# Naive Bayes

JAFH

# Teorema de bayes

$$\mathrm{P}(A|B) = \frac{P(A) * P(B|A)}{P(B)}$$

# Naive bayes

Un **clasificador es un método supervisado** que permite entrenar un modelo, para que determine **la clase *c* más probable** que debemos asignar a un registro, dado un conjunto de características ***xi***. Para ello el conjunto de entrenamiento tendrá una serie de N registros clasificados, que permiten al ordenador aprender a clasificar nuevos registros sin clase.

# Formula

Dada la variable de clase ***c*** y vector de características ***xi***

$$\mathrm{P}(c|x1, \dots xn) = \frac{P(c) * P(x1, \dots, xn|c)}{P(x1, \dots, xn)}$$

Aplicando el supuesto 'ingenuo' de independencia condicional

$$\mathrm{P}(xi|c, x1, \dots, xi-1, xi+1, \dots xn) = P(xi|c)$$

Tenemos que:

$$\mathrm{P}(c|x1, \dots xn) = \frac{P(c) * \prod_{i=1}^{n} P(xi|c)}{P(x1, \dots, xn)}$$

# Formula Naive Bayes

Dado que P(x1,…,xn) es constante, Tomemos px= P(x1,…,xn)
Vemos que al comparar 2 valores (a/px) y (b/px) , como px es positivo y constante (a/px) > (b/px) implica que a>b dada la entrada. Por tanto como:

- *P*(*X*) es la **probabilidad de la evidencia** (los datos de entrada)
- **Es la misma** para todas las clases *Cj* que estamos comparando
- **P(x) No afecta** la comparación relativa entre clases y lo podemos quitar de la formula

$$\mathrm{P}(c|x1,\ldots xn) = P(c) * \prod_{i=1}^{n} P(xi|c)$$

# Algoritmo

Paso 1: Para cada clase $c_j$, calcular la probabilidad, dados los atributos xi:

$$\mathrm{P}(c_j|x1, \dots xn) = P(c_j) * \prod_{i=1}^{n} P(xi|c_j)$$

Pass 2: seleccionar la clase con mayor probabilidad

# Versión con logaritmos

Para evitar underflow aplicar la formula:

$$Clase\ predicha = \max(\mathrm{c_j}) * [\log\left(P(c_j)\right) + \sum \log(P(xi|cj))]$$

# Estimación de probabilidades

Estimar P(c) es tan sencillo como calcular la frecuencia de la clase en el conjunto de entrenamiento:

$$P(c)=\frac{n_c}{N}$$

Siendo $\boldsymbol{n_c}$ el número de registros de clase **c** en el conjunto de entrenamiento. Y $\boldsymbol{N}$ el total de elementos en el conjunto de entrenamiento.

**Estimar la verosimilitud $P(x_i|c)$ depende de**:

- Si los **atributos $x_i$ son discretos** (caso DNA splicing) se cuentan las frecuencias.
- Si los **atributos $x_i$ son continuos** (caso conjunto Iris) se supone que *cada atributo* se distribuye de acuerdo a una normal

$$x_i|c \sim N(\mu_i^c, (\sigma_i^c)^2)$$

y se utiliza la función de densidad gaussiana:

$$P(x_i|c) = \phi(x_i; \mu_i^c, (\sigma_i^c)^2) = \frac{1}{\sqrt{2\pi}\sigma_i^c} exp\left(-\frac{(x_i - \mu_i^c)^2}{2(\sigma_i^c)^2}\right)$$

# Referencias

https://dcain.etsin.upm.es/~carlos/bookAA/02.1_MetodosdeClasificacion-Naive-Bayes.html