Pattern Classification

03. Pattern Classification Methods

AbdElMoniem Bayoumi, PhD

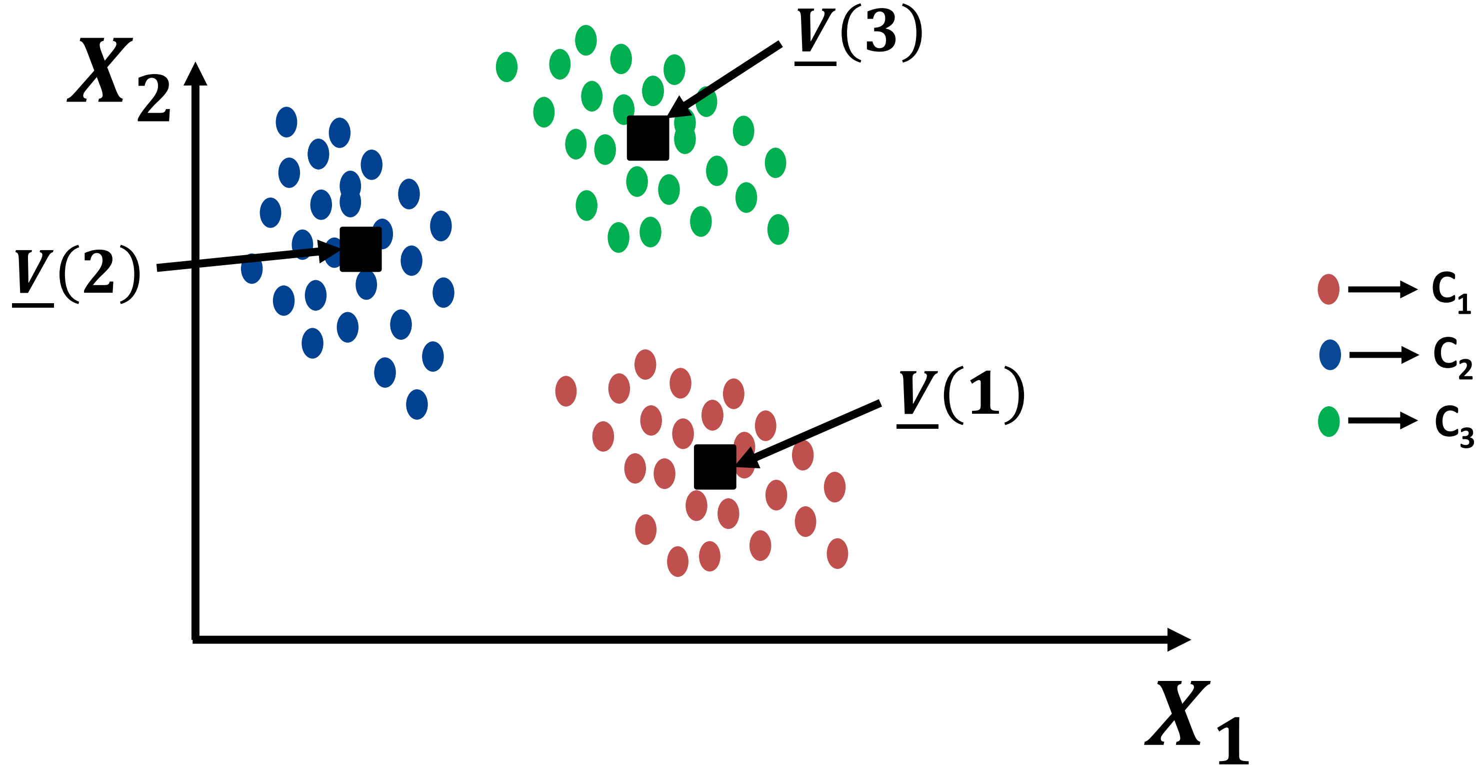
Fall 2021

Acknowledgment

- These slides have been created relying on lecture notes of Prof. Dr. Amir Atiya

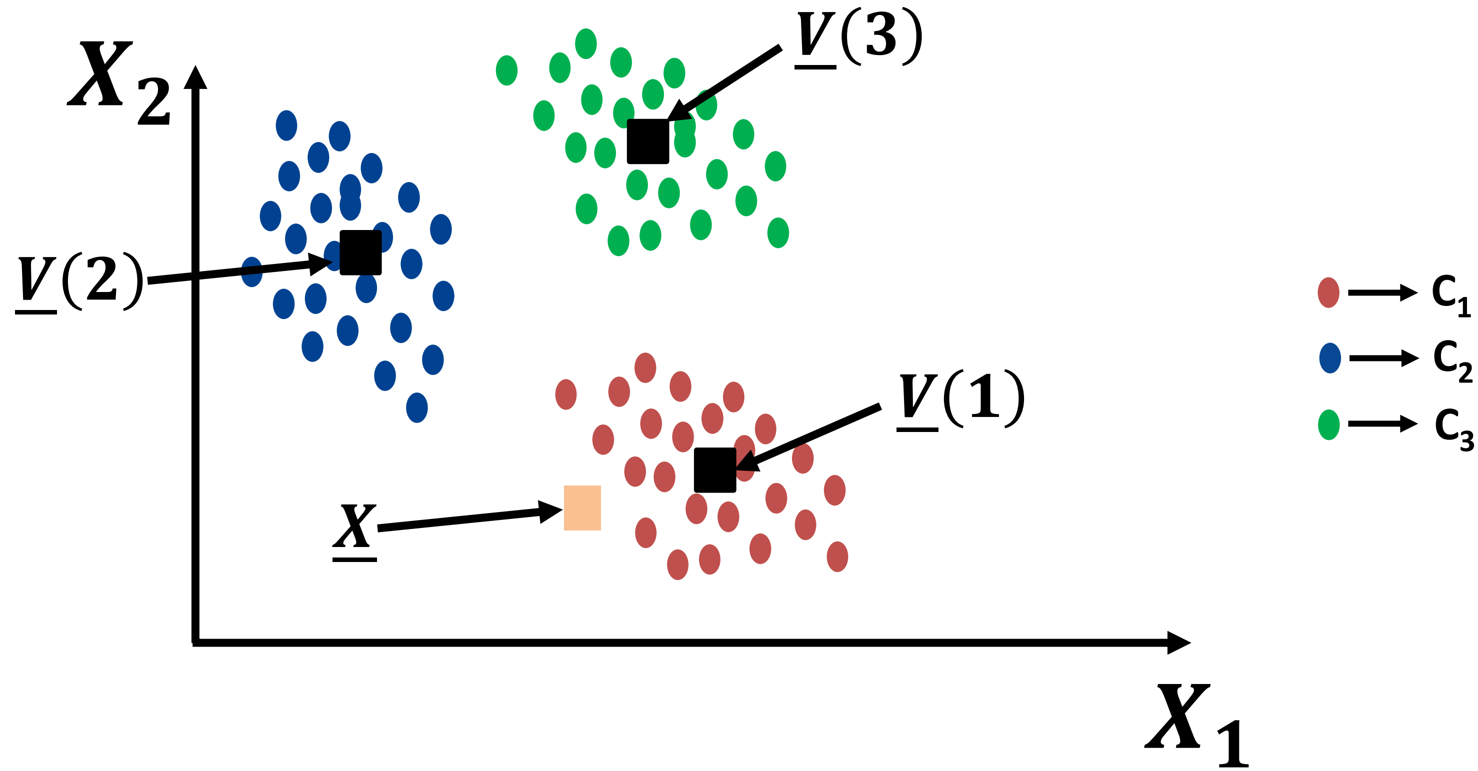
Minimum Distance Classifier

- Choose a center or a representative pattern from each class → $\underline{V}(k)$, where k is the class index



