Bayes' Classifier

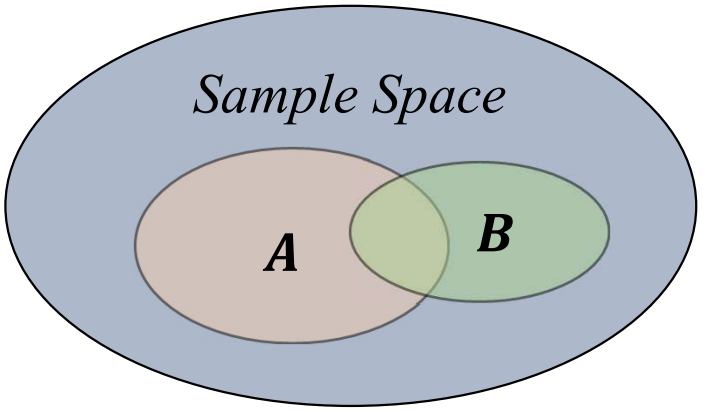
Sourav Karmakar

souravkarmakar29@gmail.com

Revisiting Bayes' Rule

Bayes' theorem is simply a consequence of conditional probabilities of two events A and B:

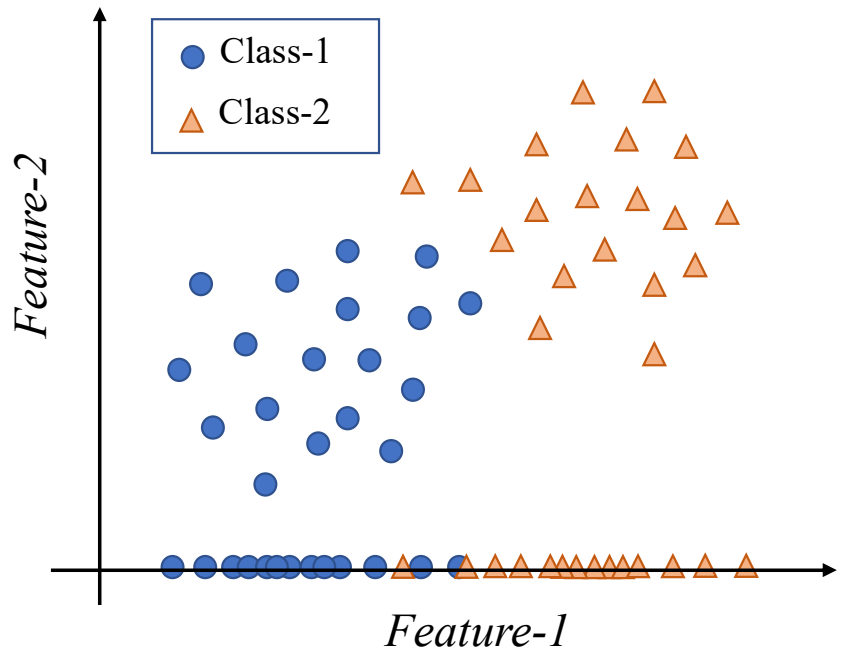
$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$



- $P(A)$: Prior Probability of event A
- $P(B|A)$: Likelihood of event B given event A
- $P(B)$: Evidence of event B
- $P(A|B)$: Posterior probability of event A given event B

Bayesian Classification

- The classification problem is posed in probabilistic terms.
- Create models for the distribution of objects of different classes.
- Probabilistic framework is used to make classification decisions.

