

Aprendizaje Bayesiano

PhD (c) Juan M. Rodríguez





¿A quién está dirigido este curso?

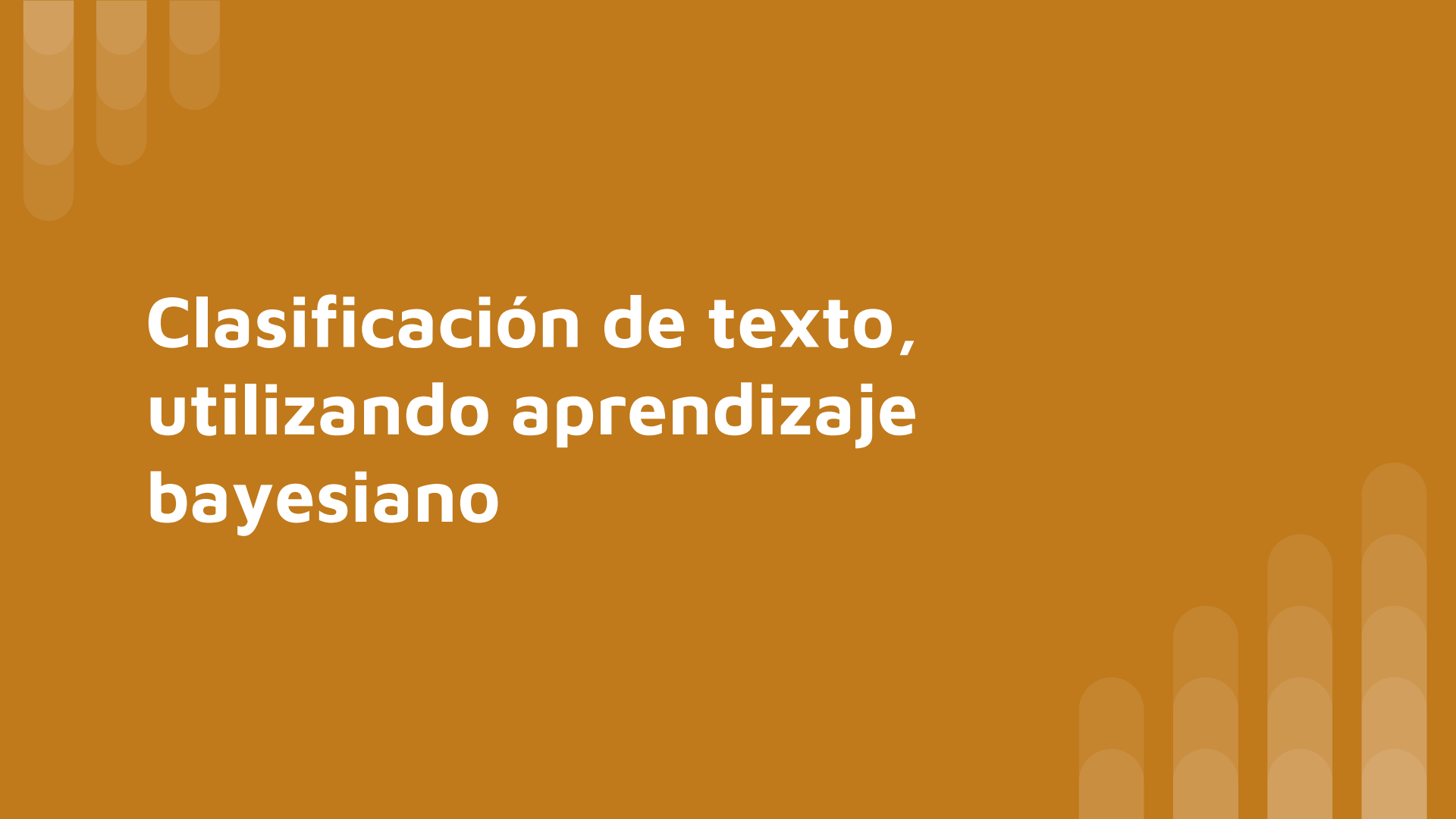
Alumnos en el último año o anteultimo de la carrera de Ingeniería en Informática. Se asume que los alumnos tienen conocimientos sólidos en:

- Programación
- Matemáticas generales
- Probabilidad y estadística



Aprendizaje bayesiano

Uno de los usos más comunes y en donde más éxito ha tenido esta técnica es en la clasificación de documentos o clasificación de texto.