Minimum Distance Classifier

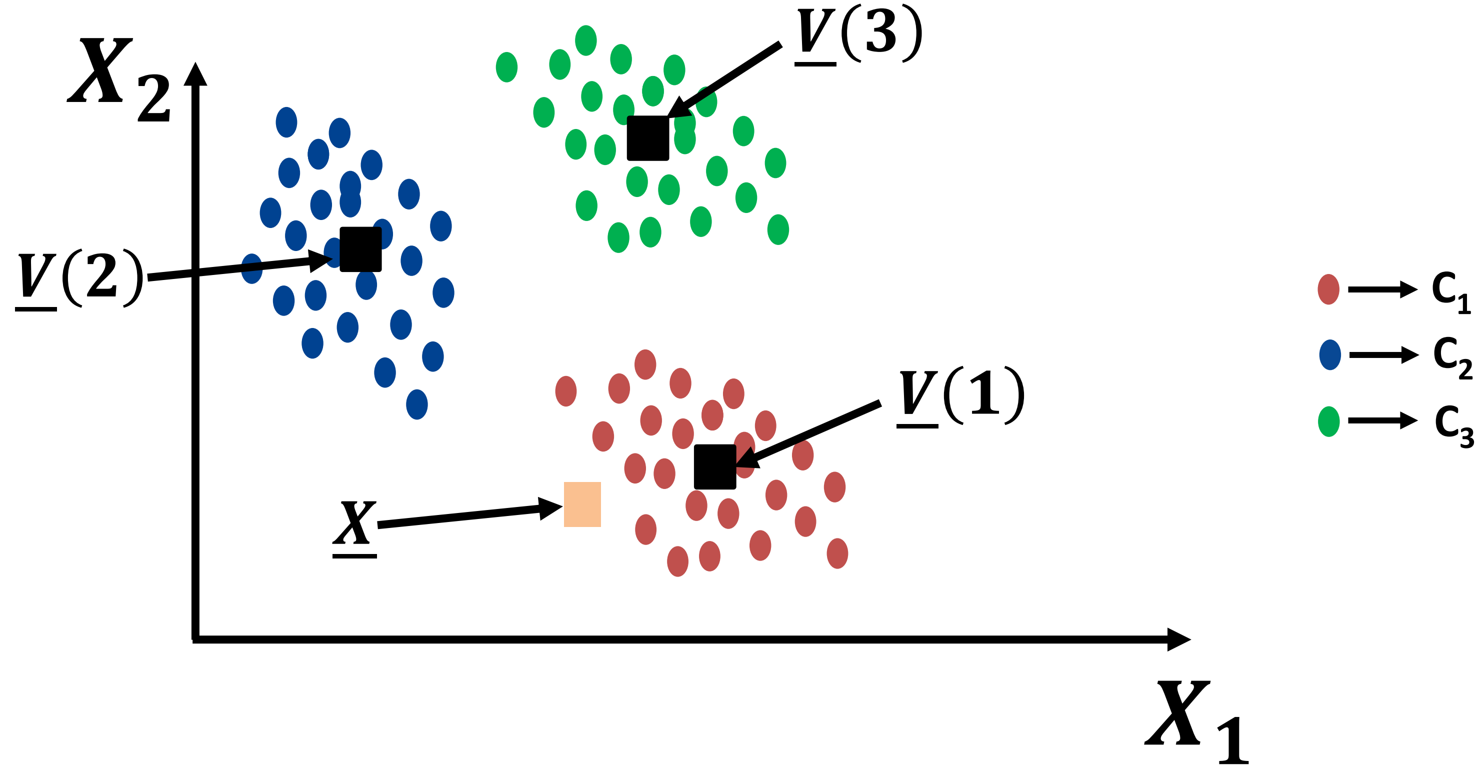
- Given a pattern $\underline{X}$ that we would like to classify



Minimum Distance Classifier

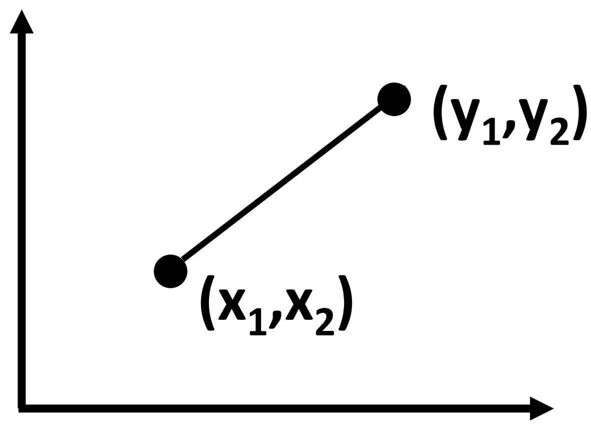
- Compute the distance from $\underline{X}$ to each center $\underline{V}(k)$:

$$\boldsymbol{d(k)} = \sum_{i=1}^{N} [\boldsymbol{V_i(k)} - \boldsymbol{X_i}]^2 \equiv \left\| \underline{\boldsymbol{V}}(\boldsymbol{k}) - \underline{\boldsymbol{X}} \right\|^2$$



Recap: Euclidean Distance

- 2D:



$$d^2 = (y_2 - x_2)^2 + (y_1 - x_1)^2$$

- N-dimensions:

$$d^2(\underline{X}, \underline{Y}) = \sum_{i=1}^{N} (Y_i - X_i)^2$$

Minimum Distance Classifier

- Find ϗ corresponding to the minimum distance:

$$\boldsymbol{ϗ} = \underset{1 \le k \le K}{\mathbf{argmin}}\, \boldsymbol{d}(\boldsymbol{k})$$

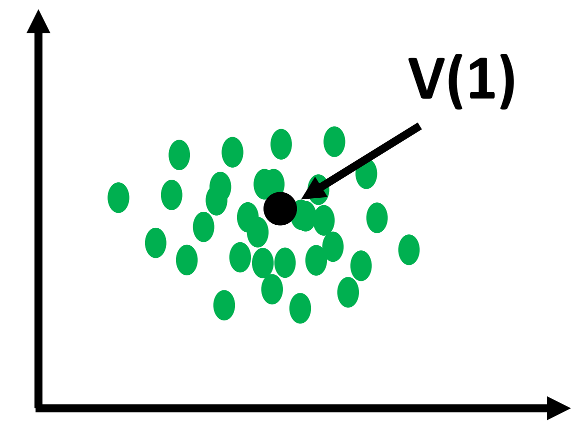
- Then our classification of is $\underline{X}$ class $C_ϗ$

- $\underline{X}$ is classified as belonging to the class corresponding to the nearest class center

Class Center Estimation

- Let $\underline{X}(m) \in C_1$,

$$\underline{V}(1) = \frac{1}{M_1} \sum_{m=1}^{M_1} \underline{X}(m)$$



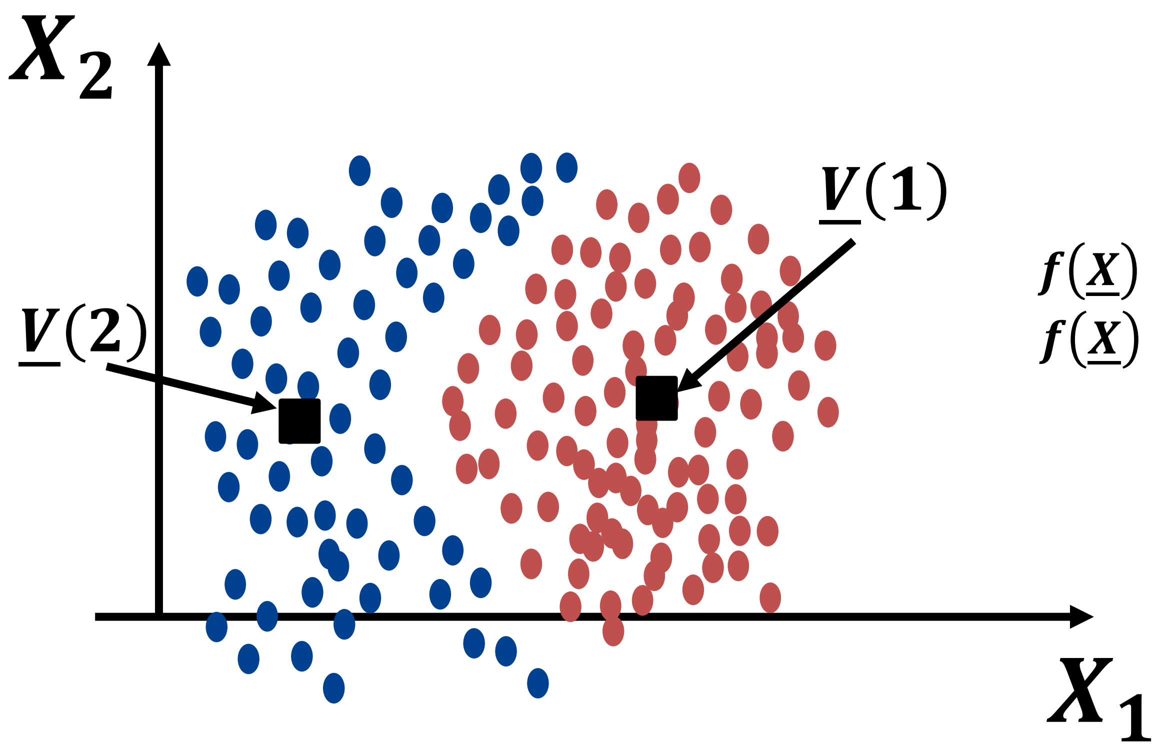
where, M1 is the number of training patterns from class C1