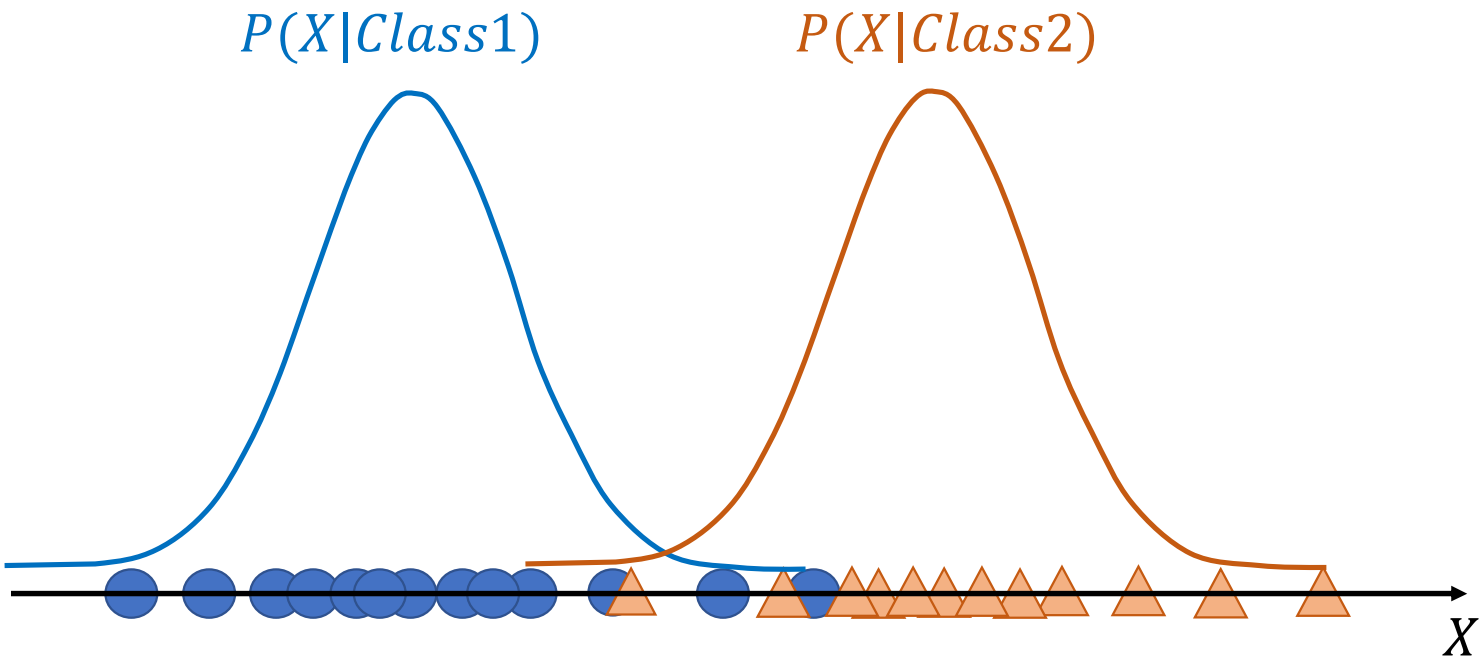


- Each object is usually associated with multiple features / predictors.
- We will look at the case of just one feature for now.
- We are now going to define two key concepts.
  - Likelihood or Class conditional probability distribution.
  - Prior probability of a class.

Bayesian Classification

- **Likelihood or Class Conditional Probability Distribution**

We model the likelihood function $P(X|Class\boldsymbol{k})$ using a probability distribution function.



In this case we have ***modelled*** these distributions using Gaussian / Normal distribution

