

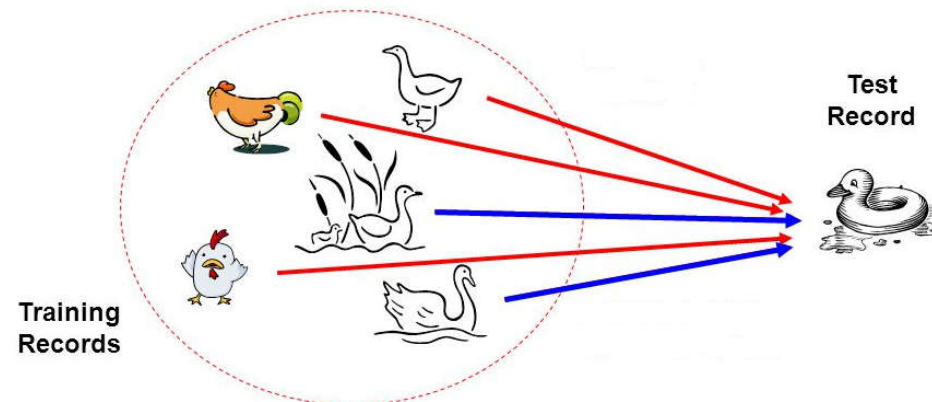
Bayesian Classifier

Simi S

Department of Computer Science& Engineering, School of Engineering, Amritapuri

Naïve Bayes Classifier

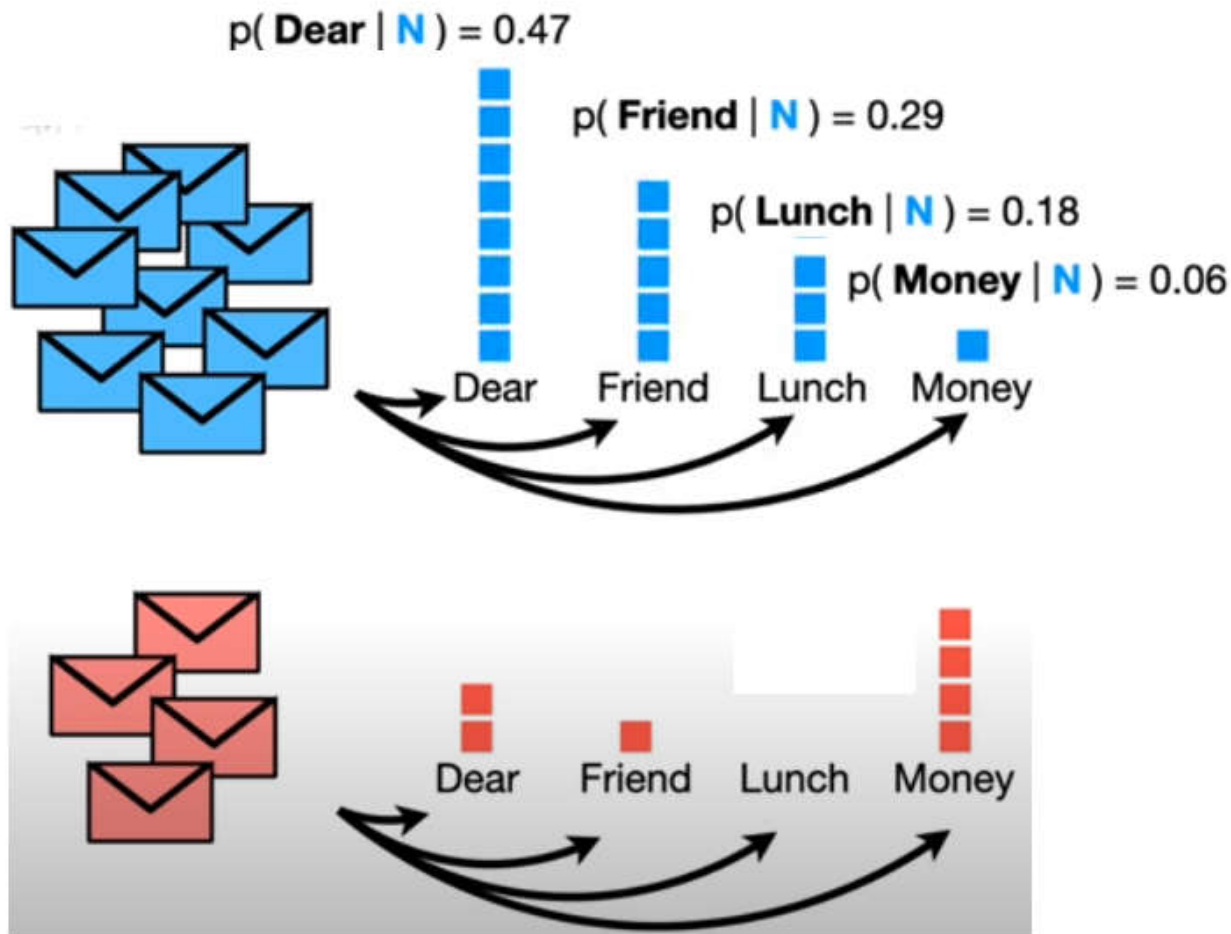
- Given a dataset $X = \{x_1, x_2, \dots, x_m\}$ a set of classes $C = \{c_1, c_2, \dots, c_k\}$, the **classification problem** is to define a mapping $f: X \rightarrow C$, Where each x_i is assigned to one class.
- Naïve Bayes algorithm is a supervised learning algorithm, which is based on **Bayes theorem** and used for solving classification problems
- Probabilistic Approach to Learning. Instead of learning $F: X \rightarrow C$, learn $P(C|X)$.
- can design algorithms that learn functions with uncertain outcomes