- This corresponds to component-wise averaging

$$V_i(1) = \frac{1}{M_1} \sum_{m=1}^{M_1} X_i(m)$$

Minimum Distance Classifier

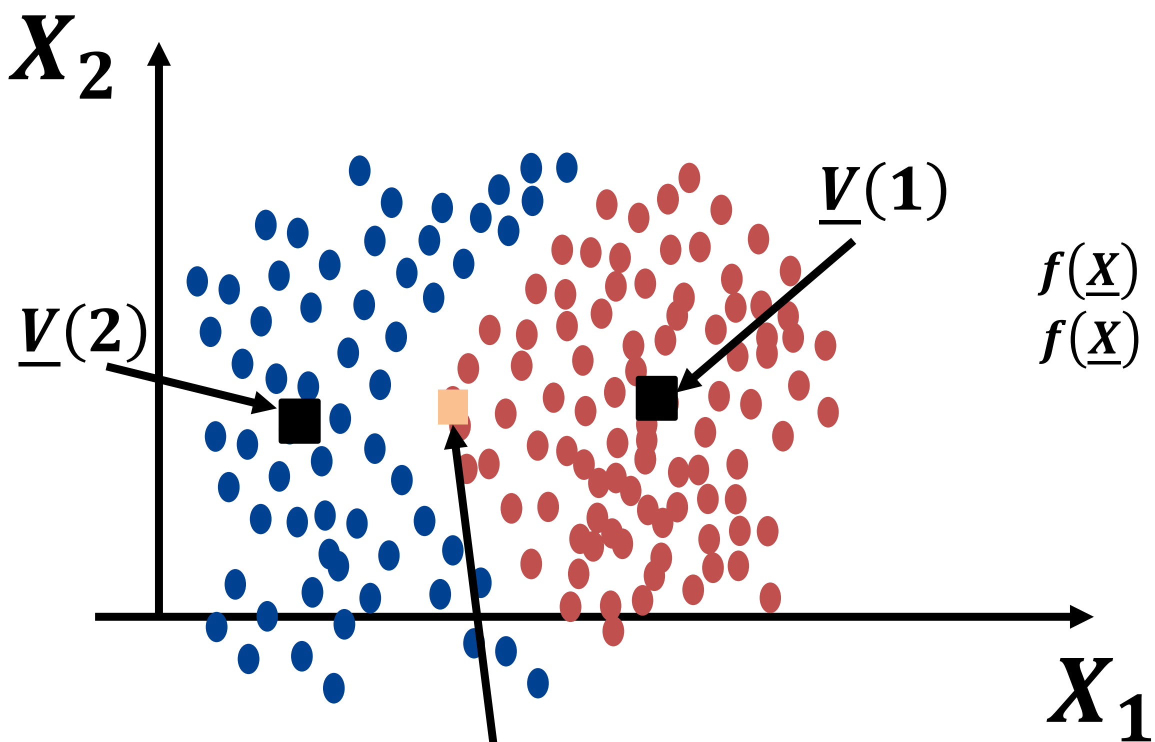
- Too simple to solve difficult problems



$f(\underline{X}) > 0 \longrightarrow c_1$
$f(\underline{X}) < 0 \longrightarrow c_2$

Minimum Distance Classifier

- Too simple to solve difficult problems

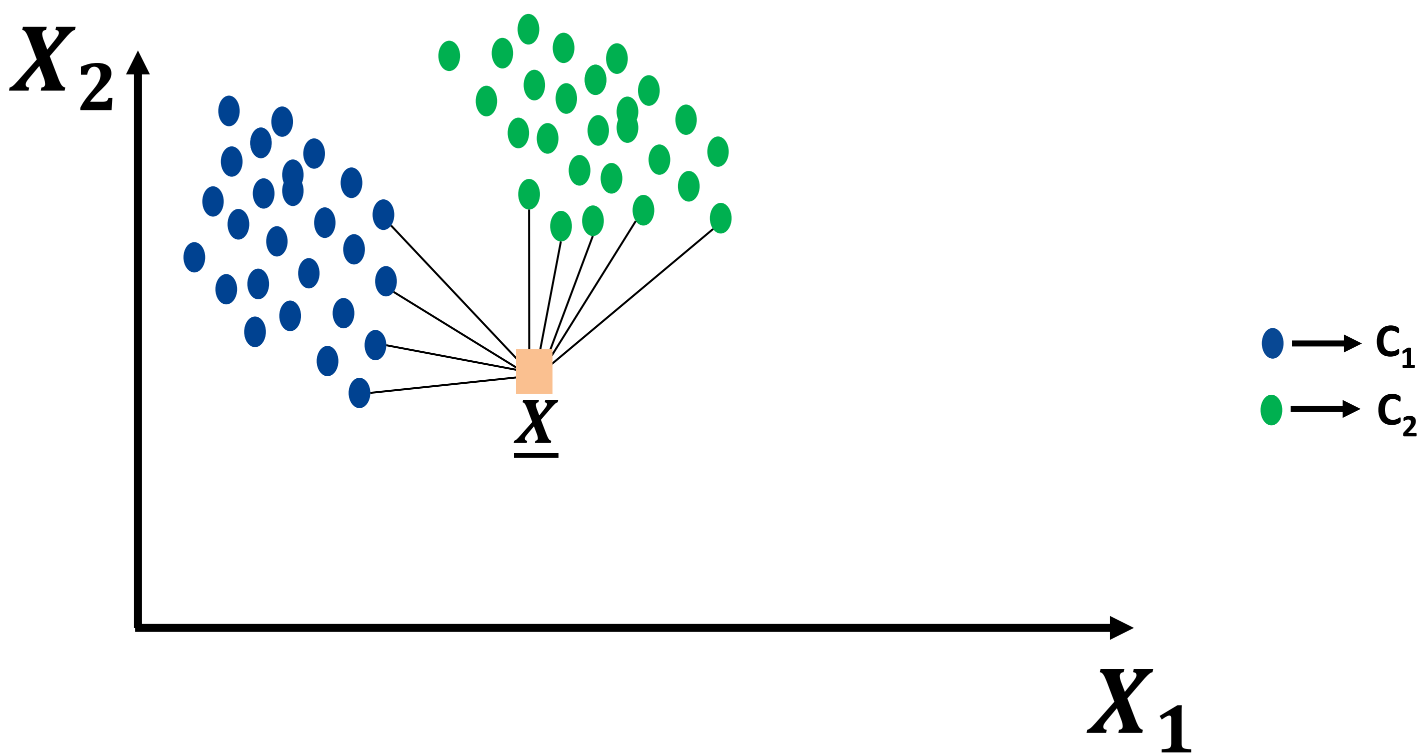


$f(\underline{X}) > 0 \longrightarrow C_1$
$f(\underline{X}) < 0 \longrightarrow C_2$

$\underline{X}$ will be classified as C_2

Nearest Neighbor Classifier

- The class of the nearest pattern to $\underline{X}$ determines its classification



Nearest Neighbor Classifier

- Compute the distance between pattern $\underline{X}$ and each pattern $\underline{X}(m)$ in the training set

