BAYES’ CLASSIFIER

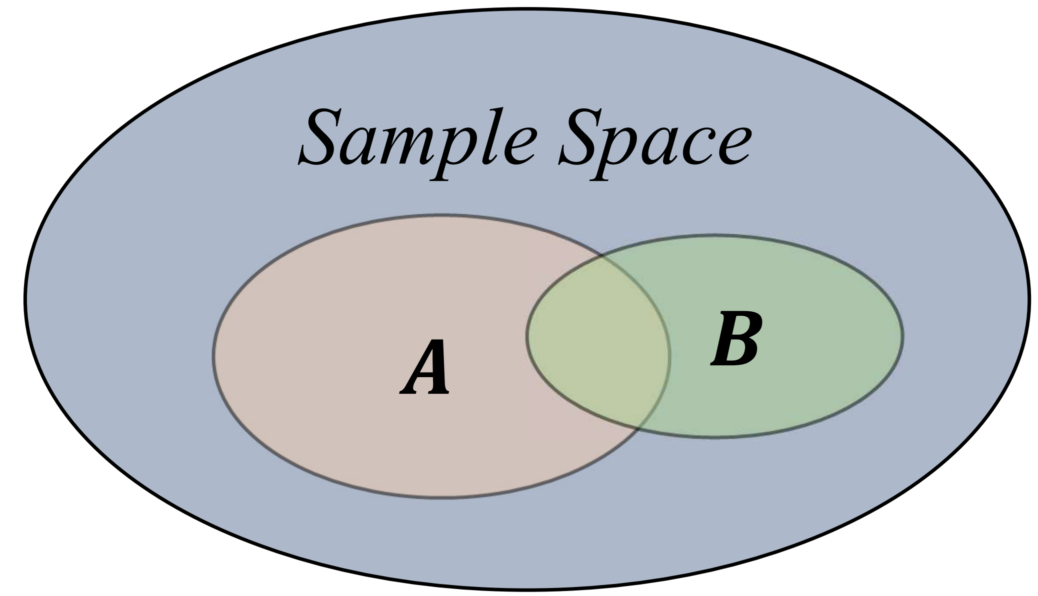
Sourav Karmakar

souravkarmakar29@gmail.com

REVISITING BAYES' RULE

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

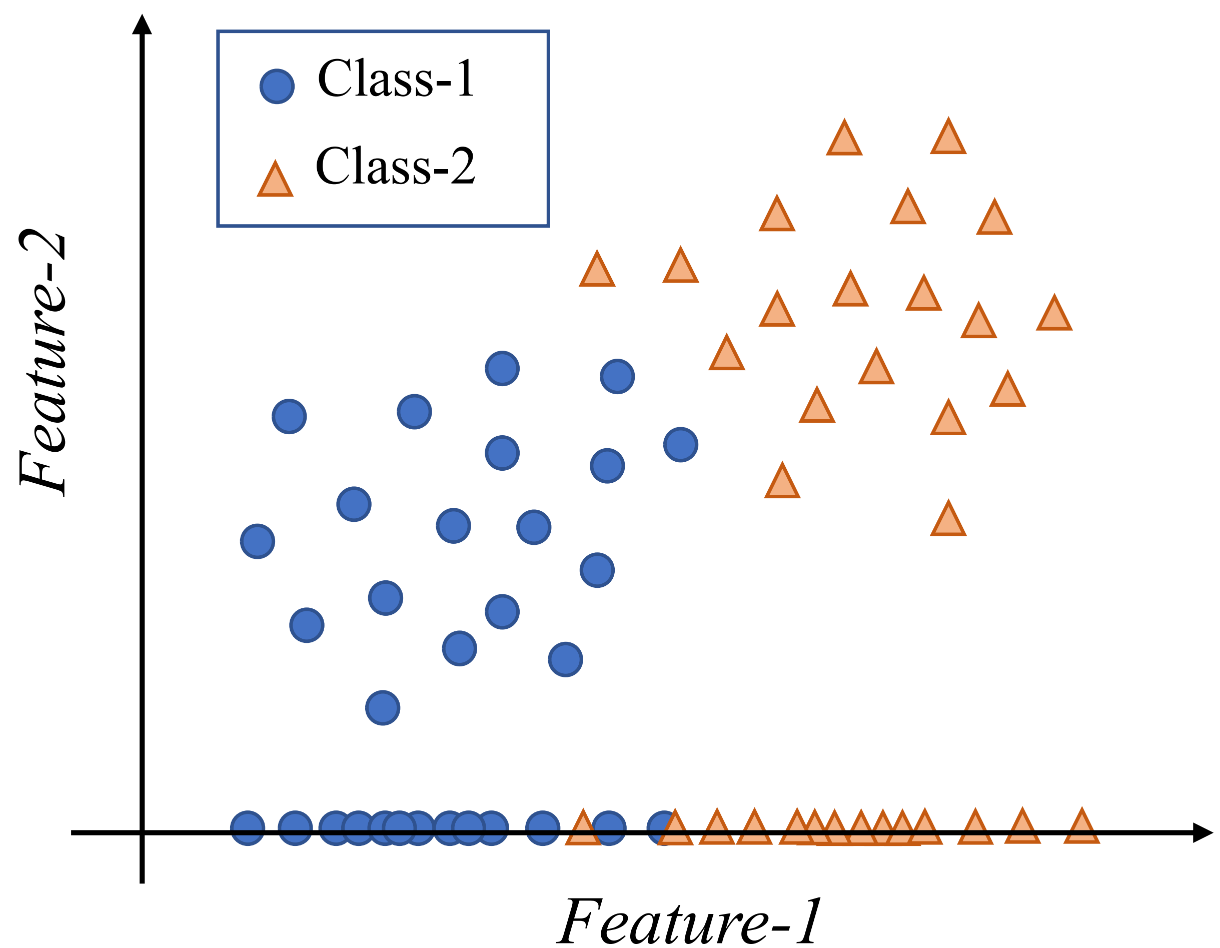
$$\boxed{P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}}$$