Clasificación de texto, utilizando aprendizaje bayesiano



¿Qué es la clasificación de textos y para qué sirve?

La clasificación de textos sirve para asignar un tópico o categoría de forma automática a cualquier extracto de un texto. Lo podríamos utilizar, por ejemplo para:

- clasificar un email como "spam" o "no spam"
- identificar al autor de un texto
- identificar el sexo o edad del autor de un texto
- identificar el lenguaje en el cual está escrito un texto
- realizar trabajo de análisis de sentimientos (sentiment analysis en inglés)



Clasificación de textos

Definiciones:

Entradas:

- un documento d
- un conjunto prefijado de clases $C = \{c_1, c_2, c_3, \dots, c_j\}$

Salidas:

- un clase c perteneciente al conjunto C



Métodos de Clasificación de Textos

Reglas escritas a manos:

Para la **detección del spam**, por ejemplo, podría tener una serie de reglas escritas por una persona que conozca sobre ese tópico:

REGLA: sí remitente (campo from) está en una lista-negra OR el asunto (subject) contiene la palabra: "viagra" => SPAM

- **Pros**: la precisión puede ser muy alta.
- **Contras**: construir y mantener las reglas puede ser costoso.



Métodos de Clasificación de Textos

Aprendizaje automático supervisado

Entradas:

- un documento d
- un conjunto prefijado de clases $C=\{c_1, c_2, c_3, \dots, c_j\}$
- un conjunto m de documentos clasificados $m=\{(d_1, c_1), \dots, (d_n, c_j)\}$