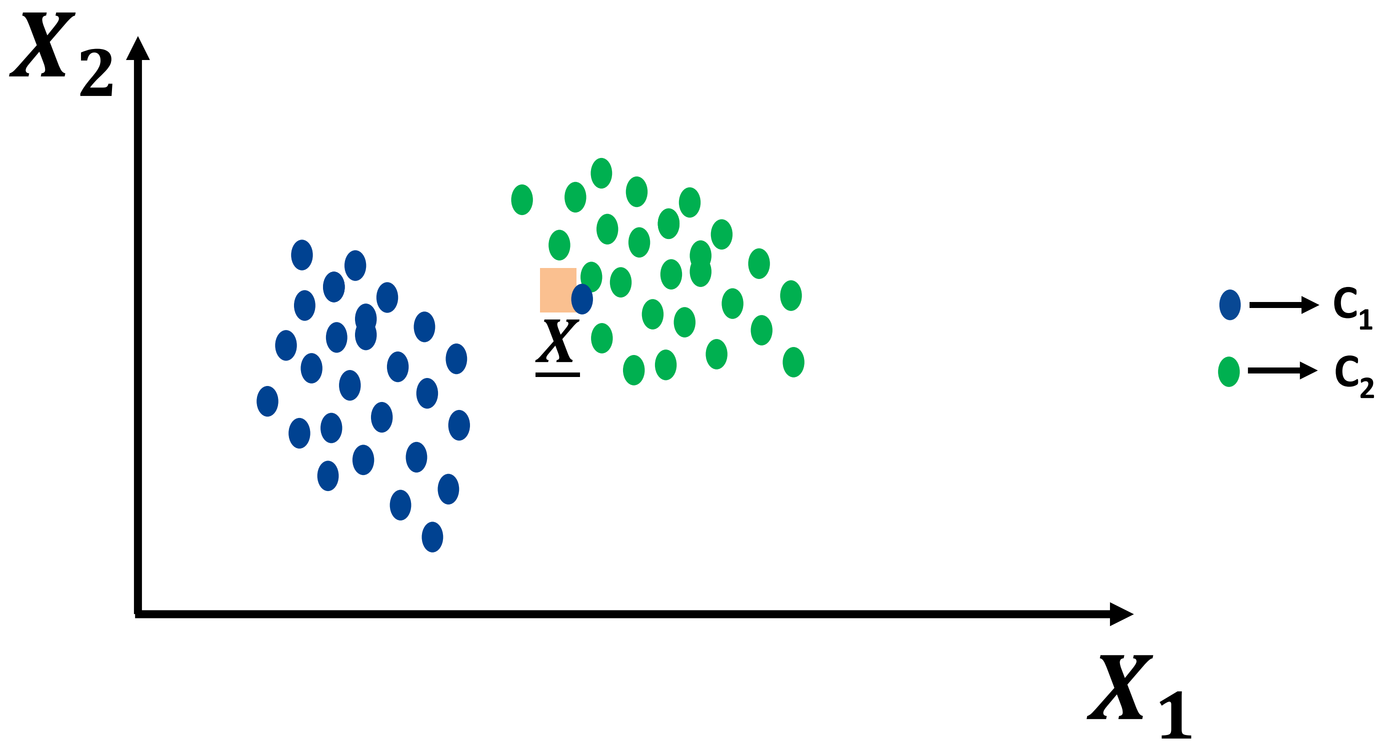
$$\boldsymbol{d}(\boldsymbol{m}) = \left\|\underline{\boldsymbol{X}} - \underline{\boldsymbol{X}}(\boldsymbol{m})\right\|^2$$

- The class of the pattern m that corresponds to the minimum distance is chosen as the classification of $\underline{X}$

Nearest Neighbor Classifier

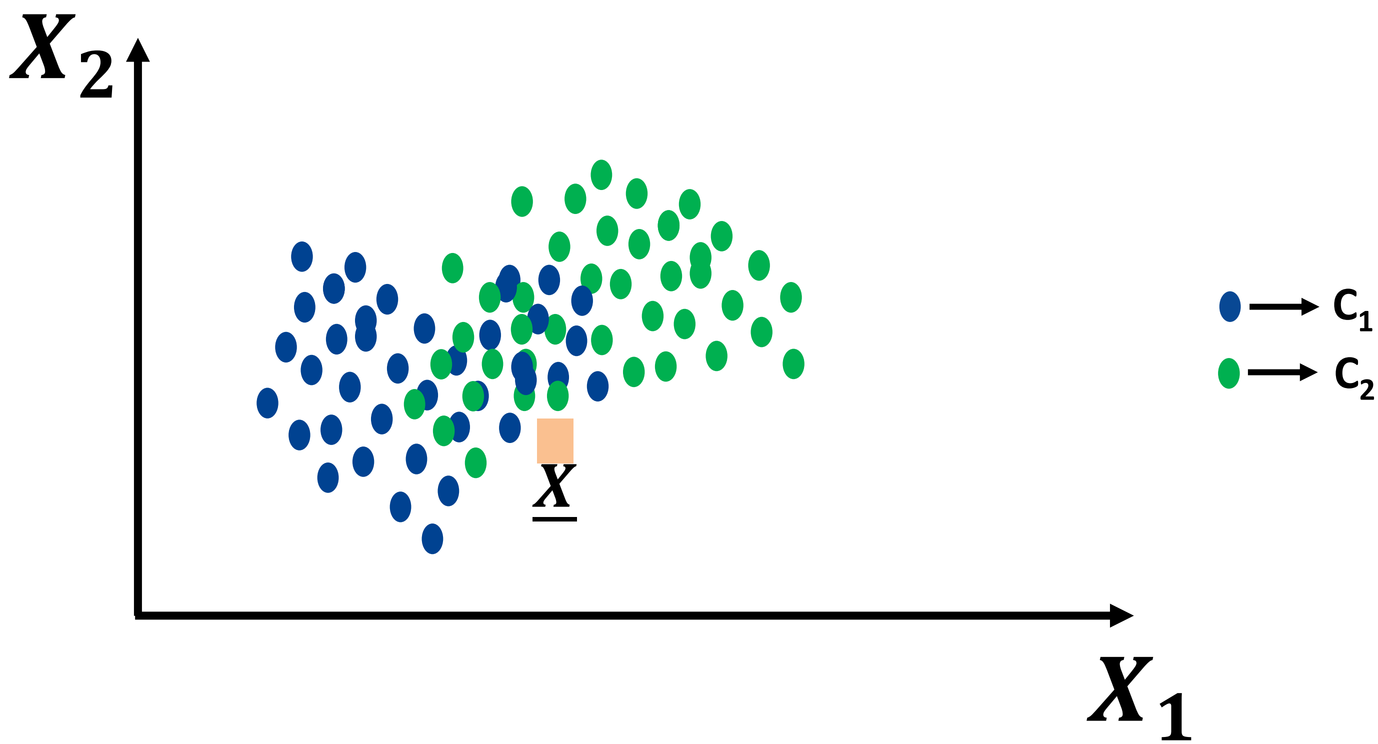
- The advantage of the nearest neighbor classifier is its simplicity

- However, a rouge pattern can affect the classification negatively



Nearest Neighbor Classifier

- Also, for patterns with large overlaps between the classes, the overlapping patterns can negatively affect performance



