve**)

$$P(a_1, a_2, \ldots, a_n|c_j) = P(a_1|c_j) * P(a_2|c_j) * \ldots * P(a_n|c_j)$$

- Construção do modelo
  - estimar as probabilidades
    - de cada classe: $P(c_j)$
    - Sabendo o valor de cada atributo dada a classe: $P(a_i|c_j)$
- Previsão de um exemplo
  - escolher a classe que maximiza a expressão

$$P(c_j) \prod P(a_i|c_j)$$

Exemplo

outlook	*temp*	*humid*	*wind*	*sport*
sunny	hot	high	weak	no
sunny	hot	high	strong	no
overcast	hot	high	weak	yes
rainy	mild	high	weak	yes
rainy	cool	normal	weak	yes
rainy	cool	normal	strong	no
overcast	cool	normal	strong	yes
sunny	mild	high	weak	no
sunny	cool	normal	weak	yes
rainy	mild	normal	weak	yes
sunny	mild	normal	strong	yes
overcast	mild	high	strong	yes
overcast	hot	normal	weak	yes
rainy	mild	high	strong	no

- Qual a classe?
  - x = {sunny, cool, high, strong}
- Probabilidades à priori das classes
  - P(yes) = 9/14
  - P(no) = 5/14

outlook	*temp*	*humid*	*wind*	*sport*
sunny	hot	high	weak	no
sunny	hot	high	strong	no
overcast	hot	high	weak	yes
rainy	mild	high	weak	yes
rainy	cool	normal	weak	yes
rainy	cool	normal	strong	no
overcast	cool	normal	strong	yes
sunny	mild	high	weak	no
sunny	cool	normal	weak	yes
rainy	mild	normal	weak	yes
sunny	mild	normal	strong	yes
overcast	mild	high	strong	yes
overcast	hot	normal	weak	yes
rainy	mild	high	strong	no

- Qual a classe?
  - x = {sunny, cool, high, strong}
- Probabilidades à priori das classes
  - P(yes) = 9/14
  - P(no) = 5/14
- Prob. dos atributos dada a classe
  - P(sunny|yes) = 2/9
  - P(cool|yes) = 3/9
  - P(high|yes) = 3/9
  - P(strong|yes) = 3/9
  - P(sunny|no) = 3/5
  - P(cool|no) = 1/5
  - P(high|no) = 4/5
  - P(strong|no) = 3/5

outlook	*temp*	*humid*	*wind*	*sport*
sunny	hot	high	weak	no
sunny	hot	high	strong	no
overcast	hot	high	weak	yes
rainy	mild	high	weak	yes
rainy	cool	normal	weak	yes
rainy	cool	normal	strong	no
overcast	cool	normal	strong	yes
sunny	mild	high	weak	no
sunny	cool	normal	weak	yes
rainy	mild	normal	weak	yes
sunny	mild	normal	strong	yes
overcast	mild	high	strong	yes
overcast	hot	normal	weak	yes
rainy	mild	high	strong	no

- Qual a classe?
  - x = {sunny, cool, high, strong}
- Probabilidades à priori das classes
  - P(yes) = 9/14
  - P(no) = 5/14
- Prob. dos atributos dada a classe
  - P(sunny|yes) = 2/9
  - P(cool|yes) = 3/9
  - P(high|yes) = 3/9
  - P(strong|yes) = 3/9
  - P(sunny|no) = 3/5
  - P(cool|no) = 1/5
  - P(high|no) = 4/5
  - P(strong|no) = 3/5
- Classes
  - yes : 9/14 * 2/9 * 3/9 * 3/9 * 3/9 = 0.0053
  - **no** : 5/14 * 3/5 * 1/5 * 4/5 * 3/5 = **0.0206**

Estimador de probabilidades

- $P(x) = n_x/total$
  - n_x: número de vezes que x ocorre
  - total: número máximo possível
- Características
  - Se o valor estimado for 0, o termo domina o classificador
  - (sempre que um valor de atributo não aparece no conjunto de treino)

Estimador suavizado

- Estimador

$$P(x) = \frac{n_x + \alpha}{total + \alpha * nvals}$$

- alfa=1
  - estimador de Laplace

- Qual a classe?
  - x = {sunny, cool, high, strong}
  - estimador de Laplace
- Prob. das classes
  - P(yes) = 10/16 (sport: 2 vals diferentes)
  - P(no) = 6/16
- Prob. dos atts dada a classe
  - P(sunny|yes) = 3/12 (outlook: 3 vals diferentes)
  - P(cool|yes) = 4/12 (temp: 3 vals diferentes)
  - P(high|yes) = 4/11 (humid: 2 vals diferentes)
  - P(strong|yes) = 4/11 (wind: 2 vals diferentes)

Parâmetros, características e variações

Parâmetros

- alfa
  - controla a complexidade do modelo
    - valores maiores correspondem a modelos menos complexos (estatísticas mais suavizadas)
  - o desempenho do modelo é relativamente robusto ao valor do alfa
    - a sua definição não é crítica para um bom desempenho

Características

- **Características**
  - "Olha" para cada atributo individualmente
    - assume a **independência** entre atributos (dada a classe)
  - Calcula estatísticas simples de cada atributo por classe
- **Pontos fortes**
  - Algoritmo de aprendizagem e classificação **rápido**
  - Processo de aprendizagem de **fácil** compreensão
  - Produz **bons resultados** em dados de muitas dimensões (muitos atributos)
  - Requer um **número de exemplos relativamente pequeno**
  - Bom baseline