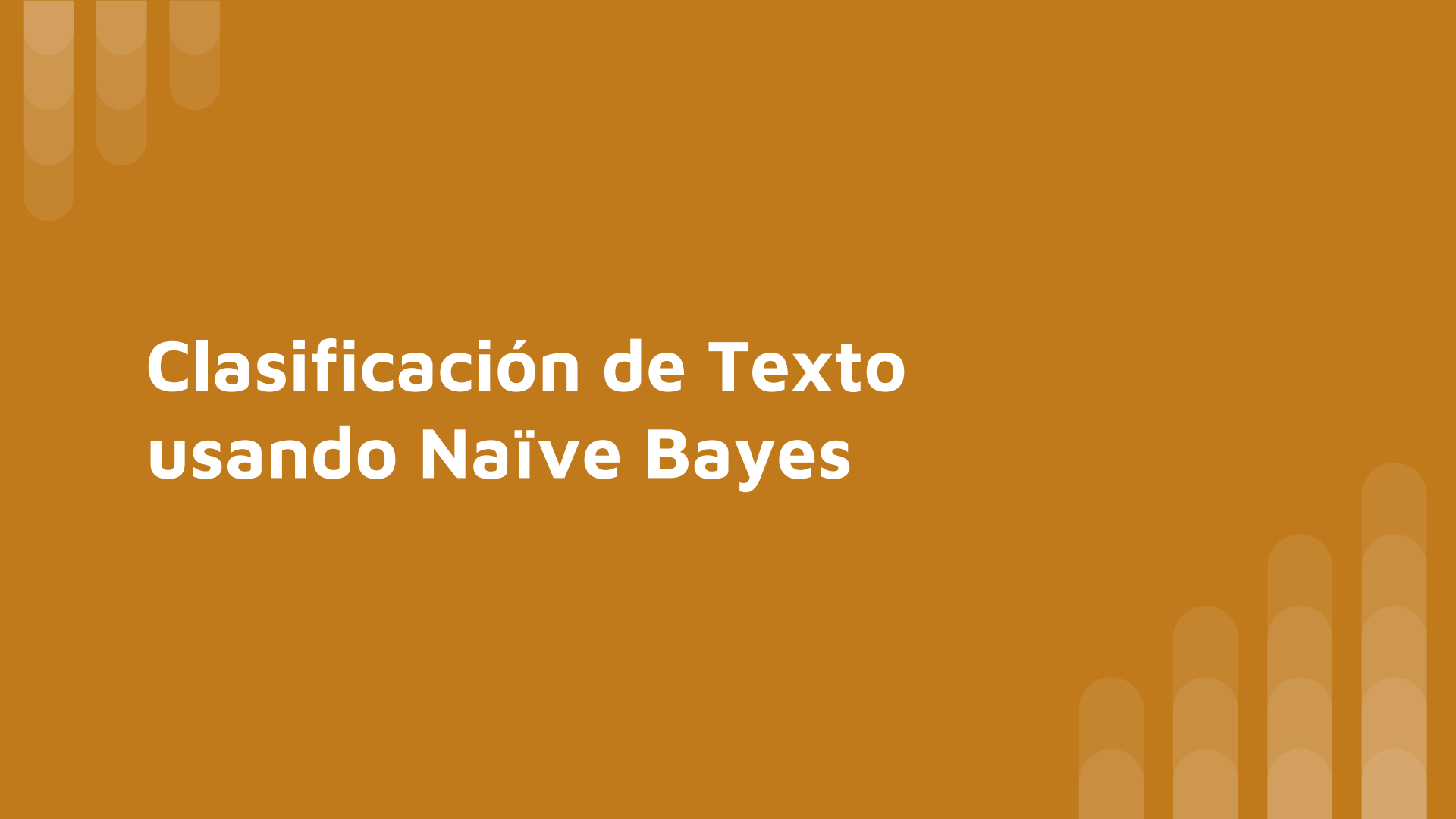
Salidas:

Un clasificador entrenado $y: d \rightarrow c$



Tipos de clasificadores

- Naïve Bayes (Bayes ingenuo o bayes simple)
- Logistic Regression (Regresión logística)
- Support-Vector Machines (Maquinas de Soporte de vectores)
- K-Nearest Neighbors (K-vecinos más cercanos)



Clasificación de Texto usando Naïve Bayes



Naïve Bayes - clasificación de texto

Un enfoque posible para la resolución del problema de la clasificación de texto es encararlo por el lado estadístico, entonces diría que si tengo n documentos y x clases posibles, podría preguntarme:

¿cuál es la probabilidad de que el documento d pertenezca a la clase c ?



Naïve Bayes - clasificación de texto

¿cuál es la probabilidad de que el documento **d** pertenezca a la clase **c**?

Parafraseado como probabilidad condicional:

Dado el documento **d**, ¿cuál es la probabilidad de que pertenezca a **c**?
 $= P(c \mid d)$

Y por el teorema de Bayes se puede plantear lo siguiente:

$$P(c \mid d) = \frac{P(d \mid c) P(c)}{P(d)}$$



Naïve Bayes - clasificación de texto

Si tengo un **conjunto C** de clases, según Bayes un documento **d**, pertenecerá a aquella clase que maximice su probabilidad condicional:

$C_{\text{map}} = \text{argmax } P(c \mid d)$ para $c \in C$, (el conjunto de todas las clases).

- **map**: máximo a posteriori, C_{map} es la clase candidata
- **argmax**: función que devuelve el argumento máximo



Naïve Bayes - clasificación de texto

$C_{\text{map}} = \text{argmax } P(c \mid d)$ para $c \in \mathbf{C}$, (el conjunto de todas las clases).