K-Nearest Neighbor Classifier

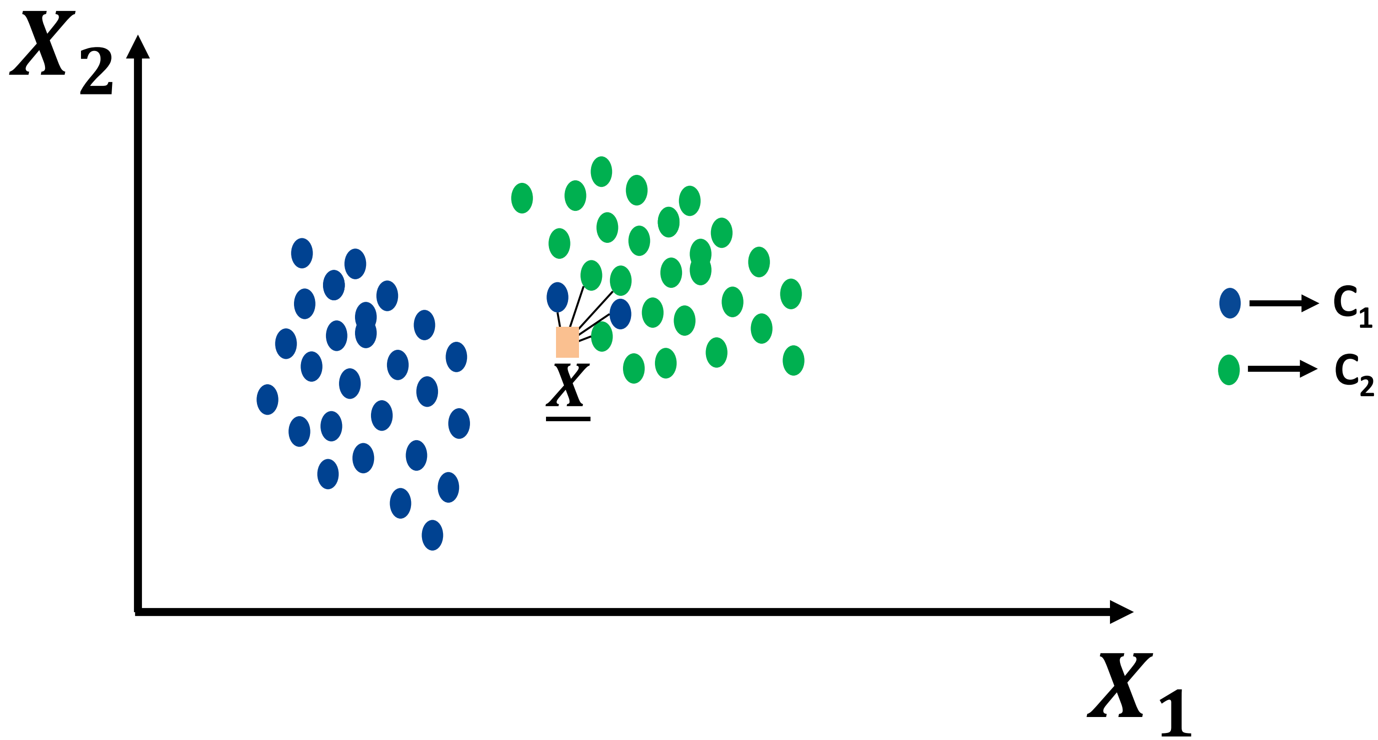
- To alleviate the problems of the NN classifier there is the k-nearest neighbor classifier

- Take the k-nearest points to point $\underline{X}$

- Choose the classification of $\underline{X}$ as the class most often represented in these k points

K-Nearest Neighbor Classifier

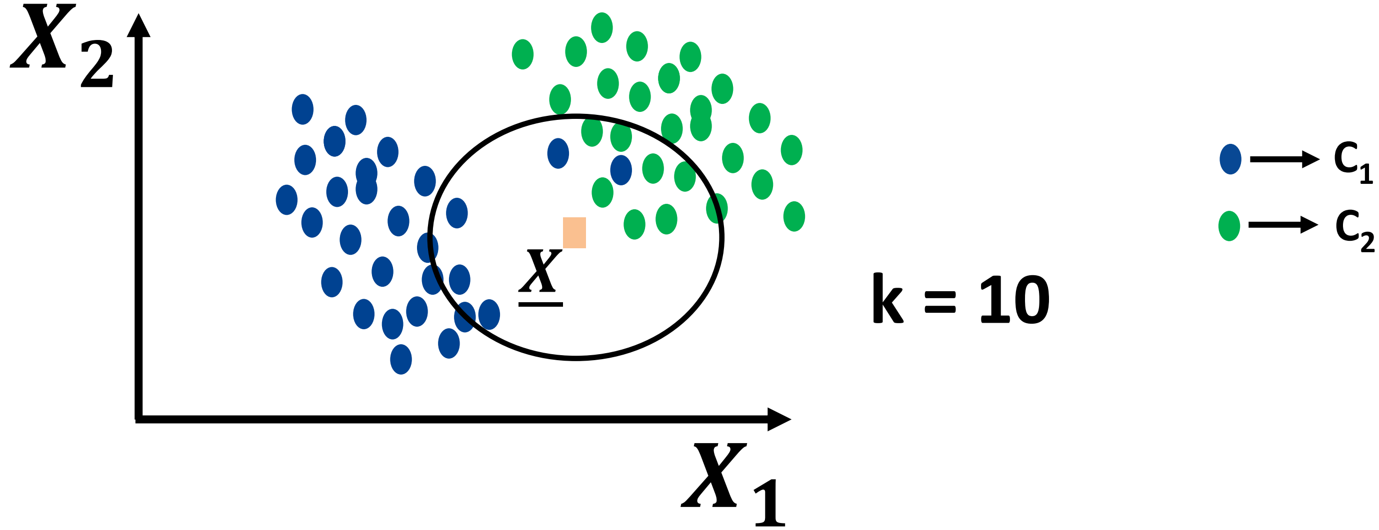
- Take k = 5



- One can see that C_2 is the majority → classify $\underline{X}$ as C_2
- The KNN rule is less dependent on strange patterns compared to the nearest neighbor classification rule

K-Nearest Neighbor Classifier

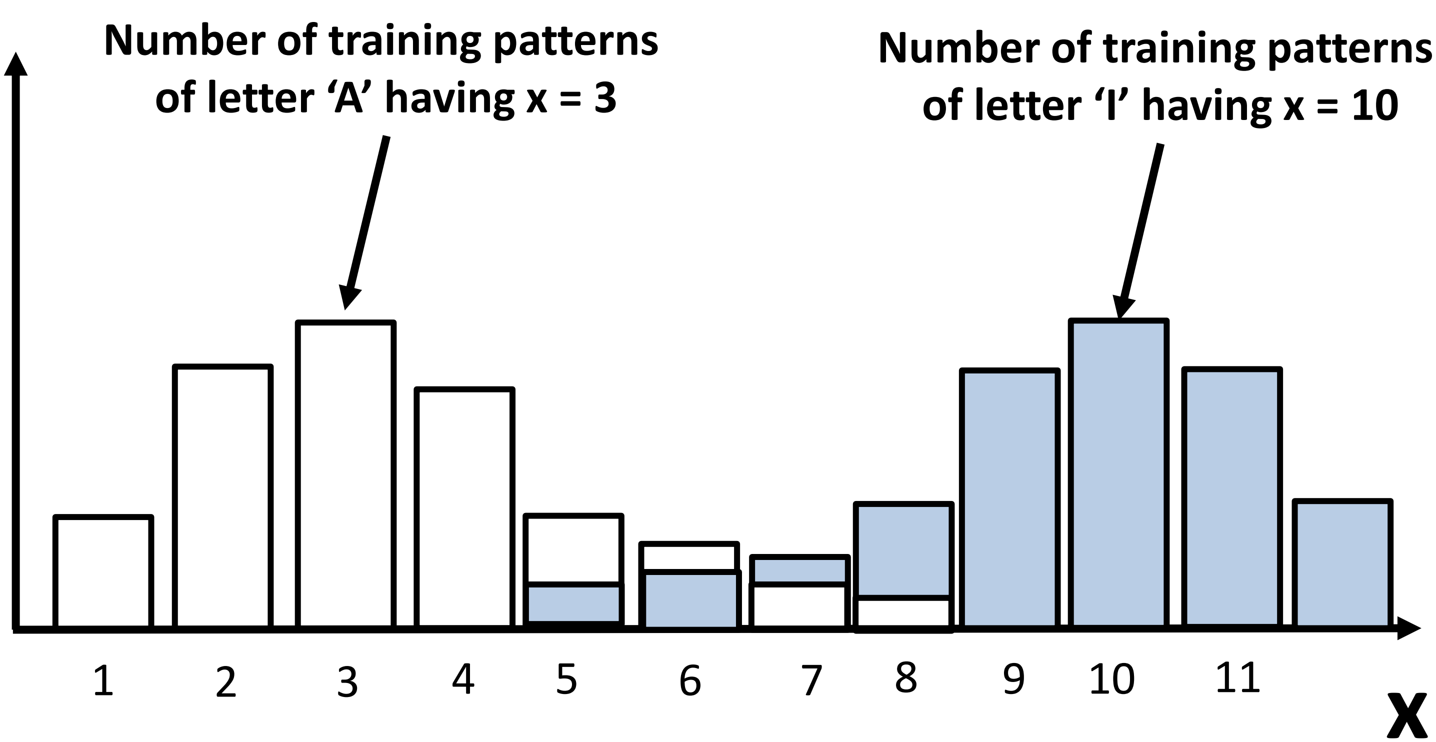
- The k-nearest neighbors could be a bit far away from $\underline{X}$