Applications

- Face Recognition
- Weather Prediction
- Medical Diagnosis
- News Classification
- ...

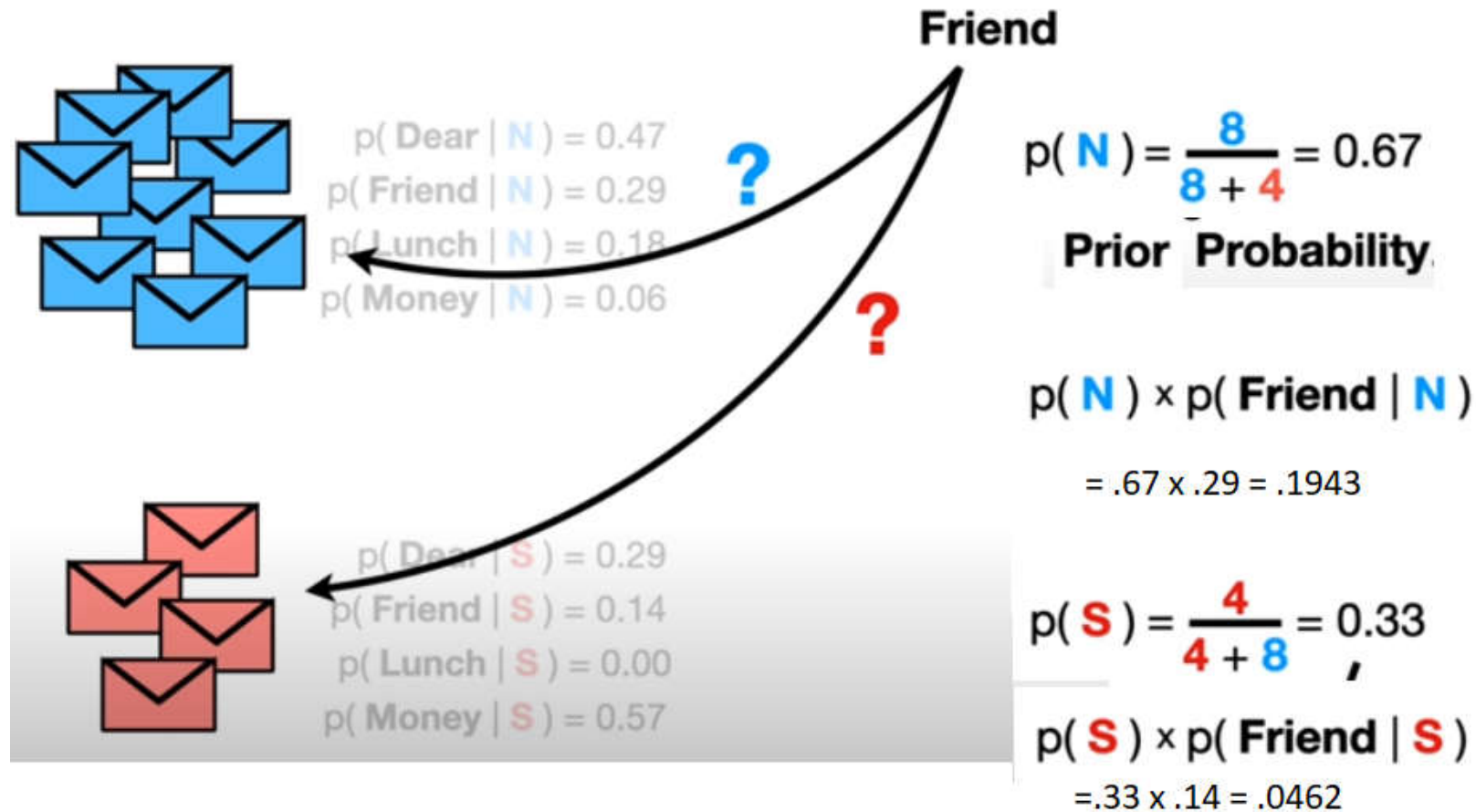
Example: Normal / Spam mail Classification



