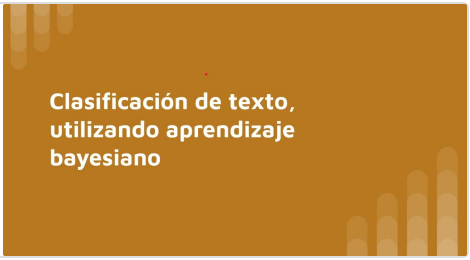
Aprendizaje Bayesiano

PLN Clasificación de texto

Procesamiento de lenguaje natural - Clasificación de Texto utilizando Bayes Naive.
Se explica brevemente como funcionan las redes Bayesianas

https://www.youtube.com/watch?v=a2oX8MLvDxE



Clasificación de texto, utilizando aprendizaje bayesiano
¿Qué es la clasificación de textos y para qué sirve?
Métodos de Clasificación de Textos
Reglas escritas a mano
Aprendizaje automático supervisado
Tipos de clasificadores
Clasificación de Texto usando Naïve Bayes
Supuestos
Bag of Words
Armado del Vocabulario:
Volviendo a Naive Bayes
Ejemplo de entrenamiento
Laplace smoothing
Naïve Bayes paso a paso con Laplace Smoothing
Redes Bayesianas

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 texto es un problema de clasificación como cualquier otro problema de clasificación, solo que en este caso lo que tengo como entrada es texto en lenguaje

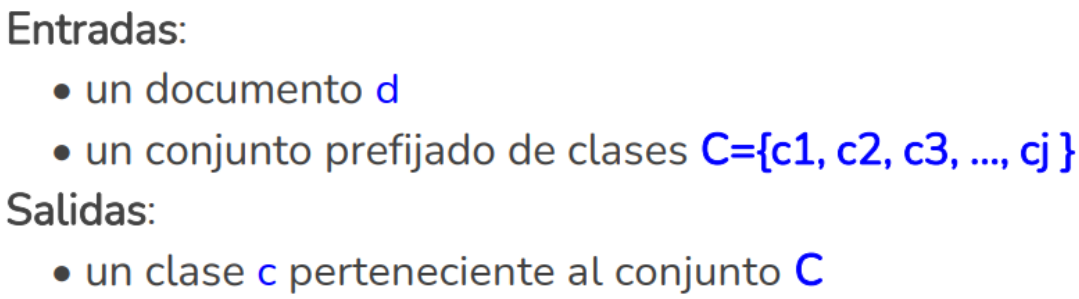
natural.

Lenguaje natural → cualquier lenguaje que hablamos las personas

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)



Las clases a las cual yo quiero asignar este documento las tengo que conocer de antemano y van a ser finitas.

Métodos de Clasificación de Textos

Reglas escritas a mano

Escribir una regla de tipo pattern matching. Si se cumple cierto patron le asigno tal categoria.

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.

Aprendizaje automático supervisado

Entradas:

- un documento **d**
- un conjunto prefijado de clases **C={c1, c2, c3,...,cj}**
- un conjunto m de documentos clasificados **m={(d1,c1),...,(dn,cj)}**

Salidas:

Un clasificador entrenado **y: d → c**

El conjunto m fueron etiquetados a mano.

Tipos de clasificadores

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

Naive Bayes es uno de los que mejor se desempeña en la clasificación de texto.

Clasificación de Texto usando Naïve Bayes

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?

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

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

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

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

C_{map} = **argmax** P(c | d) para **c** ∈ **C**, (el conjunto de todas las clases).

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

Por Bayes:

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

En el denominador nos queda P(d) como una constante. Lo eliminamos.
El máximo de esta formula no va a depender del denominador porque es una constante.

⇒ esto ya no va a ser una probabilidad. Pero eso no importa porque lo que queremos ver es cual es la clase que maximiza este argumento

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

Como calculamos estas probabilidades?