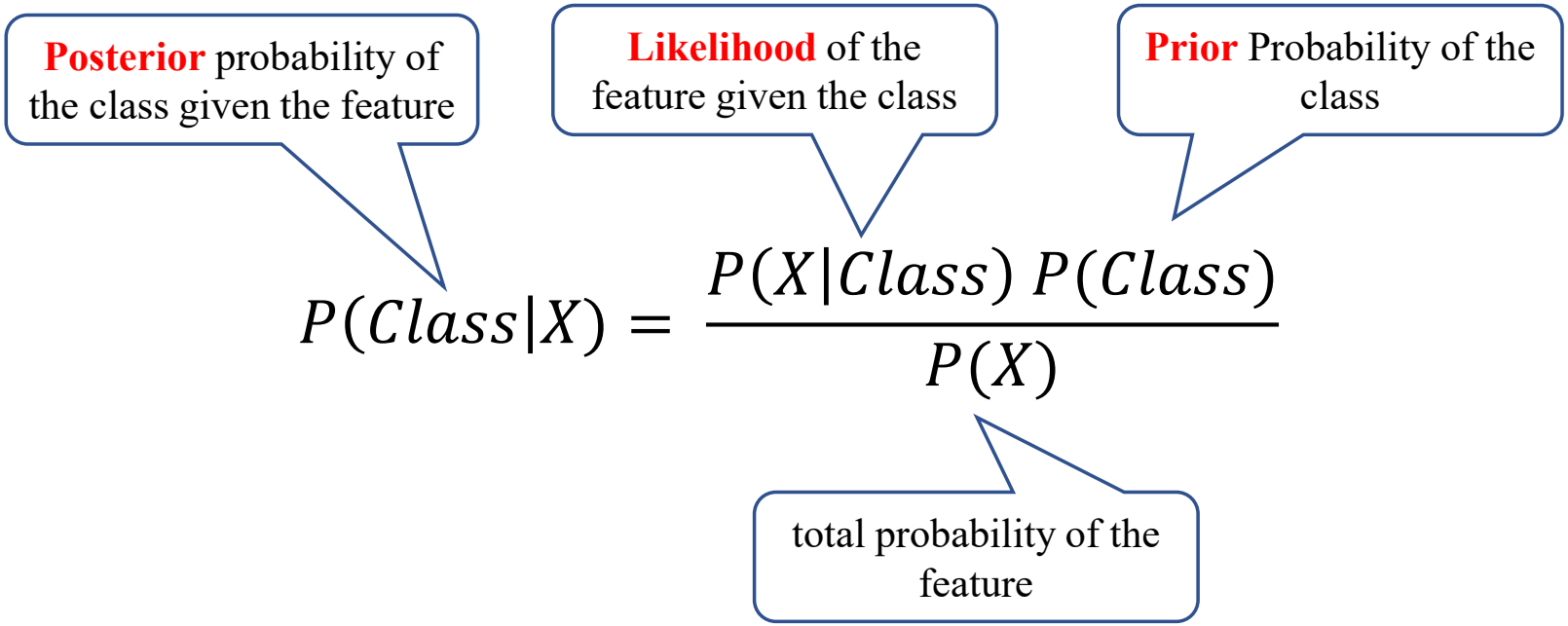
Bayesian Classification

- **Prior probabilities of classes**
- We model *prior probabilities* to quantify the expected *a priori* chance of seeing a class.
- Let there are total m many training samples and out of which m_1 number of samples belong to Class-1 and $m - m_1 = m_2$ number of samples belong to Class-2
- Then prior probabilities are calculated as: $P(Class1) = \frac{m_1}{m}$ and $P(Class2) = \frac{m_2}{m}$
- Now we have *prior probabilities* of each class: $P(Class1), P(Class2)$ and we also have models for the *likelihood functions* given each class. i.e. $P(X|Class1)$ and $P(X|Class2)$. Usually the likelihood function is modelled using some standard probability distribution function such as gaussian distribution.
- We want the ***probability of the class given a pattern X***. i.e. $P(Class1|X)$ *or* $P(Class2|X)$

How do we get $P(Class|X)$ knowing $P(X|Class)$ and $P(Class)$*?*

Bayesian Classification

- We apply Bayes' rule to obtain $P(Class|X)$:



Bayes' Decision Rule

- If we observe an object with feature X, how do we decide if the object is from Class-1?
- Bayes' decision rule is simply choose Class-1 if:

$$P(Class1|X) > P(Class2|X)$$

$$or, \quad \frac{P(X|Class1)\,P(Class1)}{P(X)} > \frac{P(X|Class2)\,P(Class2)}{P(X)}$$

$$or, \quad P(X|Class1)\,P(Class1) > P(X|Class2)\,P(Class2)$$

$$or, \quad \frac{P(X|Class1)\,P(Class1)}{P(X|Class2)\,P(Class2)} > 1$$

$$or, \quad \log\left(\frac{P(X|Class1)\,P(Class1)}{P(X|Class2)\,P(Class2)}\right) > 0$$

$$G(X)$$

If $G(X) > 0$, we classify as Class-1, called MAP rule.