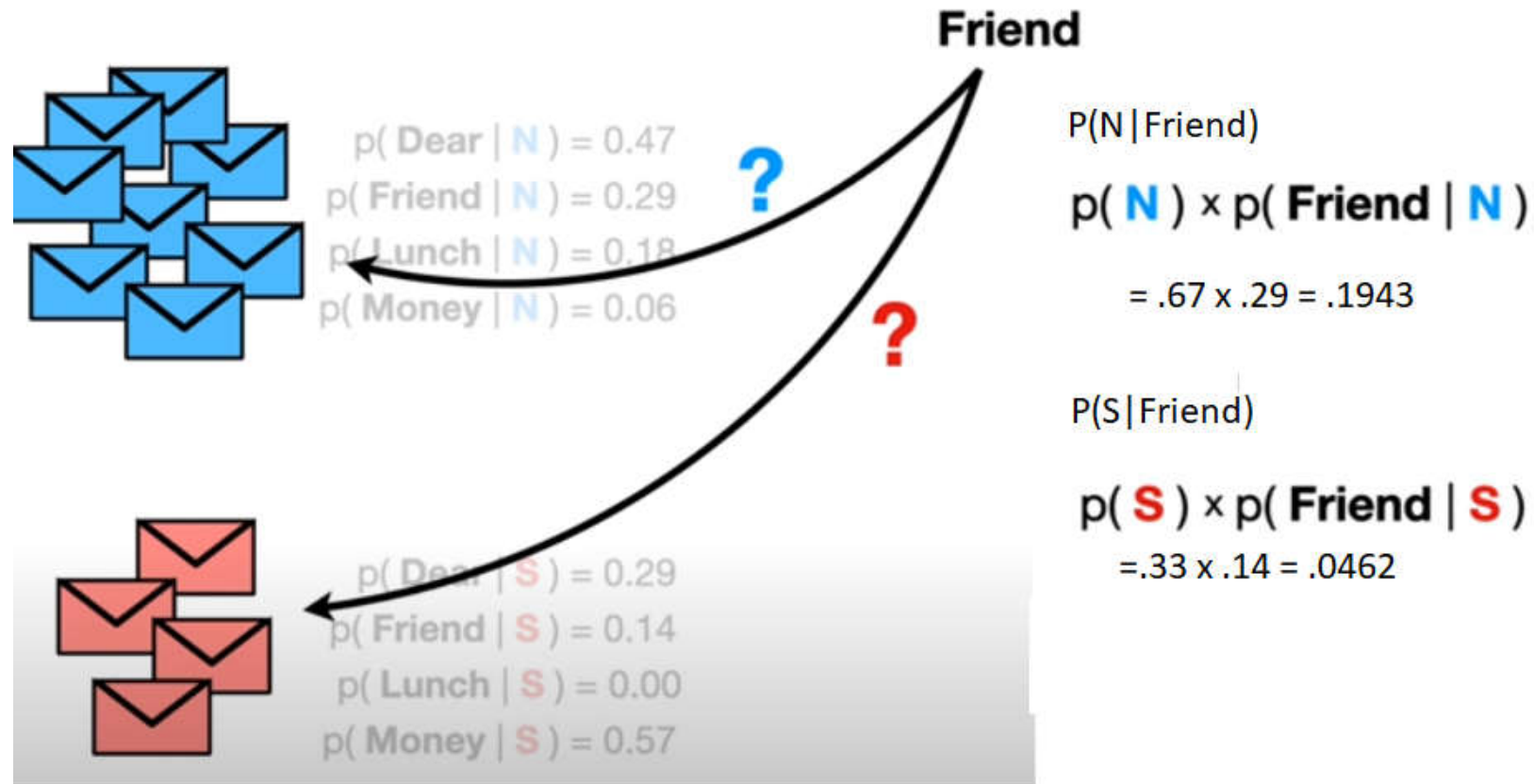
$$\begin{aligned} p(\text{Dear} | \text{Normal}) &= \frac{8}{17} = 0.47 \\ p(\text{Friend} | \text{Normal}) &= \frac{5}{17} = 0.29 \\ p(\text{Lunch} | \text{Normal}) &= \frac{3}{17} = 0.18 \\ p(\text{Money} | \text{N}) &= 0.06 \end{aligned}$$