Por Bayes:

En el denominador nos queda **$P(d)$** como una constante. Lo eliminamos.

$$C_{\text{map}} = \frac{\text{argmax } P(d \mid c) P(c)}{P(d)}, \quad c \in \mathbf{C}$$

$$C_{\text{map}} = \text{argmax } P(d \mid c) P(c)$$



Naïve Bayes - clasificación de texto

$C_{\text{map}} = \text{argmax } P(d \mid c) P(c)$ para $c \in \mathbf{C}$, (el conjunto de todas las clases).

- ¿Cómo calculamos $P(d \mid c)$?
- ¿Cómo calculamos $P(c)$?



Naïve Bayes - clasificación de texto

$P(c)$ es la probabilidad que tiene la clase de aparecer en una cantidad dada de documentos.

$$p(c) = \frac{\text{cantidad de documentos de clase } c}{\text{cantidad de documentos totales}}$$

Este valor no podemos conocerlo. Pero usando el conjunto de entrenamiento T , podemos estimar cuál sería esta probabilidad:

$$p'(c) = \frac{\text{cantidad de documentos de clase } c \text{ en } T}{\text{cantidad de documentos totales en } T}$$

Naïve Bayes - clasificación de texto

$P(d|c)$ es la probabilidad de que dada una clase c , d sea un documento de ella. Esto es un poco más difícil de identificar. Y para ello tendremos que definir una forma de representar un documento.

Un documento, para Bayes Naive será una bolsa de características: $x_1, x_2, x_3, \dots, x_n$
Para nosotros, estas características serán las palabras que componen al documento.

Para ello asumiremos dos supuestos, muy importantes:

- No importa el orden de las palabras
- Las probabilidades de cada característica, dada una clase c : $P(x_i | c)$ independientes entre sí





Naïve Bayes - clasificación de texto

Problemas con los supuestos:

Él es una buena persona y no un violento = Él es un violento y no una buena persona

- Buenos
- Nueva
- Troche y
- Aires
- York
- Moche

Naïve = ingenuo



Naïve Bayes - clasificación de texto

Una comedia entretenida que nos muestra la pasión por la música, la amistad, el amor y los conflictos en las relaciones humanas. Un guión sin desperdicio, una dirección con profesionalismo y actuaciones memorables. Muy recomendable para ver en familia.



Filtramos palabras (opcional)

xxx comedia entretenida xxx xxx xxxxxxxx xx xxxxx xxx xx xxxx, xx xxxxx, xx xxxx x xxx
amor xx xxx xxxxxxxx xxxxxxxx xx xxxxx xxx xxxxxxxxxxx, xxxx xxxxxxxx xxx
profesionalismo x xxxxxxxxxxxx memorables xxxx recomendable xxxx xxx xx xxxxxxxx



Naïve Bayes - clasificación de texto

Tokenización y cuenta de palabras:

Palabras	Cantidad de apariciones
amor	1
recomendable	2
comercial	2
memorables	2
..	...



Naïve Bayes - clasificación de texto

Retomando las fórmulas, ahora que sabemos cómo representar un documento:

$$P(d|c) = P(x_1, x_2, x_3, \dots, x_n | c)$$

$$P(x_1, x_2, x_3, \dots, x_n | c) = P(x_1|c) * P(x_2|c) * P(x_3|c) * \dots * P(x_n | c)$$

$$C_{\text{map}} = \underset{c_j \in C}{\text{argmax}} P(c_j) \prod_{i \in \text{Posiciones}} P(x_i | c_j)$$



Naïve Bayes - clasificación de texto

¿Cómo calcular $P(x_i | c_j)$?