Variações

- Usam outros estimadores / regras de decisão
- Algoritmos
  - Multinomial Naive Bayes
    - para atributos inteiros
  - Gaussian Naive Bayes
    - para atributos contínuos
  - Bernoulli Naive Bayes
    - para atributos binários
  - **Categorical Naive Bayes** (exemplo mostrado atrás)
    - para atributos nominais
  - Complement Naive Bayes
    - para conjuntos de dados desequilibrados

      (proporção desequilibrada entre diferentes classes)

Ruído

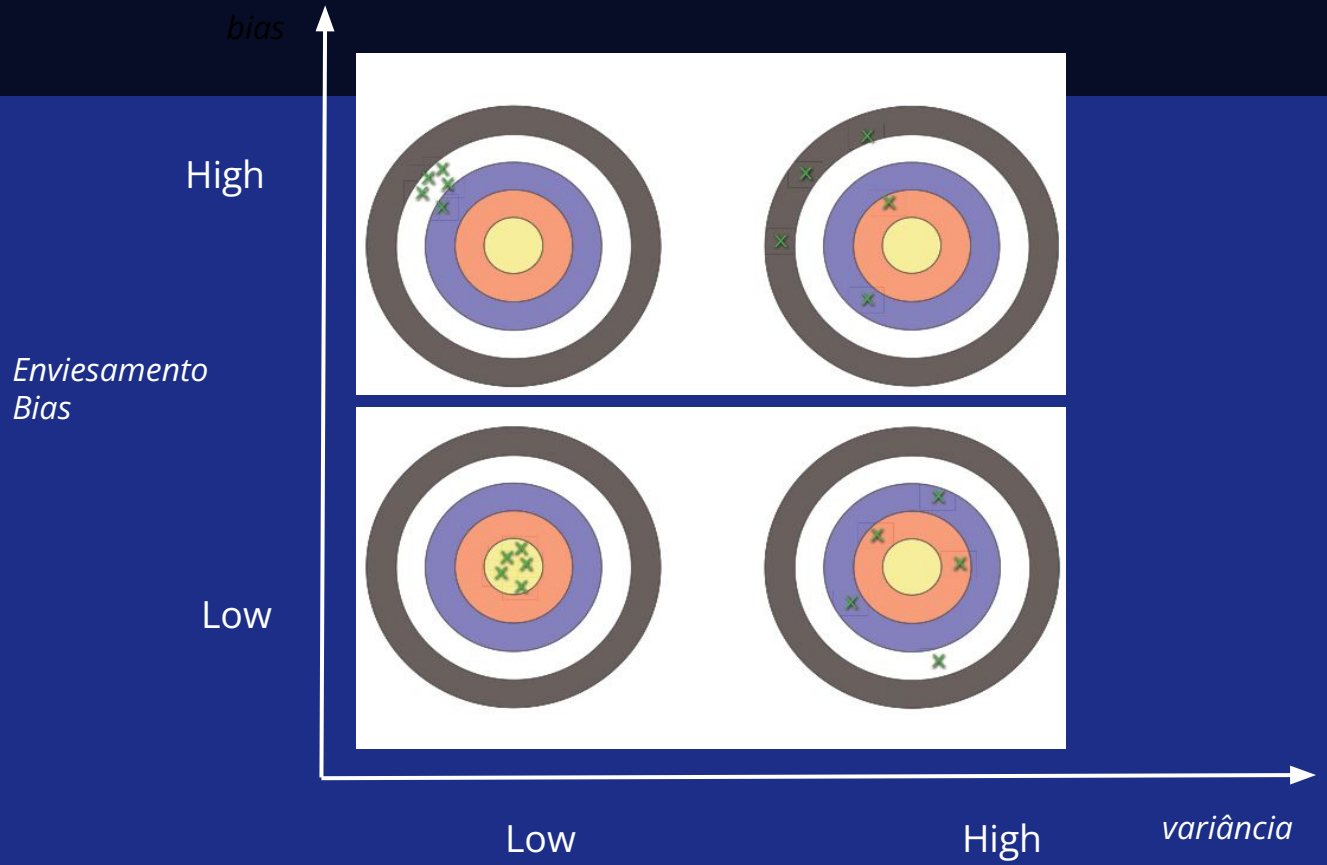
Dados ruidosos

- Ruído
  - Ruído na etiqueta
    - Quando se observa uma **etiqueta corrompida** para uma instância, l’ em vez de l=l(x)
  - Ruído na instância
    - Quando se observa uma **instância corrompida**, x’ em vez de x
- Consequência
  - O algoritmo não deve tentar fazer corresponder exatamente os dados de treino, já que pode conduzir ao **sobre-ajustamento do ruído**

Sobre-ajustamento

- Regra do polegar
  - Para evitar o sobre-ajustamento, o número de parâmetros estimados a partir dos dados deve ser **consideravelmente inferior** ao número de pontos
- Example
  - Polinómio
    - # parâmetros = grau do polinómio +1
    - $a\,x^2 + b\,x + c$

Enviesamento e variância



Dilema enviesamento-variância

- Modelos de baixa complexidade
  - Sofrem menos da variabilidade devido a variações aleatórias nos dados de treino
  - Pode introduzir um enviesamento sistemático
- Modelos de grande complexidade
  - Eliminam o enviesamento
  - Podem sofrer de erros não sistemáticos devido à variância

Classificação com ruído

Classificação com ruído

- Sem enviesamento
  - Aumentar a quantidade dos dados pode compensar a existência de ruído
- Sobre-ajustamento
  - Introdução de ruído funciona como “suavizador”, evita sobre-ajustamento
  - … mas destrói informação.
- Data leakage - Dados em duplicado
  - Introdução de ruído previne em parte a *Data leakage*
  - … mas destrói informação.