MAP: Maximum A Posteriori

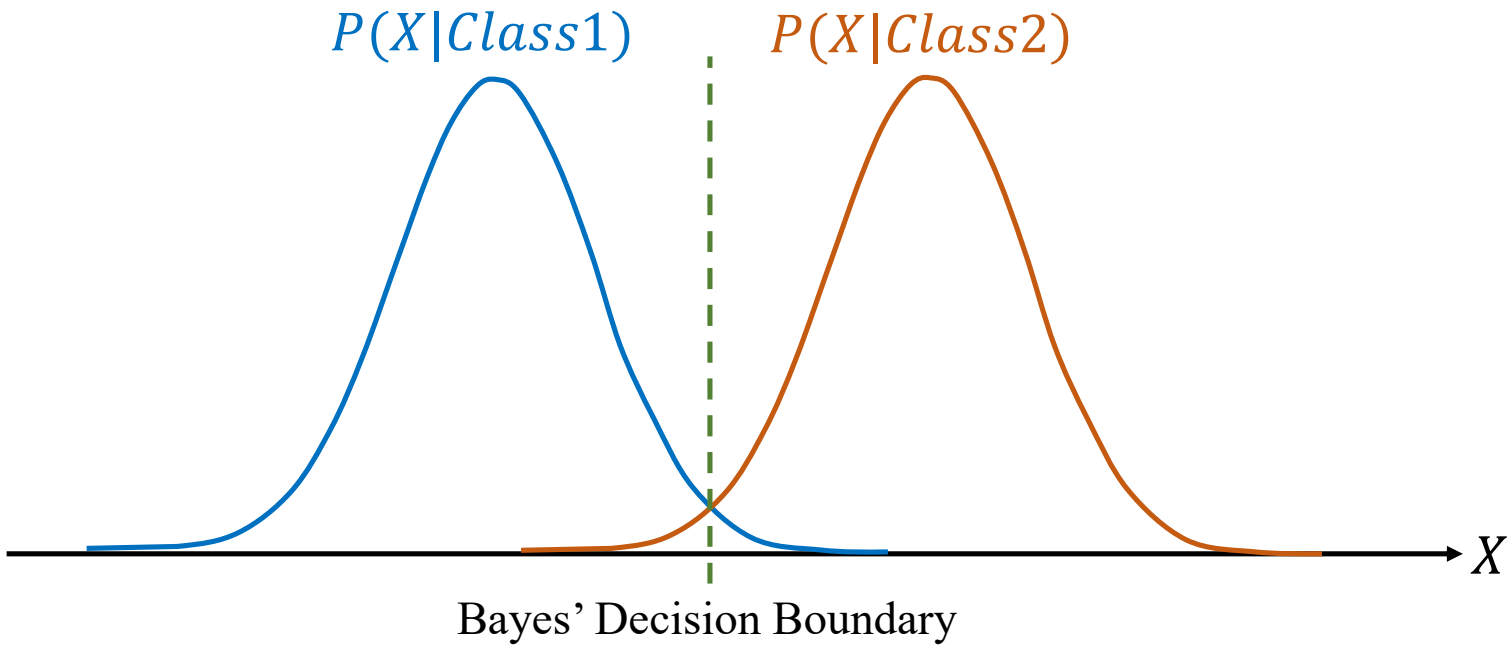
Bayes' Decision Boundary

- Bayes' decision boundary is obtained as:

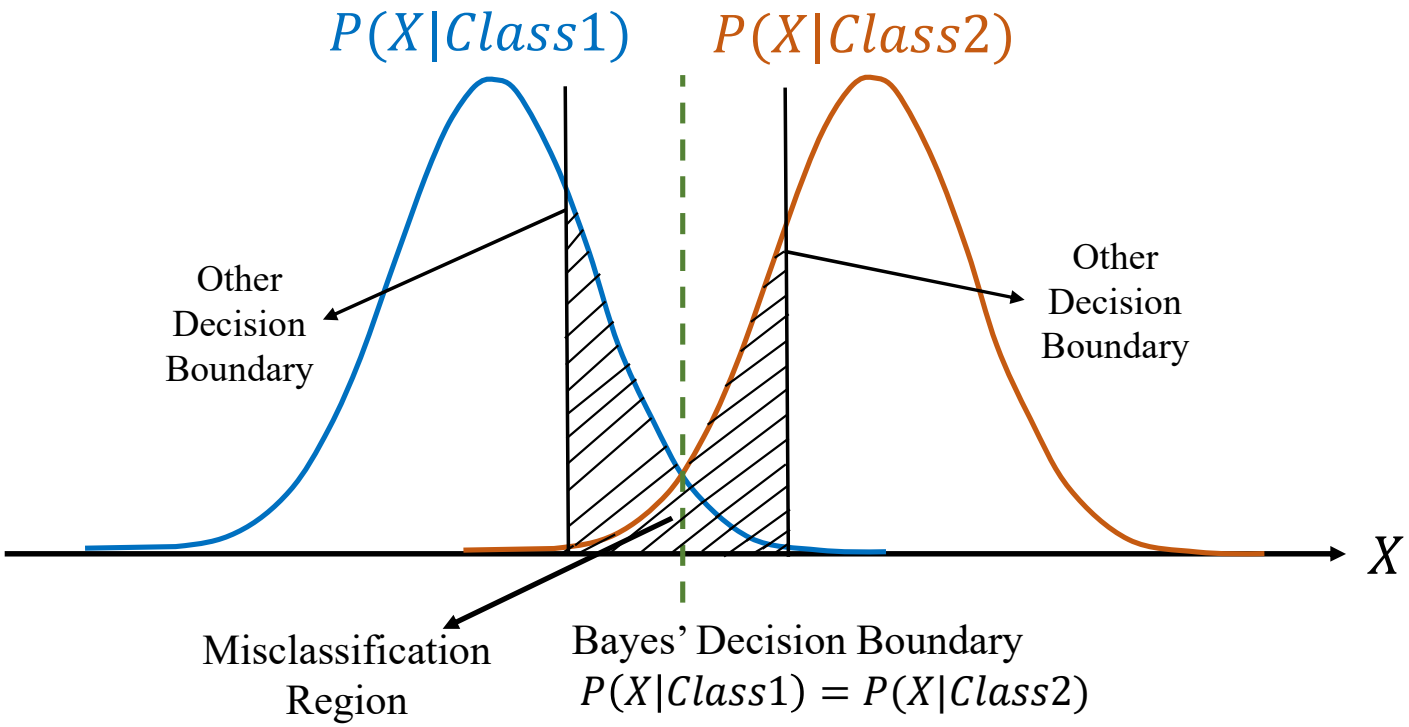
$$G(X) = 0 \;\Rightarrow\; P(X|Class1)P(Class1) = P(X|Class2)P(Class2)$$