$p(w_i|c) = \frac{\text{cantidad de veces que aparece } w_i \text{ en documentos en la clase } c}{\text{cantidad de palabras que aparecen en los documentos de la clase } c}$

Nuevamente no conozco estos valores, pero los puedo estimar del conjunto de entrenamiento T

$p'(w_i|c) = \frac{\text{cantidad de veces que aparece } w_i \text{ en documentos en la clase } c \text{ en } T}{\text{cantidad de palabras que aparecen en los documentos de la clase } c \text{ en } T}$



Naïve Bayes - clasificación de texto

Finalmente, estas son las dos ecuaciones que tenemos que calcular para entrenar un clasificador Bayes Naive

$$p'(w_i | c_j) = \frac{\text{cantidad}(w_i | c_j)}{\sum_{w \in V} \text{cantidad}(w, c_j)}$$

V: vocabulario (según T)

$$p'(c_j) = \frac{\text{cantidad de documentos de clase } c_j \text{ en } T}{\text{cantidad de documentos totales en } T}$$



Ejemplo de entrenamiento

Supongamos lo siguiente:

Ya entrené un clasificador Bayes Naive con un conjunto de entrenamiento:

- **Clases:** “Críticas Positivas”, “Críticas Negativas”
- Conjunto de entrenamiento: 1000 críticas cinematográficas de IMDB. 500 y 500

Lo pongo a prueba con una nueva crítica:

En el documento de prueba aparece por primera vez la palabra *fantástica*

- cantidad ("fantástica" | "Críticas Positivas") = 0
- cantidad ("fantástica" | "Críticas Negativas") = 0

Laplace smoothing



Laplace smoothing

La forma de solucionar esto es con *Laplace smoothing* también conocido como *Add-one*

Laplace smoothing aplicado a Naïve Bayes:

$$p'(w_i | c_j) = \frac{\text{cantidad}(w_i | c_j) + 1}{\sum_{w \in V} (\text{cantidad}(w, c_j) + 1)}$$



Laplace smoothing

Sumamos 1 a cada cantidad(w_i, c_j) calculada, y normalizamos agregando uno también por cada $\mathbf{w} \in \mathbf{V}$, o lo que es lo mismo sumamos en el denominador la cantidad de palabras en el vocabulario:

$$p'(w_i | c_j) = \frac{\text{cantidad}(w_i | c_j) + 1}{\sum_{w \in V} \text{cantidad}(w, c_j) + |V|}$$

$$p'(\text{"fantástica"} | c) = \frac{1}{\sum_{w \in V} \text{cantidad}(w, c) + |V| + 1}$$

Naïve Bayes paso a paso con Laplace Smoothing



Naïve Bayes paso a paso

Corpus	Documento	Palabras	Clase
Entrenamiento	1	Chileno Santiago Chileno	C
	2	Chileno Chileno Valparaiso	C
	3	Chileno Allende	C
	4	Montevideo Uruguay Chileno	U
Prueba	5	Chileno Chileno Chileno Montevideo Uruguay	?



Naïve Bayes paso a paso

Las fórmulas de Naïve Bayes son:

$$p'(c) = \frac{N_c}{N}$$

donde N es el número de documentos y N_c los documentos de la clase C

$$p'(w_i | c) = \frac{\text{cantidad}(w_i | c) + 1}{\text{cantidad}(w | c) + |V|} \quad (\text{para todo } w \text{ en docs de } C)$$

y luego calculamos la clase del documento 5 viendo cual maximiza su probabilidad:

$$C_{\text{map}} = \underset{c_j \in C}{\text{argmax}} P(c_j) \prod P(x_i | c_j)$$



Naïve Bayes paso a paso - resultados

- $P(C) = 3/4$ Tengo 3 documentos de clase C, de un total de 4
- $P(U) = 1/4$ Tengo 1 solo documento de clase U, de un total de 4

$|V| = 6$ Estoy considerando todas las palabras una única vez. Incluso las del documento de prueba, aunque este documento no aporta palabras nuevas.