$$\begin{aligned} p(\text{Dear} | \text{S}) &= 0.29 \\ p(\text{Friend} | \text{S}) &= 0.14 \\ p(\text{Lunch} | \text{S}) &= 0.00 \\ p(\text{Money} | \text{S}) &= 0.57 \end{aligned}$$

Example: Normal / Spam mail Classification

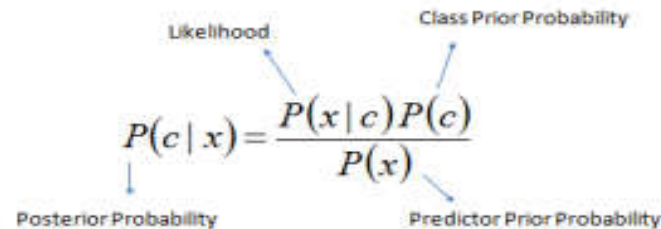


Example: Normal / Spam mail Classification



Naïve Bayesian Classifier

- Suppose, c is a class variable and $X = \{X_1, X_2, \dots, X_n\}$ is a set of attributes, with instance of c .
- Naïve Bayesian classifier calculate this posterior probability using Bayes' theorem



The diagram shows the formula for Bayes' theorem: $P(c | x) = \frac{P(x | c)P(c)}{P(x)}$. Arrows point from labels to parts of the formula: 'Likelihood' points to $P(x | c)$, 'Class Prior Probability' points to $P(c)$, 'Posterior Probability' points to $P(c | x)$, and 'Predictor Prior Probability' points to $P(x)$.

$$P(c | X) = P(x_1 | c) \times P(x_2 | c) \times \dots \times P(x_n | c) \times P(c)$$

- The probability $P(c/X)$ (also called class conditional probability) is therefore proportional to $P(X/c) \cdot P(c)$.
- Thus, $P(c/X)$ can be taken as a measure of c given that X .

$$P(c/X) \approx P(X|c) \cdot P(c)$$

Weather data set

Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

Frequency Table		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

https://www.saedsayad.com/naive_bayesian.htm

Likelihood Table

Frequency Table		Play Golf	
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

$P(x | c) = P(\text{Sunny} | \text{Yes}) = 3 / 9 = 0.33$

Likelihood Table		Play Golf		
Outlook	Sunny	3/9	2/5	5/14
	Overcast	4/9	0/5	4/14
	Rainy	2/9	3/5	5/14
		9/14	5/14	

$P(c) = P(\text{Yes}) = 9 / 14 = 0.64$

$P(x) = P(\text{Sunny}) = 5 / 14 = 0.36$

$$P(c | x) = P(\text{Yes} | \text{Sunny}) = 0.33 \times 0.64 \div 0.36 = 0.60$$

Frequency Table

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3



Likelihood Table

		Play Golf	
		Yes	No
Outlook	Sunny	3/9	2/5
	Overcast	4/9	0/5
	Rainy	2/9	3/5

		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1



		Play Golf	
		Yes	No
Humidity	High	3/9	4/5
	Normal	6/9	1/5

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1



		Play Golf	
		Yes	No
Temp.	Hot	2/9	2/5
	Mild	4/9	2/5
	Cool	3/9	1/5

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3



		Play Golf	
		Yes	No
Windy	False	6/9	2/5
	True	3/9	3/5

Prediction on test data

Outlook	Temp	Humidity	Windy	Play
Rainy	Cool	High	True	?