- If we assume that prior probabilities of the classes are identical (balanced distr.) then:

$$P(X|Class1) = P(X|Class2)$$



Bayes' Decision Boundary



- Changing the decision boundary increases the misclassification error

- Bayes' decision boundary gives minimum misclassification error

For Multiple Features

- We have feature vector comprises of n features: $\vec{X} = [X_1, X_2, \dots, X_n]^T$
- Bayes' Classification: $P(C|\vec{X}) \propto P(\vec{X}|C)\, P(C)$
- Now: $P(\vec{X}|C) = P(X_1, X_2, \dots, X_n|C)$
- Difficulty: Learning the joint conditional probability $P(X_1, X_2, \dots, X_n|C)$

Naïve Bayes' Classification:

- Assumption that all input features are conditionally independent:

$$P(X_1, X_2, \dots, X_n|C) = P(X_1|C)\, P(X_2|C) \dots P(X_n|C) = \prod_{i=1}^{n} P(X_i|C)$$

- Maximum A Posteriori (MAP) rule: for $\vec{X} = [X_1, X_2, \dots, X_n]^T$ it belongs to class C_1 if:

$$[P(X_1|C_1)\, P(X_2|C_1) \dots P(X_n|C_1)]P(C_1) > [P(X_1|C_2)\, P(X_2|C_2) \dots P(X_n|C_2)]P(C_2)$$

Naïve Bayes' Classification

Advantages:

- Training is very fast; just require to consider each attribute in each class separately to model the likelihood function using some standard distribution (like Gaussian) and prior probabilities of each class.
- Test is straightforward: Calculate the likelihood function of each feature and multiply them with class a priori probabilities and then compare using MAP rule.
- Performance competitive to most of the state-of-the-art classifiers for simple use-cases.
- Many successful applications. E.g. Spam mail filtering.

Naïve Bayes' Classification

Relevant Issues:

- Violation of Independence assumption:
  - For many real world tasks, $P(X_1, X_2, \ldots, X_n|C) \neq P(X_1|C)\, P(X_2|C) \ldots P(X_n|C)$
  - Nevertheless, Naïve Bayes' works surprisingly well even when independence assumption is violated.

- Zero Conditional Probability Problem in case of non-continuous features:
  - If no example contains the attribute value $X_j = a_{jk}$, then $\hat{P}\left(X_j = a_{jk} \mid C = C_i\right) = 0$
  - In this circumstance, $\hat{P}(X_1|C_i) \ldots \hat{P}\left(a_{jk}|C_i\right) \ldots \hat{P}(X_n|C_i) = 0$ during test.
  - For a remedy, conditional probability is calculated with a smoothing technique called *Laplace Smoothing*.