- Leading to using information that might not be relevant to the considered point $\underline{X}$

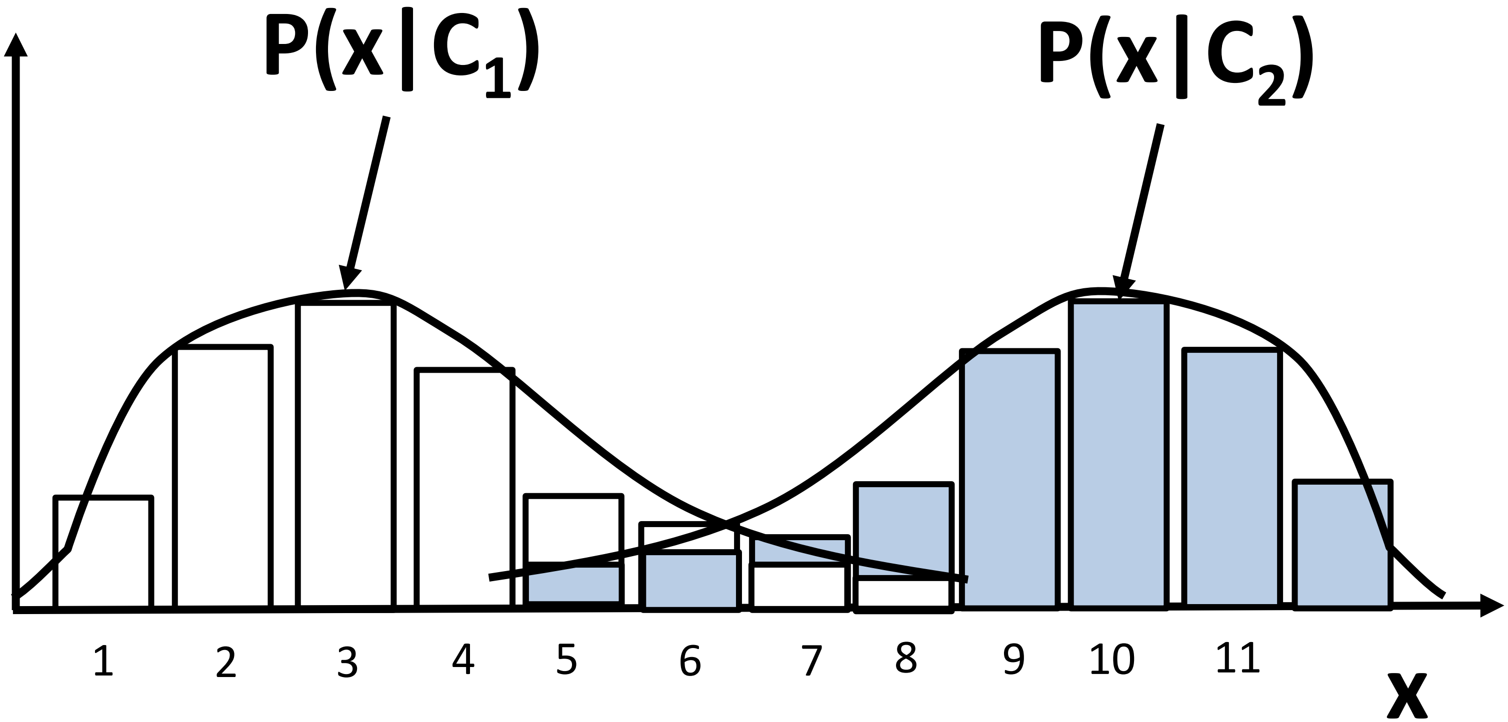
Bayes Classification Rule

- Recall: histogram for feature x from class C_1 (e.g., letter 'A')



Bayes Classification Rule

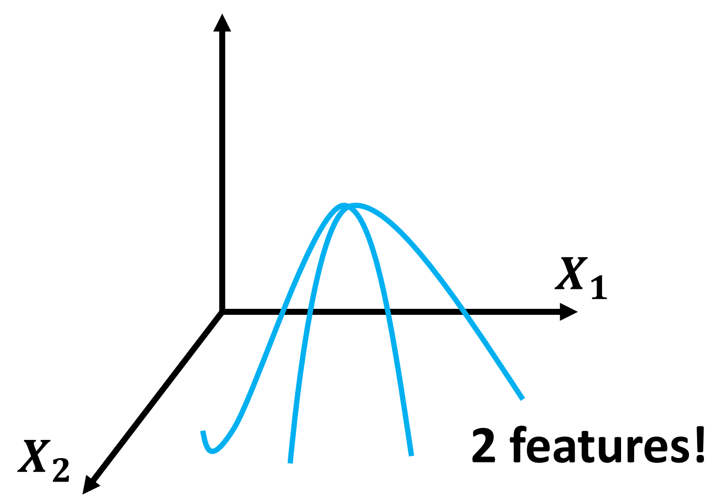
$P(x|\text{class } C_i) \equiv$ class conditional probability function
$\equiv$ probability density of feature x, given that x comes from class C_i



Bayes Classification Rule

- If $\underline{X} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_N \end{bmatrix}$ is a feature vector then:

$$\boldsymbol{P}(\underline{\boldsymbol{X}}|\boldsymbol{C_i}) = \boldsymbol{P}(\boldsymbol{X_1}, \boldsymbol{X_2}, \cdots, \boldsymbol{X_N}|\boldsymbol{C_i})$$



Bayes Classification Rule

- Given a pattern $\underline{X}$ (with unknown class) that we wish to classify:
  - Compute $P(C_1|\underline{X})$, $P(C_2|\underline{X})$, … , $P(C_K|\underline{X})$
  - Find the k giving maximum $P(C_k|\underline{X})$
- This is our classification according to the Bayes classification rule
- We classify the data point (pattern) as belonging to the ***most likely*** class

Bayes Classification Rule

- To compute $P(C_i|\underline{X})$, we use Bayes rule:

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

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

Bayes Rule:

P(A,B) = P(A|B)P(B) = P(B|A)P(A)

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

Bayes Classification Rule

- To compute $P(C_i|\underline{X})$, we use ***Bayes rule***:

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

- $P(\underline{X}|C_i)$ ≡ Class-conditional density (***defined before***)

- $P(C_i)$ ≡ Probability of class C_i before or without observing the features $\underline{X}$
  ≡ a priori probability of class C_i

Bayes Classification Rule

- The a priori probabilities represent the frequencies of the classes irrespective of the observed features

- For example in OCR, the a priori probabilities are taken as the frequency or fraction of occurrence of the different letters in a typical text

  - For the letters E & A → $P(C_i)$ will be higher

  - For letters Q & X → $P(C_i)$ will be low because they are infrequent

Bayes Classification Rule

- Find C_k giving max $P(C_k|\underline{X})$

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

  - $P(C_k|\underline{X}) \equiv$ posterior prob.
  - $P(C_k) \equiv$ a priori prob.
  - $P(\underline{X}|C_k) \equiv$ class-conditional densities

- $P(\underline{X}) = \sum_{i=1}^{K} P(\underline{X}, C_i) = \sum_{i=1}^{K} P(\underline{X}|C_i)P(C_i)$

Recap: Marginalization

- Discrete case:

$$P(A) = \sum_{i=1}^{N} P(A, B = B_i)$$

- Continuous case:

$$P(x) = \int_{-\infty}^{\infty} P(x, y)\, dy$$

- So:

$$P(\underline{X}) = \underbrace{\sum_{i=1}^{K} P(\underline{X}, C_i)}_{\text{Marginalization}} = \underbrace{\sum_{i=1}^{K} P(\underline{X} \mid C_i) P(C_i)}_{\text{Bayes rule}}$$