$$P(Yes | X) = P(Rainy | Yes) \times P(Cool | Yes) \times P(High | Yes) \times P(True | Yes) \times P(Yes)$$

$$P(Yes | X) = 2/9 \times 3/9 \times 3/9 \times 3/9 \times 9/14 = 0.00529$$

$$P(No | X) = P(Rainy | No) \times P(Cool | No) \times P(High | No) \times P(True | No) \times P(No)$$

$$P(No | X) = 3/5 \times 1/5 \times 4/5 \times 3/5 \times 5/14 = 0.02057$$

Naïve Bayesian Classifier Algorithm

Input: Given a set of k mutually exclusive and exhaustive classes $C = \{c_1, c_2, \dots, c_k\}$, which have prior probabilities $P(C_1), P(C_2), \dots, P(C_k)$.

There are n -attribute set $A = \{A_1, A_2, \dots, A_n\}$, which for a given instance have values $A_1 = a_1, A_2 = a_2, \dots, A_n = a_n$

Step: For each $c_i \in C$, calculate the class condition probabilities, $i = 1, 2, \dots, k$

$$p_i = P(C_i) \times \prod_{j=1}^n P(A_j = a_j | C_i)$$

$$p_x = \max\{p_1, p_2, \dots, p_k\}$$

Output: C_x is the classification

Naïve Bayesian Classifier Pros and Cons

Pros

- simple and easy to implement
- doesn't require as much training data
- handles both continuous and discrete data
- fast and can be used to make real-time predictions
- not sensitive to irrelevant features

Cons

- assumption of independent predictors
- zero frequency problem

Approach to overcome zero frequency problem

Classifying a new instance

- ▶ Consider the instance: (Sunny, Cool, High, Strong)
 - ▶ We can estimate each term using the data, for example:
 - ▶ $\Pr(\text{Yes}) = 9/14$
 - ▶ $\Pr(\text{No}) = 5/14$
 - ▶ $\Pr(\text{Outlook} = \text{Sunny} \mid \text{Yes}) = 2/9$
 - ▶ $\Pr(\text{Outlook} = \text{Sunny} \mid \text{No}) = 3/5$
 - ▶ We end up with $\frac{9}{14} \cdot \frac{2}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} = .0053$ for Yes and $\frac{5}{14} \cdot \frac{3}{5} \cdot \frac{1}{5} \cdot \frac{4}{5} \cdot \frac{3}{5} = .0206$ for No.
 - ▶ We thus predict that No is the output