Modelling the Likelihood

For continuous features we model $P(X|C)$ using Gaussian / Normal distribution

$y \in \{C_1, C_2\}$

x_1	x_2	.....	x_n	y
				C_1
				C_2

$x_1 \rightarrow$ Class-1: [values] $\rightarrow \mu_1^{(C_1)}, \sigma_1^{(C_1)}$

$x_1 \rightarrow$ Class-2: [values] $\rightarrow \mu_1^{(C_2)}, \sigma_1^{(C_2)}$

PDF of Gaussian distribution?

$$p(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

$$P(x_1|C_1) = \frac{1}{\sqrt{2\pi}\,\sigma_1^{(C_1)}} e^{-\frac{1}{2}\left(\frac{x_1-\mu_1^{(C_1)}}{\sigma_1}\right)^2}, \quad P(x_1|C_2) = \frac{1}{\sqrt{2\pi}\,\sigma_1^{(C_2)}} \cdot e^{-\frac{1}{2}\left(\frac{x_1-\mu_1^{(C_2)}}{\sigma_1^{(C_2)}}\right)^2}$$

Test sample:- $X_{test} = [x_1^{(t)}, x_2^{(t)}, x_3^{(t)}, \ldots, x_n^{(t)}]$

$P(C_1|X_{test})$ & $P(C_2|X_{test})$

means

	C_1	C_2
X_1	$\mu_1^{(C_1)}$	$\mu_1^{(C_2)}$
X_2	$\mu_2^{(C_1)}$	$\mu_2^{(C_2)}$
X_3	$\mu_3^{(C_1)}$	$\mu_3^{(C_2)}$
$\vdots$	$\vdots$	$\vdots$
X_n	$\mu_n^{(C_1)}$	$\mu_n^{(C_2)}$

standard-dev

	C_1	C_2
X_1	$\sigma_1^{(C_1)}$	$\sigma_1^{(C_2)}$
X_2	$\sigma_2^{(C_1)}$	$\sigma_2^{(C_2)}$
X_3	$\sigma_3^{(C_1)}$	$\sigma_3^{(C_2)}$
$\vdots$	$\vdots$	$\vdots$
X_n	$\sigma_n^{(C_1)}$	$\sigma_n^{(C_2)}$

$$P(C_1 \mid X_{test}) = \left[P(x_1^{(t)} \mid C_1) \cdot P(x_2^{(t)} \mid C_1) \cdots P(x_n^{(t)} \mid C_1) \right] \cdot P(C_1)$$

$$P(C_2 \mid X_{test}) = \left[P(x_1^{(t)} \mid C_2) \, P(x_2^{(t)} \mid C_2) \cdots P(x_n^{(t)} \mid C_2) \right] P(C_2)$$

$$\prod_{i=1}^{n} P(x_i^{(t)} \mid C_1) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\,\sigma_i^{(C_1)}} \cdot e^{-\frac{1}{2}\left[\frac{x_i^{(t)} - \mu_i^{(C_1)}}{\sigma_i^{(C_1)}}\right]^2} = P(X_{test} \mid C_1)$$

$$\Rightarrow \log_e\left(P(X_{test} \mid C_1)\right) = \sum_{i=1}^{n} \log_e \left[\frac{1}{\sqrt{2\pi}\,\sigma_i^{(C_1)}} \times e^{-\frac{1}{2}\left[\frac{x_i^{(t)} - \mu_i^{(C_1)}}{\sigma_i^{(C_1)}}\right]^2} \right]$$

$\log_e(\cdot)$: log-likelihood

$$= \sum_{i=1}^{n} \left[\log_e \frac{1}{\sqrt{2\pi}\,\sigma_i^{(C_1)}} - \frac{1}{2}\left[\frac{x_i^{(t)} - \mu_i^{(C_1)}}{\sigma_i^{(C_1)}}\right]^2 \right]$$

$$\log_e P(X_{test} \mid C_1) = -\sum_{i=1}^{n} \left[\log \sqrt{2\pi} \cdot \sigma_i^{(C_1)} + \frac{1}{2} \cdot \left(\frac{x_i^{(t)} - \mu_1^{(C_1)}}{\sigma_i^{(C_1)}} \right)^2 \right]$$

$$= -\sum_{i=1}^{n} \left[\log \sqrt{2\pi} + \log \sigma_i^{(C_1)} + \frac{1}{2} \cdot \left(\frac{x_i^{(t)} - \mu_1^{(C_1)}}{\sigma_i^{(C_1)}} \right)^2 \right]$$