- $P(A)$: 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.

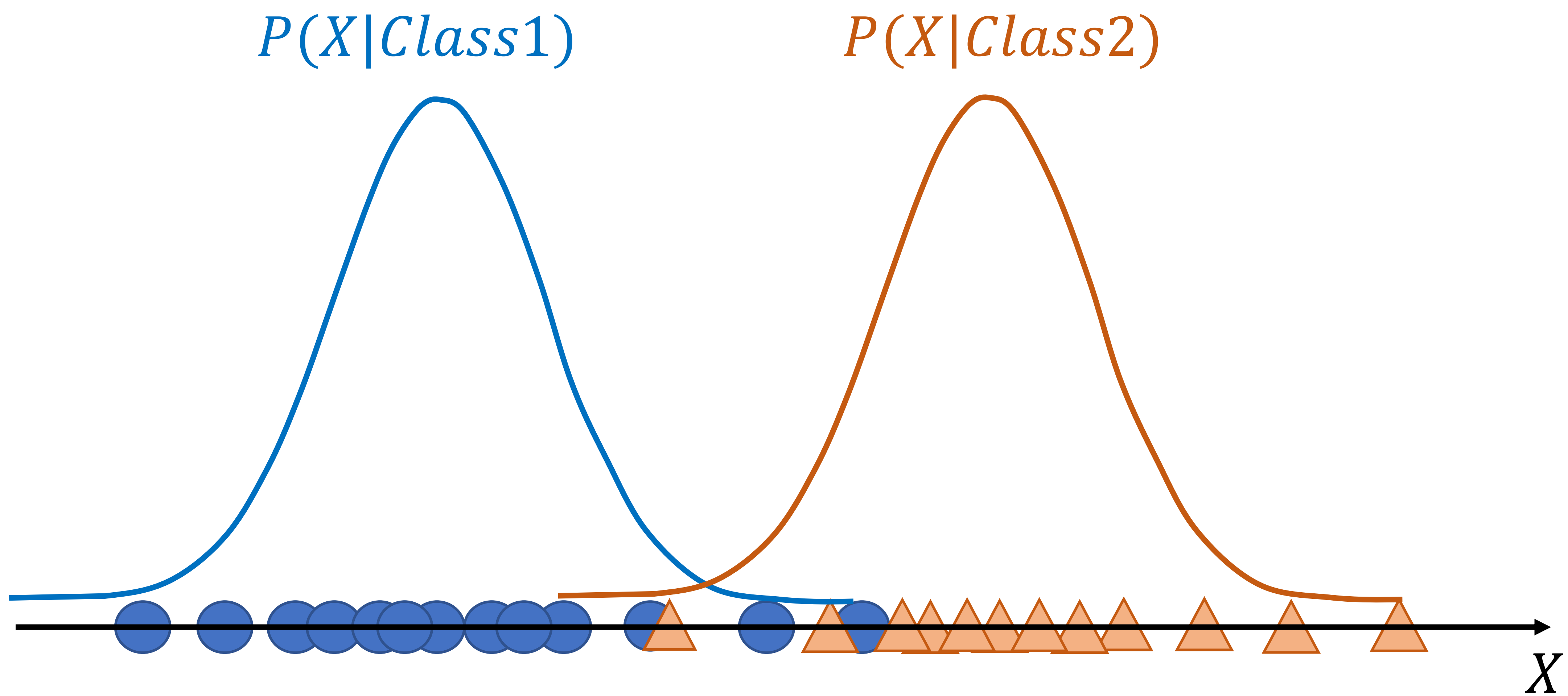


- Each object is usually associated with multiple features / predictors.
- We will look at the case of just one feature for now.
- We are now going to define two key concepts.

BAYESIAN CLASSIFICATION

- **Class Conditional Probability Distribution**

Patterns for each class is drawn from Class Conditional Probability Distribution (CCPD)



Our first goal will be to ***model*** these distributions

BAYESIAN CLASSIFICATION

- 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 priors defining *a priori* probability of a class: $P(Class1), P(Class2)$ and we also have models for the probability pattern given each class. i.e. $P(X|Class1)$ and $P(X|Class2)$. Usually this 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