Law of total probability

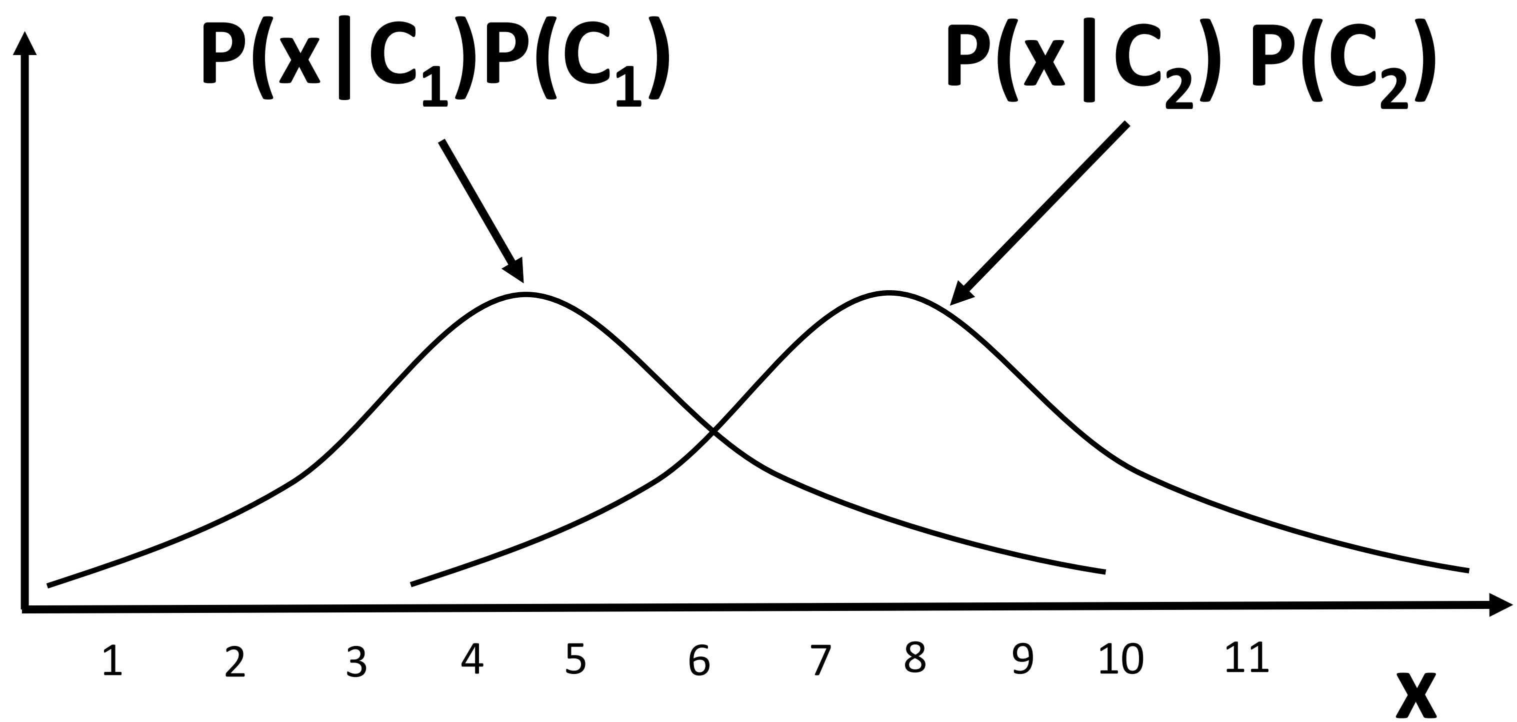
Bayes Classification Rule

$$P(C_k|\underline{X}) = \frac{P(\underline{X}|C_k)\,P(C_k)}{\sum_{i=1}^{K} P(\underline{X}|C_i)P(C_i)}$$

- In reality, we do not need to compute $P(\underline{X})$ because it is a common factor for all the terms in the expression for $P(C_k|\underline{X})$
- Hence, it will not affect which terms will end up being maximum

Bayes Classification Rule

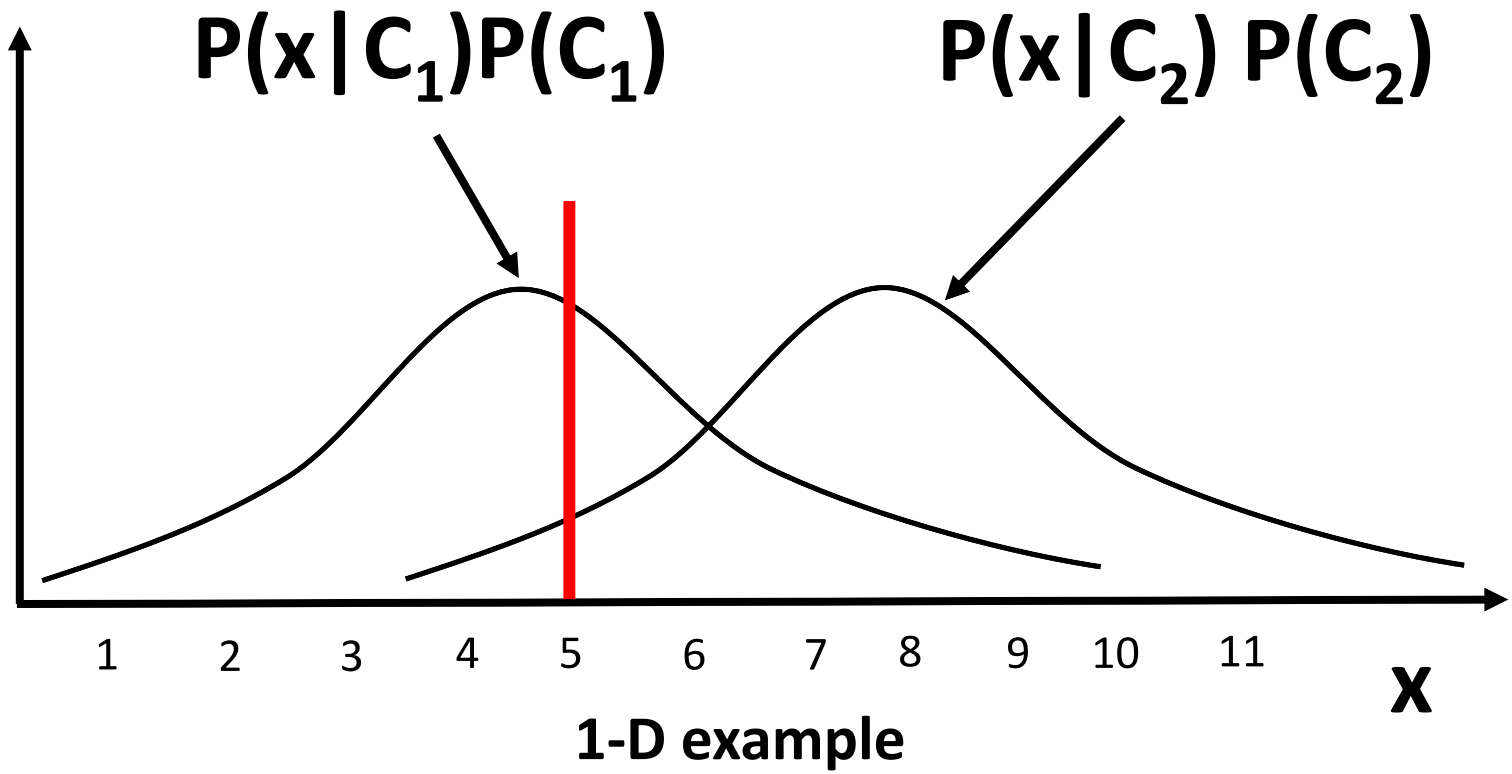
- Classify $\underline{X}$ to the class corresponding to max $P(\underline{X}|C_k)\,P(C_k)$



1-D example

Bayes Classification Rule

- Classify $\underline{X}$ to the class corresponding to max $P(\underline{X}|C_k)\,P(C_k)$



- For x=5, $P(x|C_1)P(C_1)$ has a higher value compared to $P(x|C_2)P(C_2)$
  → classify as C_1

Classification Accuracy

$$\boldsymbol{P\left(correct\ classification \middle| \underline{X}\right) = \max_{1 \le i \le K} P(C_i | \underline{X})}$$