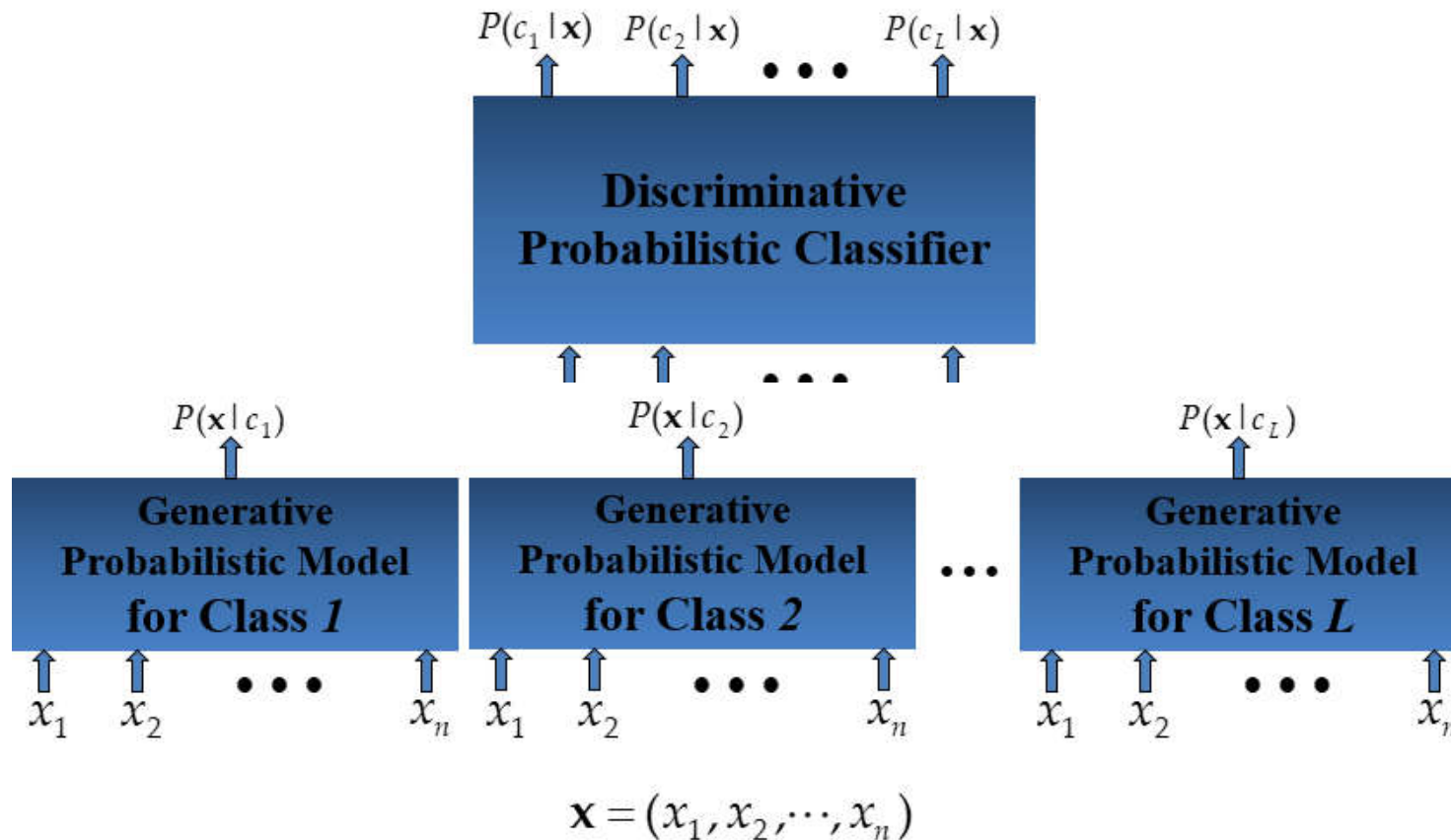
If the posterior probability for one of the attribute is zero, then the overall class-conditional probability for the class vanishes.

This problem can be addressed by using the **M-estimate approach**.

M-Estimate Approach

- ▶ Note that we estimated conditional probabilities $\Pr(A | B)$ by $\frac{n_c}{n}$ where n_c is the number of times $A \wedge B$ happened and n is the number of times B happened in the training data
- ▶ This can cause trouble if $n_c = 0$
- ▶ fix the following numbers p and m
 - ▶ A nonzero prior estimate p for $\Pr(A | B)$, and
 - ▶ A number m that says how confident we are of our prior estimate p , as measured in number of samples
- ▶ Then instead of using $\frac{n_c}{n}$ for the estimate, use $\frac{n_c + mp}{n + m}$

Generative Model



Generative classifier learn joint probability $p(\mathbf{x}, y)$ and use bayes' rule for computing $P(y | \mathbf{x})$
Discriminative models directly learn $p(y | \mathbf{x})$ from data

Summary

- Naive Bayes is a probabilistic supervised learning algorithm
- Based on the independence assumption . Called “Naïve” because of this assumption
- Uses prior knowledge with observed data
- Training and testing is very easy and fast
- Generative classifier model

Reference

Data Mining: Concepts and Techniques, (3rd Edn.), Jiawei Han, Micheline Kamber, Morgan Kaufmann, 2015.

Introduction to Data Mining, Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, Addison-Wesley, 2014

Text Classification using Naïve Bayes

Text Classification Problem

Natural Language Processing (NLP) task

Applications include

- **Social media monitoring:**
 - **Customer feedback:**
 - **Market research:**
-
- *Input:*
 - a document d
 - a fixed set of classes $C = \{c_1, c_2, \dots, c_J\}$
 - *Output:* a predicted class $c \in C$