$P(Class1|X) > P(Class2|X)$
then X belongs to class1

$P(Class2|X) > P(Class1|X)$
then X belongs to class2

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

Posterior or Belief after evidence

Likelihood of the evidence

Prior or Belief before evidence

$$P(Class|X) = \frac{P(X|Class)\,P(Class)}{P(X)}$$

Evidence

$$P(Class1|X) = \frac{P(X|Class1)\,P(Class1)}{P(X)}$$

$$P(Class2|X) = \frac{P(X|Class2)\,P(Class2)}{P(X)}$$

BAYES' DECISION RULE

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

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

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

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

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

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

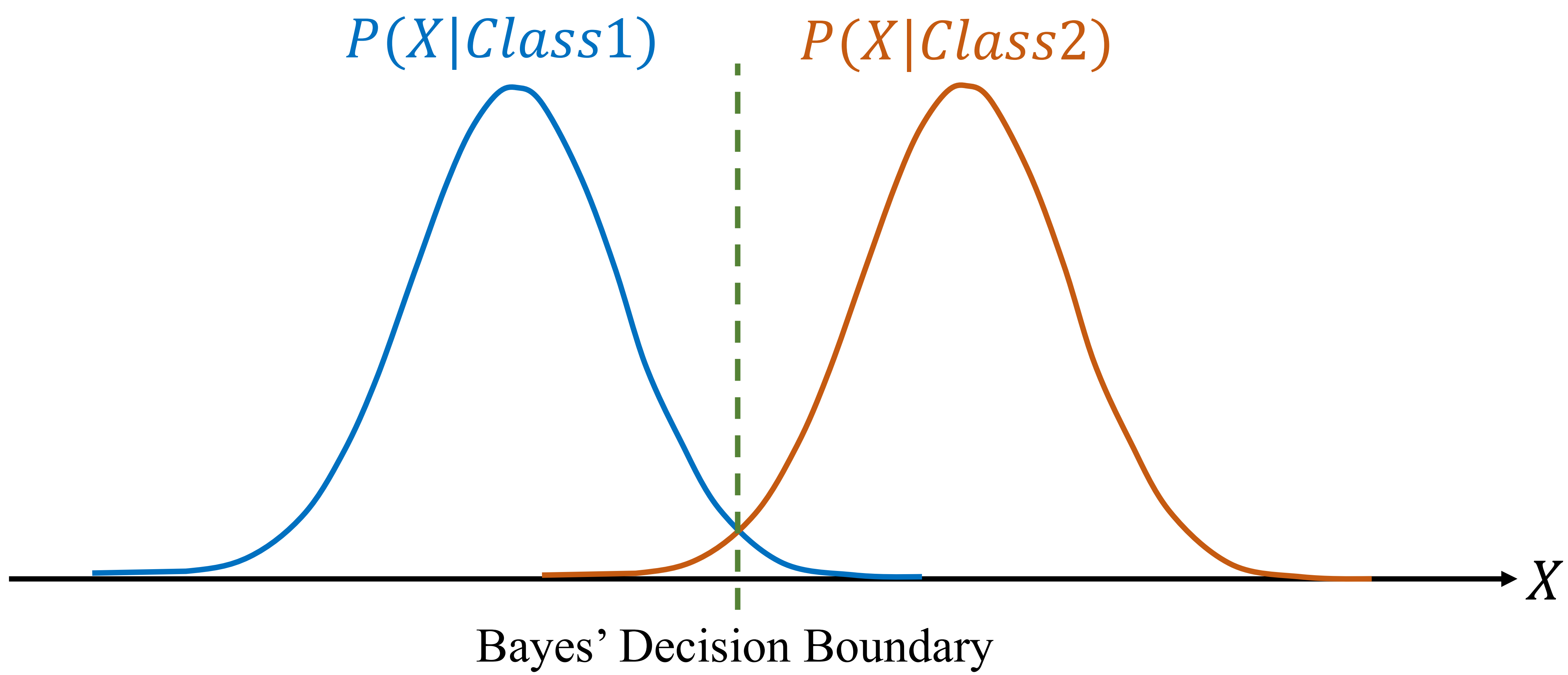
BAYES' DECISION BOUNDARY

- Bayes' decision boundary is obtained as:

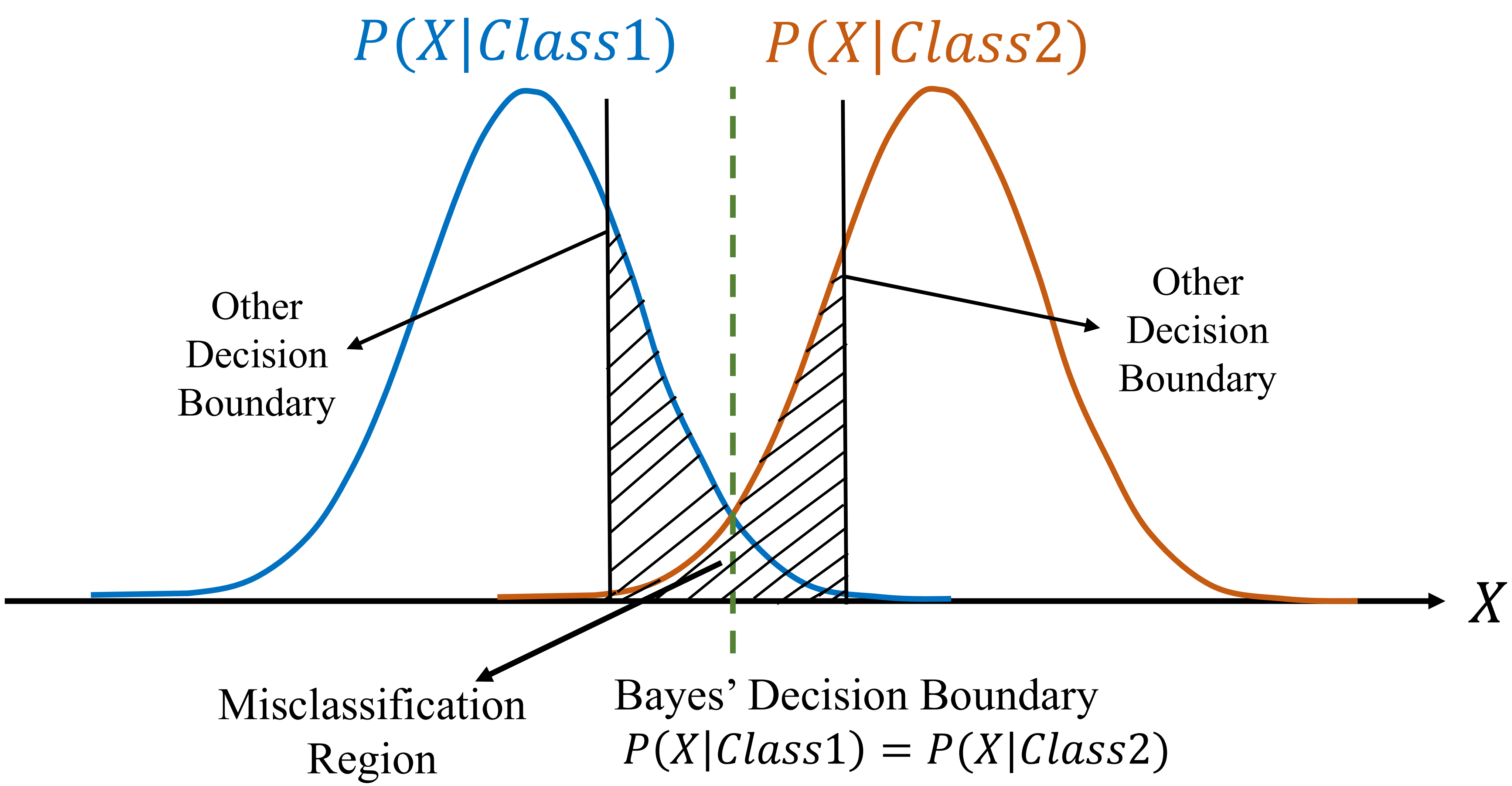
$$G(X) = 0 \;\;\Rightarrow\;\; P(X|Class1)P(Class1) = P(X|Class2)P(Class2)$$