- $P(\text{Chileno} | C) = (5+1) / (8+6) = 6/14 = 3/7$
- $P(\text{Montevideo} | C) = (0+1) / (8+6) = 1/14$
- $P(\text{Uruguay} | C) = (0+1) / (8+6) = 1/14$
- $P(\text{Chileno} | U) = (1+1) / (3+6) = 2/9$
- $P(\text{Montevideo} | U) = (1+1) / (3+6) = 2/9$
- $P(\text{Uruguay} | U) = (1+1) / (3+6) = 2/9$
- $P(C | \text{doc5}) \propto 3/4 * 3/7 * 3/7 * 3/7 * 1/14 * 1/14 \approx 0.0003$
- $P(U | \text{doc5}) \propto 1/4 * 2/9 * 2/9 * 2/9 * 2/9 * 2/9 \approx 0.0001$

**CLASE C:
CHILE**



Redes Bayesianas



Redes Bayesianas

¿Cómo se puede entender o modelar el conocimiento de un clasificador Bayes Naive?

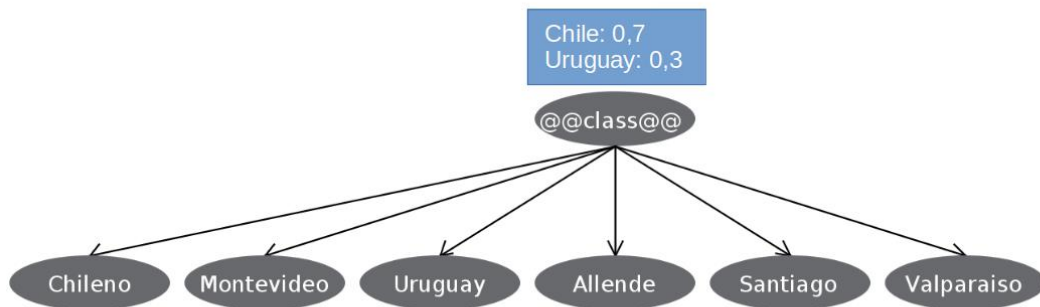
Redes Bayesianas:

- Grafo aciclico dirigido
- Los nodos representan variables
- Las aristas representan dependencias condicionales



Redes Bayesianas

En este caso, cada palabra depende de la clase: “Chile” o “Uruguay”, pero no hay dependencias entre ellas, ya que Bayes Naive ignora estas dependencias.





Redes Bayesianas

Veamos un ejemplo más complejo

Una persona en los Ángeles compró una alarma “anti-robo”

La alarma puede activarse si entra un ladrón con cierta probabilidad.

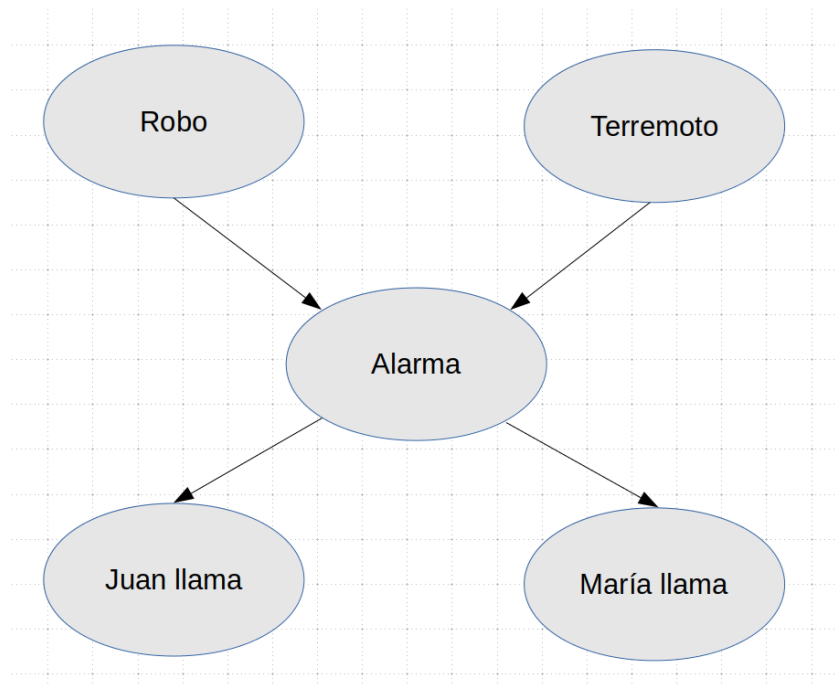
A veces se activa por un pequeño terremoto y no porque haya habido un robo.