$P(C_1 \mid X_{test})$ & $P(C_2 \mid X_{test})$

$\downarrow \propto$

$P(X_{test} \mid C_1) \cdot P(C_1)$

$\propto$

$P(X_{test} \mid C_2) \cdot P(C_2)$

$\Rightarrow \log_e (P(X_{test} \mid C_1)) + \log_e (P(C_1))$

log of likelihood + log of prior

$P(C_1 \mid X_{test}) > P(C_2 \mid X_{test})$

$\log_e (P(C_1 \mid X_{test})) > \log_e P(C_2 \mid X_{test})$

Gaussian Naive Bayes

Categorical Naive Bayes : Categorical features

$x \in \{ cat\_1, cat\_2, cat\_3, \cdots, cat\_k \}$

$y \in \{ c_1, c_2 \}$

x	y
⋮	⋮

$$P(x = cat\text{-}1 \mid y = c_1) = \frac{N_{x=cat-1 \ \& \ y=c_1}}{N_{y=c_1}}$$

$$P(x = cat\text{-}i \mid y = c_j) = \frac{N_{x=cat-i \ \& \ y=c_j}}{N_{y=c_j}} = \frac{(N_{x=cat-i \ \& \ y=c_j}) + \alpha}{N_{y=c_j} + k\alpha}$$

$\alpha \rightarrow$ Smoothing parameter

$\alpha \geq 0$ $\quad \alpha = 0$: no smoothing , $\underline{\alpha = 1}$: Laplace smoothing

$0 < \alpha < 1$: Lidstone Smoothing

For Categorical Naive-Bayes the features are ordinal encoded.

cat-1 : 0 , cat-2 : 1 , cat-3 : 2 , ... , cat-k : k-1

Bernoulli Naive Bayes : Binary / Boolean features

x y $\quad x \in \{0, 1\}$, $y \in \{c_1, c_2\}$

$$P(x=0 \mid y=c_1) = \frac{N_{x=0 \,\&\, y=c_1} + \alpha}{N_{y=c_1} + 2\alpha}$$

$$P(x=1 \mid y=c_j) = \frac{N_{x=1 \,\&\, y=c_j} + \alpha}{N_{y=c_j} + 2\alpha}$$

Bernoulli Naive Bayes is subset of categorical naive bayes where the feature can take only two values.

For Bernoulli naive bayes the features must be binarized & to be encoded in 0 & 1.

Email Spam / Non-spam classification Using Naive Bayes

Spam = 10 — Spam : 1

non-spam = 40 — non-spam : 0

From all the emails find out the unique words (tokens).

Vocabulary (V) = { Set of all the unique words / tokens in a corpus }

↳ set of documents

$|V|$ → cardinality of vocabulary set / total number of unique words.

$V = \{ \underset{w_1}{\text{'dear'}}, \underset{w_2}{\text{'friend'}}, \underset{w_3}{\text{'money'}}, \text{'prize'}, \ldots, \underset{w_V}{\text{'plan'}} \}$

		w_1	w_2	w_3	.....	w_V	
(1)	Spam	n_{11}	n_{21}	n_{31}		n_{V1}	n_{spam}
(0)	non-spam	n_{10}	n_{20}	n_{30}		n_{V0}	$n_{non\text{-}spam}$

count-vectorize

$$p(w_i|spam) = \frac{n_{i1} + \alpha}{n_{spam} + \alpha|V|}$$

α : smoothing parameter

New email (E_1): "dear money prize" | "money dear prize"

$P(\text{spam} \mid E_1)$ vs $P(\text{non-spam} \mid E_1)$

$\propto$ $\propto$

$P(E_1 \mid \text{spam}) \cdot P(\text{spam})$: $P(E_1 \mid \text{non-spam}) \cdot P(\text{non-spam})$

$= P(\text{dear} \mid \text{spam}) \times p(\text{'money'} \mid \text{spam})$: $= P(\text{dear} \mid \text{non-spam}) \times p(\text{money} \mid \text{non-spam})$

$\times p(\text{prize} \mid \text{spam}) \times p(\text{spam})$: $\times p(\text{prize} \mid \text{non-spam}) \times p(\text{non-spam})$

$= p_1$ $= p_2$

Multinomial Naive Bayes : (Spam filtering)

$p_1 > p_2 \rightarrow$ spam $\quad p_1 < p_2 \rightarrow$ non-spam

$\log(p_1) > \log(p_2)$: log-transformation to convert multiplication to addition & to avoid underflow problem due to multiplication of small numbers.

Thank You