- 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

x_1	x_2	Y
$\vdots$		

x_1 can take 10 different values.

x_2 can take 5 " "

$\underbrace{P(x_1 | c)}$ $\quad$ $P(x_2 | c)$

$x_2 = 10, 20, 30, 40, 50$

$x_1 = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10$

$100 \rightarrow c_1$

$100 \rightarrow c_2$

$P(x_1 = 1 | c_1)$ $\qquad$ $P(x_1 | c_1)$

$P(x_1 = 2 | c_1)$ $\dots$ $P(x_1 | c_2)$

$(x_1, x_2) \rightarrow$ 50 different combinations.

$(x_1, x_2, x_3, \dots, x_n)$

$x_1 \rightarrow k_1$, $x_2 \rightarrow k_2$, $x_3 \rightarrow k_3$, $x_n \rightarrow k_n$

$(k_1 \, k_2 \, k_3 \dots k_n)$

Independence

$P(A \cdot B) = P(A \cap B)$ If A & B are independent then —

$$P(AB) = P(A) \cdot P(B)$$

$P(ABC) = P(A) \cdot P(B) \cdot P(C)$ Iff A, B, C are independent.

Conditional Independence :-

$P(A, B \mid D)$ If A & B are conditionally independent

$= P(A \mid D) \cdot P(B \mid D)$

$$P(A, B, C \mid D) = P(A \mid D) \cdot P(B \mid D) \cdot P(C \mid D)$$

FOR MULTIPLE FEATURES

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

$$P(X_1, X_2, \dots, X_n|C) = P(X_1|C)\, P(X_2|C) \dots P(X_n|C)$$

- 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.
- Test is straightforward; just looking up tables or calculating conditional probabilities with normal distributions.
- Performance competitive to most of the state-of-the-art classifiers.
- Many successful applications. E.g. Spam mail filtering.

NAÏVE BAYES' CLASSIFICATION

Relevant Issues:

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

- Zero Conditional Probability Problem:
  - 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, $\hat{P}(X_1|C_i) \dots \hat{P}\left(a_{jk}|C_i\right) \dots \hat{P}(X_n|C_i) = 0$ during test.
  - For a remedy, conditional probability is calculated with the following formula

$$\hat{P}\left(X_j = a_{jk}|C = C_i\right) = \frac{n_c + mp}{n + m}$$

    - n_c: Number of training examples for which $X_j = a_{jk}$ and $C = C_i$
    - n: Number of training examples for which $C = C_i$
    - p: prior estimates (usually, $p = 1/t$ for t possible values of X_j)
    - m: weight to prior (number of *"virtual"* examples, $m \geq 1$)

Thank You