

# Aprendizaje bayesiano

# Aprendizaje bayesiano

- Basado en el **teorema de Bayes**

- Permite combinar los datos de ejemplo con conocimiento a priori

- Usado como clasificador, puede obtener probabilidades de pertenecer a cada clase

- Posibilidad de construir representaciones más complejas (Modelos de ocultos de Markov, redes bayesianas, etc).

## Fundamentos. Teorema de Bayes

• **Aprendizaje bayesiano:** buscar la hipótesis  $h$  (de entre todas las  $H$  posibles) más probable si hemos observado una serie de datos  $D$  (*máximo a posteriori o MAP*)

$$h_{MAP} \equiv \operatorname{argmax} P(h | D)$$

• **Base: Teorema de Bayes**

$$P(h | D) = \frac{P(D | h)P(h)}{P(D)}$$

## Fundamentos. MAP y ML

$$h_{MAP} \equiv \operatorname{argmax} P(h | D)$$

*Máximo a posteriori o MAP*

$$h_{MAP} = \operatorname{argmax}_{h \in H} \frac{P(D | h)P(h)}{P(D)}$$

$$h_{MAP} = \operatorname{argmax}_{h \in H} P(D | h)P(h)$$

*ya que  $P(D)$  cte.  
independiente de  $h$*

*Si además suponemos  $P(h)=cte$*

*(a priori, todas las hipótesis son igualmente probables), entonces*

$$h_{ML} = \operatorname{argmax}_{h \in H} P(D | h)$$

*Máxima verosimilitud  
o  
maximum likelihood (ML)*

## Clasificador bayesiano

- Las hipótesis son las clases a las que puede pertenecer un ejemplo
- Suponemos ejemplos caracterizados como **tuplas de atributos**  $\langle a_1, a_2, \dots, a_n \rangle$

$$C_{MAP} \equiv \underset{c_i \in C}{\operatorname{argmax}} P(c_i | a_1, a_2, \dots, a_n)$$

$$C_{MAP} \equiv \underset{c_i \in C}{\operatorname{argmax}} \frac{P(a_1, a_2, \dots, a_n | c_i) P(c_i)}{P(a_1, a_2, \dots, a_n)}$$

$$C_{MAP} \equiv \underset{c_i \in C}{\operatorname{argmax}} P(a_1, a_2, \dots, a_n | c_i) P(c_i)$$

## Estimar las probabilidades

- Necesitamos

$$C_{MAP} \equiv \underset{c_i \in C}{\operatorname{argmax}} P(a_1, a_2, \dots, a_n | c_i) P(c_i) \quad \xrightarrow{\text{blue arrow}} \quad \frac{n^{\circ} \text{ de ejemplos de la clase } i}{n^{\circ} \text{ total de ejemplos}}$$

$$\downarrow \quad \frac{n^{\circ} \text{ de ejemplos con atributos } a_1, a_2, \dots, a_n}{n^{\circ} \text{ de ejemplos de la clase } i} \quad \xrightarrow{\text{red arrow}} \quad \begin{array}{l} n^{\circ} \text{ demasiado pequeño} \\ \text{(estimación inadecuada)} \end{array}$$

**Simplificación:** suponemos que los valores de los atributos son condicionalmente independientes para una clase dada (*naive bayes classifier*)

$$P(a_1, a_2, \dots, a_n | c_i) \simeq \prod_{j=1}^n P(a_j | c_i)$$

## Ejemplo: predicción del tiempo

DÍA	TEMP.	DIR. VIENTO	CIELO	PRESIÓN	Tiempo
1	$\leq 0$	Sur	Nuboso	Subiendo	Sol
2	$> 0$	Oeste	Claro	Estable	Sol
3	$> 0$	Norte	Claro	Subiendo	Sol
4	$> 0$	Norte	Claro	Bajando	Lluvia
5	$> 0$	Oeste	Nuboso	Bajando	Lluvia
6	$\leq 0$	Norte	Nuboso	Bajando	Nieve
7	$> 0$	Sur	Nuboso	Estable	Lluvia
8	$> 0$	Sur	Claro	Subiendo	Sol
9	$\leq 0$	Este	Nuboso	Bajando	Nieve
10	$\leq 0$	Sur	Claro	Estable	Sol

## Ejemplo de clasificador bayesiano

• Predicción del tiempo para  $\langle \text{presión=estable, cielo=claro} \rangle$

$$C_{\text{NaiveBayes}} = \underset{c_i \in \{\text{sol}, \text{lluvia}, \text{nieve}\}}{\operatorname{argmax}} P(c_i) \prod_j P(a_j | c_i)$$

$$\underset{c_i \in \{\text{sol}, \text{lluvia}, \text{nieve}\}}{\operatorname{argmax}} P(c_i) \prod_j P(\text{presión=estable} | c_i) P(\text{cielo=claro} | c_i)$$

$$c_i = \text{sol} \quad \rightarrow \quad 5/10 \quad 2/5 \quad 4/5 = 0.16$$

$$c_i = \text{lluvia} \quad \rightarrow \quad 3/10 \quad 1/3 \quad 1/3 = 0.0333$$

$$c_i = \text{nieve} \quad \rightarrow \quad 2/10 \quad 0 \quad 0 = 0$$

## Ejemplo 2: clasificador de textos

- Clasificar un texto en una categoría predefinida, dados:
- El conjunto de palabras del texto (atributos)

En un lugar de la Mancha, de cuyo nombre no quiero acordarme, no ha mucho tiempo que vivía un hidalgo de los de lanza en astillero, adarga antigua, rocín flaco y galgo corredor

$a_1=en, a_2=un \dots a_{32}=corredor$

- El conjunto de posibles categorías C

$c_1=interesante, c_2=no-interesante$

## Simplificaciones

- Clasificador bayesiano “naïve”: independencia de los atributos

$$P(a_1, a_2 \dots a_n | c_i) = \prod_{j=1}^n P(a_j | c_i)$$

- No importa la posición de las palabras en el texto, solo si están presentes

$$P(a_{31} = galgo | interesante) \simeq P(galgo | interesante)$$

$$\text{Sustituimos } P(a_j | c_i) \text{ por } P(w_j | c_i)$$

## Fase 1: Aprendizaje

- Tomar un conjunto de ejemplos  $x_i \in X$  etiquetados con las clases a las que pertenecen
- $Voc$  = conjunto de palabras en  $X$  (sin considerar preposiciones, artículos, etc.)
  - Para cada clase  $c_j$  calcular  $P(c_j)$ ,  $P(w_k | c_j)$  como sigue:
- Calcular  $docs_j$ , documentos de la clase  $c_j$ 
  - Calcular prob. a priori de la clase  $c_j$   $P(c_j) = \frac{|docs_j|}{|X|}$
- Calcular
  - $text_j$ , concatenación de todos los docs. de  $docs_j$
  - $n$ , número de posiciones en  $text_j$
  - $n_k$ , número de veces que aparece  $w_k$  en  $text_j$
- Calcular  $P(w_k | c_j) = \frac{n_k + 1}{n + |Voc|}$

## Fase 2 - Clasificación

- Dado un documento  $x = w_1, w_2, \dots, w_n$
- Quedarnos con  $pos$ , posiciones de palabras que están contenidas en  $Voc$  (el resto se ignora)
- Devolver la estimación MAP

$$C_{MAP} = \arg \max_{c_j \in C} P(c_j) \prod_{k \in pos} P(w_k | c_j)$$

## Bibliografía

- Mitchell, [Machine Learning](#). McGraw Hill, Computer Science Series. 1997. Capítulo 6