Bag of words representation

Y(I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet.) = C

👍
👎

I **love** this movie! It's **sweet**, but with **satirical** humor. The dialogue is **great** and the adventure scenes are **fun**... It manages to be **whimsical** and **romantic** while **laughing** at the conventions of the fairy tale genre. I would **recommend** it to just about anyone. I've seen it **several** times, and I'm always **happy** to see it **again** whenever I have a friend who hasn't seen it yet.

```
x love xxxxxxxxxxxxxxxx sweet
xxxxxxxxx satirical xxxxxxxxxxxx
xxxxxxxxxxxxx great xxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxx fun xxxx
xxxxxxxxxxxxxxxxx whimsical xxxx
romantic xxxx laughing
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxx recommend xxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xx several xxxxxxxxxxxxxxxxxxxx
xxxxx happy xxxxxxxx again
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

Y(

great	2
love	2
recommend	1
laugh	1
happy	1
...	...

) = C

👍
👎

A bag-of-words model, is a way of extracting features from text to describe vocabulary of known words

To reduce vocabulary size, apply text cleaning techniques

Naïve Bayes Classifier

- For a document d and a class c

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

$$c_{MAP} = \operatorname{argmax}_{c \in C} P(c|d)$$

MAP is "maximum a posteriori" = most likely class

$$= \operatorname{argmax}_{c \in C} \frac{P(d|c)P(c)}{P(d)}$$

Bayes Rule

$$= \operatorname{argmax}_{c \in C} P(d|c)P(c)$$

Dropping the denominator

$$= \operatorname{argmax}_{c \in C} P(x_1, x_2, \dots, x_n | c)P(c)$$

Document d represented as features $x_1 \dots x_n$

Assumptions

- **Bag of Words assumption:** Assume position doesn't matter
- **Conditional Independence:** Assume the feature probabilities $P(x_i | c_j)$ are independent given the class c .

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

$$c_{NB} = \operatorname{argmax}_{c \in C} P(c_j) \prod_{x \in X} P(x | c)$$

Naïve Bayes Learning

- From training corpus, extract *Vocabulary*
- Calculate $P(c_j)$ terms
 - For each c_j in C do
 - $docs_j \leftarrow$ all docs with class $= c_j$
 - $$P(c_j) \leftarrow \frac{|docs_j|}{|\text{total \# documents}|}$$
- Calculate $P(w_k | c_j)$ terms
 - $Text_j \leftarrow$ single doc containing all $docs_j$
 - For each word w_k in *Vocabulary*
 - $n_k \leftarrow$ # of occurrences of w_k in $Text_j$
 - $$P(w_k | c_j) \leftarrow \frac{n_k + \alpha}{n + \alpha |Vocabulary|}$$

$$\hat{P}(c) = \frac{N_c}{N}$$

$$\hat{P}(w|c) = \frac{\text{count}(w,c)+1}{\text{count}(c)+|V|}$$

	Doc	Words	Class
Training	1	Chinese Beijing Chinese	c
	2	Chinese Chinese Shanghai	c
	3	Chinese Macao	c
	4	Tokyo Japan Chinese	j
Test	5	Chinese Chinese Chinese Tokyo Japan	?

Priors:

$$P(c) = \frac{3}{4}$$

$$P(j) = \frac{1}{4}$$

Choosing a class:

$$P(c|d5) \propto \frac{3}{4} * \left(\frac{3}{7}\right)^3 * \frac{1}{14} * \frac{1}{14} \approx 0.0003$$

Conditional Probabilities:

$$P(\text{Chinese}|c) = \frac{(5+1)}{(8+6)} = \frac{6}{14} = \frac{3}{7}$$

$$P(\text{Tokyo}|c) = \frac{(0+1)}{(8+6)} = \frac{1}{14}$$

$$P(\text{Japan}|c) = \frac{(0+1)}{(8+6)} = \frac{1}{14}$$

$$P(\text{Chinese}|j) = \frac{(1+1)}{(3+6)} = \frac{2}{9}$$

$$P(\text{Tokyo}|j) = \frac{(1+1)}{(3+6)} = \frac{2}{9}$$

$$P(\text{Japan}|j) = \frac{(1+1)}{(3+6)} = \frac{2}{9}$$

$$P(j|d5) \propto \frac{1}{4} * \left(\frac{2}{9}\right)^3 * \frac{2}{9} * \frac{2}{9} \approx 0.0001$$

Summary

- Uses Bag of Words representation
- probability of the class of document is computed using Bayes theorem
- Select a class with highest posterior probability

References

<https://web.stanford.edu/class/cs124/lec/naivebayes.pdf>

<https://nlp.stanford.edu/IR-book/html/htmledition/naive-bayes-text-classification-1.html>