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

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

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

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

Se supone que el conjunto de entrenamiento es representativo de la realidad, representa mas o menos la distribución real de documentos

Método: Bag of words

Un documento, para Bayes Naive será una bolsa de características: x1, x2, x3,... xn

Para nosotros, estas características serán las palabras que componen al documento.

Supuestos

Para ello asumimos dos supuestos, muy importantes:

● No importa el orden de las palabras

● Las probabilidades de cada característica, dada una clase c, son independientes entre sí

$P(x_i \mid c_j)$

Estos dos supuestos son FALSOS

Problemas con los supuestos:

Si no importa el orden de las palabras, entonces estos dos documentos son iguales:

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

Porque la cantidad de palabras son las mismas

Y el problema con el segundo supuesto es que al pensar que cada palabra es independiente y que una palabra que viene después de otra independientemente de la anterior y de la que tiene después, hace que sea exactamente lo mismo por ejemplo que después de la palabra Buenos venga Aires, York o Moche y en realidad lo mas probable es que venga Aires.

● Buenos …..	● Aires
● Nueva ……	● York
● Troche y …...	● Moche

Pero esto a Bayes no le importa

Por eso se lo llama ingenuo (Naive)

Bag of Words

Bayes no sabe cómo trabajar con palabras, con textos o con cadenas de caracteres.

Es por ello que tenemos que convertir los textos en números, o más específicamente en vectores, para poder manipularlos.

Hay varios modelos que permiten convertir textos a vectores, utilizaremos el más sencillo:
Bag of Words → Como su nombre lo indica se trata de una bolsa de palabras.

Es una **bolsa de palabras** y no una lista de palabras o un conjunto de palabras porque:

- No están ordenadas las palabras de ninguna forma
- Cada palabra puede aparecer más de una vez

Armado del Vocabulario:

Supongamos que tenemos un conjunto de entrenamiento con 4 textos.

Contamos entonces todas las palabras que aparecen en todos los documentos y armamos una lista de palabras únicas.

Convertimos los textos a vectores

Documento	Texto
1	Que mala película
2	Que buena película
3	Odio esta película
4	Amo esta película. La Amo.

Vocabulario = { Que, mala, película, buena, Odio, esta, Amo, la}

|V| = 8 (Vocabulario tiene 8 elementos)

Ejemplo:

Vocabulario = { Que, mala, película, buena, Odio, esta, Amo, la}
|V| = 8 (Vocabulario tiene 8 elementos)

Documento	Texto	Bag of Word
1	Que mala película	[1,1,1,0,0,0,0,0]
2	Que buena película	[1,0,1,1,0,0,0,0]
3	Odio esta película	[0,0,1,0,1,1,0,0]
4	Amo esta película. La Amo.	[0,0,1,0,0,1,2,1]

El resto de las palabras no están Presentes en el documento

La palabra “Amo” está dos veces.

Podemos usar solo algunas palabras en lugar de todas.

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 xxxxxxx xx xxxxx xxx xx xxxx, xx xxxxx, xx xxxx x xxx amor xx xxx xxxxxxxx xxxxxxxx xx xxxx xxx xxxxxxxxx, xxxx xxxxxxx xxx profesionalismo x xxxxxxxxxx memorables xxxx recomendable xxxx xxx xx xxxxxxx

Esta es otra tecnica, donde si tenemos textos largos quizas no nos interesa arrays de numeros muy largos sino que nos interezan solo algunas palabras ⇒ podemos filtrar las palabras y quedarnos con palabras que tengan carga de valor (por ejemplo adjetivos calificativos o sustantivos como comedia)

Para obtener las palabras, primero debemos **tokenizar** el texto.

Lo que originalmente es para nosotros una gran cadena de caracteres, la tenemos que partir en cadenas más pequeñas o tokens. Estos tokens coinciden aproximadamente con lo que nosotros llamamos coloquialmente: palabra.

La forma habitual de **Tokenización** es separando los caracteres alfabéticos de los demás (a veces también los numéricos), utilizando como caracteres de corte o

separación, el espacio, los signos de puntuación, de exclamación e interrogación.