A su vez a esta persona, lo pueden llamar a la oficina unos vecinos: Juan y María. Si es que ellos escuchan sonar la alarma, cosa que podría pasar con cierta probabilidad.



Redes Bayesianas

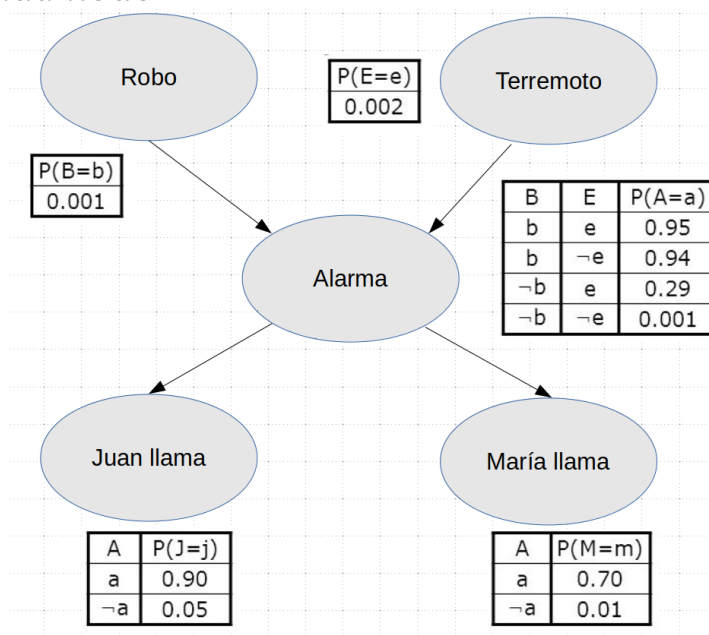
Veamos un
ejemplo más
complejo





Redes Bayesianas

Las redes bayesianas tienen asociada una tabla con probabilidades condicionales por cada nodo





Redes Bayesianas

Las redes bayesianas permiten realizar inferencia, según la observación de un evento. Por ejemplo:

- Juan llama = true
- María llama = true

¿Cual es la probabilidad de la ocurrencia de un robo?

$P(\text{Robo} | \text{Juan llama} = \text{true}, \text{María llama} = \text{true}) = \langle 0.284, 0.716 \rangle$



Redes Bayesianas

Según nuestro ejemplo de clasificación de texto, observo la ocurrencia de los “eventos”, es decir palabras en un documento:

- Chileno = true
- Montevideo = true
- Uruguay = true

¿Cual es la probabilidad de que pertenezca a la clase Chile?

$P(\text{Chile} | \text{Chileno} = \text{true}, \text{Montevideo} = \text{true}, \text{Uruguay} = \text{true}) = ?$



Aprendizaje Bayesiano

Si bien las redes bayesianas permiten inferencias mucho más precisas que la versión simplificada que construye Bayes Naive, son más complejas de construir y de mantener.

Por otro lado Bayes Naive conjuga varias características positivas:

- Es muy rápido y requiere poco almacenamiento
- Robusto ante características (palabras) irrelevantes
- Muy bueno en dominios en donde hay muchas características y todas son importantes

Además, si resulta que el supuesto sobre la independencia de las palabras es cierto, Naive Bayes es óptimo.

Bibliografía

- **Inteligencia Artificial Un Enfoque Moderno**, Stuart Russell, Peter Norving
- **Introduction to Information Retrieval**, Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, Introduction to Information Retrieval, Cambridge University Press. 2008.