Volviendo a Naive Bayes

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

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

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

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

x1, x2, … , xn son las características o palabras

Y como suponemos que son independientes, por la regla de la probabilidad, entonces P(d|c) es la multiplicación de cada una de las probabilidades que tiene cada palabra de pertenecer a la clase c dada.

¿Cómo calcular P(xi | cj) ?

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

clase c es dato, c es la certeza de esta probabilidad condicional

Sabiendo que es un documento de la clase c, cual es la probabilidad de ocurrencia de la palabra wi

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

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

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

$$p'(w_i \mid c_j) = \frac{\text{cantidad } (w_i \mid c_j)}{\sum \text{cantidad}(w, c_j)} \qquad w \in \mathbf{V} \quad \mathbf{V}\text{: vocabulario (según T)}$$

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

(se pone el apostrofe porque no es exactamente una probabilidad)

Ejemplo de entrenamiento

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

Es cero porque nunca se había visto esa palabra. Eso va a hacer que toda la ecuación se haga cero por ser una multiplicatoria, entonces la probabilidad va a ser cero para esa palabra

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

Laplace smoothing

Sumamos 1 a cada cantidad(wi,cj) calculada, y normalizamos agregando uno también por cada w∈V, o lo que es lo mismo sumamos en el denominador la cantidad de palabras en el vocabulario.

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

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

Cuando hay una palabra nueva, agrego 1 al vocabulario

Laplace smoothing aplicado a Naïve Bayes:

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

Naïve Bayes paso a paso con Laplace Smoothing

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	?

Documentos sobre Chile son clase C y los de Uruguay son clase U

Quiero saber a que clase pertenece el quinto documento

Las fórmulas de Naïve Bayes son:

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

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

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

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

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

- 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(Chileno| C) = (5+1) / (8+6) = 6/14 = 3/7
- P(Montevideo| C) = (0+1) / (8+6) = 1/14
- P(Uruguay|C) = (0+1) / (8+6) = 1/14

- P(Chileno| U) = (1+1) / (3+6) = 2/9
- P(Montevideo| U) =(1+1) / (3+6) = 2/9
- P(Uruguay|U) =(1+1) / (3+6) = 2/9

En el primero es 5 la cantidad de veces que aparece la palabra chileno en documentos de clase C y en total tenemos 8 palabras.

- $P(C|doc5) \propto 3/4 * 3/7 * 3/7 * 3/7 * 1/14 * 1/14 \approx 0.0003$
- $P(U|doc5) \propto 1/4 * 2/9 * 2/9 * 2/9 * 2/9 * 2/9 \approx 0.0001$

CLASE C: CHILE

Segun este claificador el documento 5 esta hablando sobre Chile

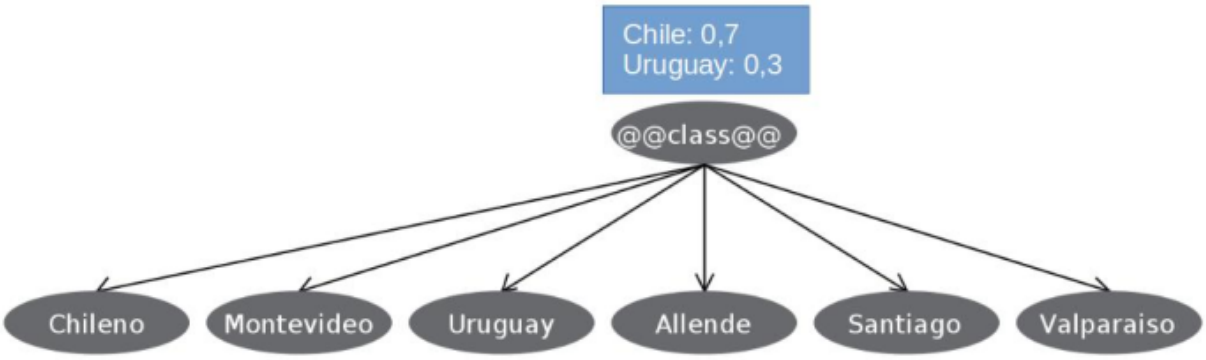
Redes Bayesianas

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

Redes Bayesianas:

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

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



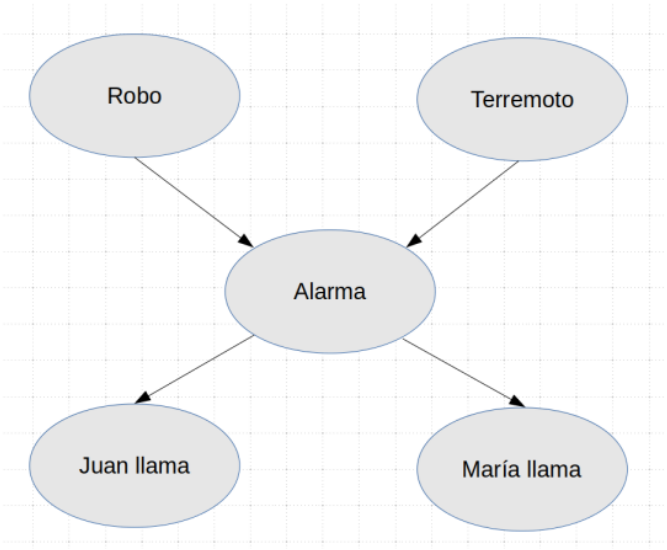
Ejemplo más complejo

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

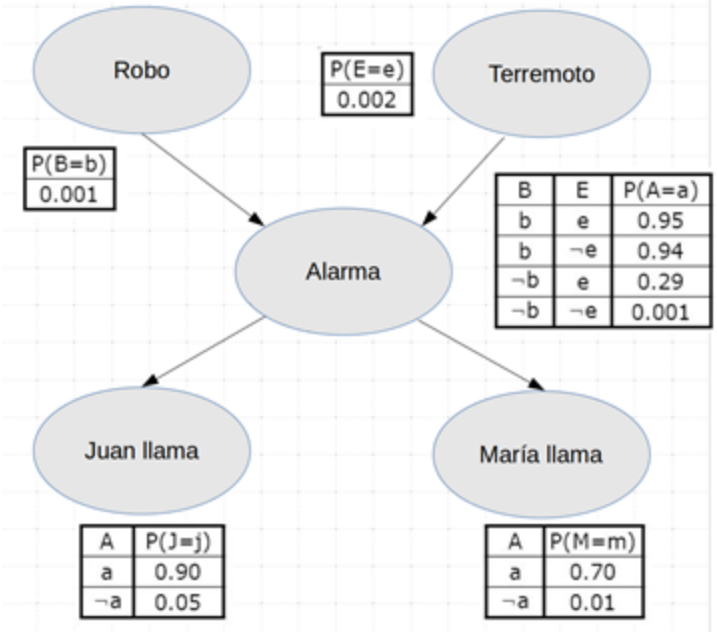
La alarma puede activarse si entra un ladrón con cierta probabilidad.

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.



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



Que Juan llame y Maria llame son eventos independientes. Si dependen de que la alarma suene.

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

Por ejemplo:

- Juan llama = true
- Maria llama = true

Cual es la probabilidad de la ocurrencia de un robo?

P(Robo|Juan llama = true , María llama = true) = <0.284, 0.716>

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(Chile|Chileno = true , Montevideo = true, Uruguay = true) = ?

Si bien las redes bayesianas permiten inferencias mucho más precisas que la versión simplificada que construye Bayes Naive, estas 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.

Naive Bayes: Text Classification Example

In this video, I explain the workings of the naive bayes algorithm using a text classification example.

https://www.youtube.com/watch?v=mqYa0LaA9WI