- Example: 3-class case:
  - $P\left(C_1 \middle| \underline{X}\right) = 0.6,\ P\left(C_2 \middle| \underline{X}\right) = 0.3,\ P\left(C_3 \middle| \underline{X}\right) = 0.1$
  - You classified $\underline{X}$ as C_1 → it has highest $P(C_i | \underline{X})$
  - The probability that your classifier is correct equals to the probability that $\underline{X}$ belongs to the same class of the classification (which is 0.6)

Classification Accuracy

- Overall P(correct) is:

$$P(correct) = \int P(correct, \underline{X})\, d\underline{X}$$

Marginal prob.

$$= \int P(correct|\underline{X}) P(\underline{X})\, d\underline{X}$$

Bayes rule

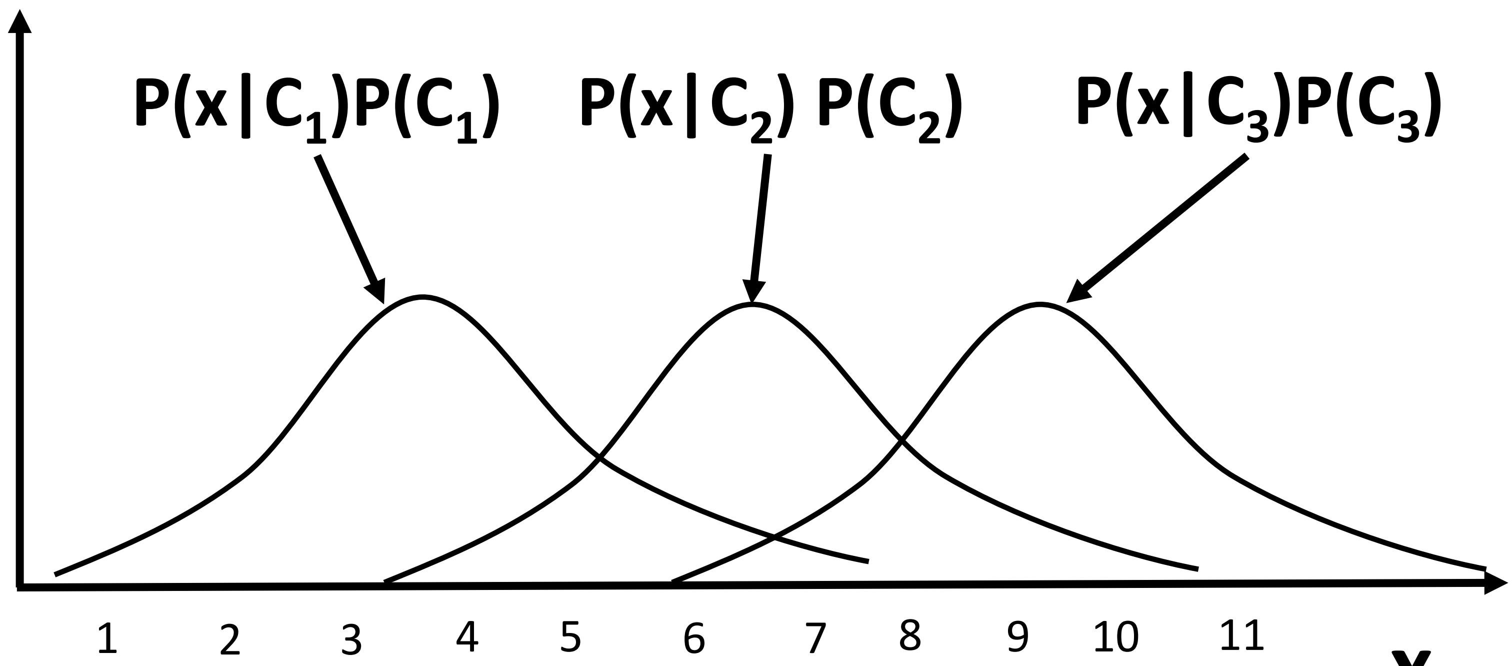
$$= \int \max_k \left[\frac{P(\underline{X}|C_k)\, P(C_k)}{\cancel{P(\underline{X})}} \right] \cancel{P(\underline{X})}\, d\underline{X}$$

$$= \int \max_k P(\underline{X}|C_k)\, P(C_k)\, d\underline{X}$$

Classification Accuracy

- Overall P(correct) is:

$$P(correct) = \int \max_k P\left(\underline{X}|C_k\right) P(C_k)\, d\underline{X}$$

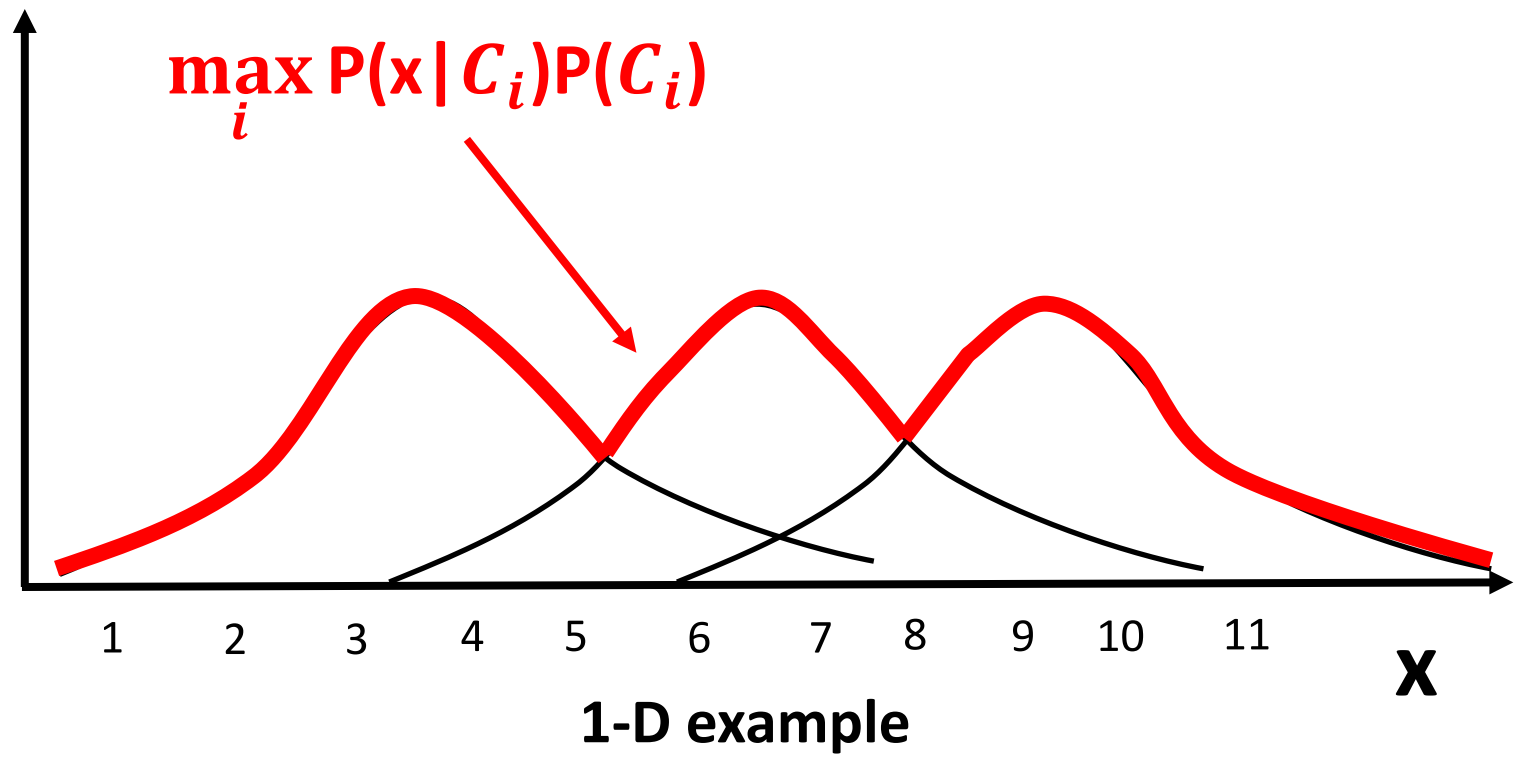


1-D example

Classification Accuracy

- Overall P(correct) is:

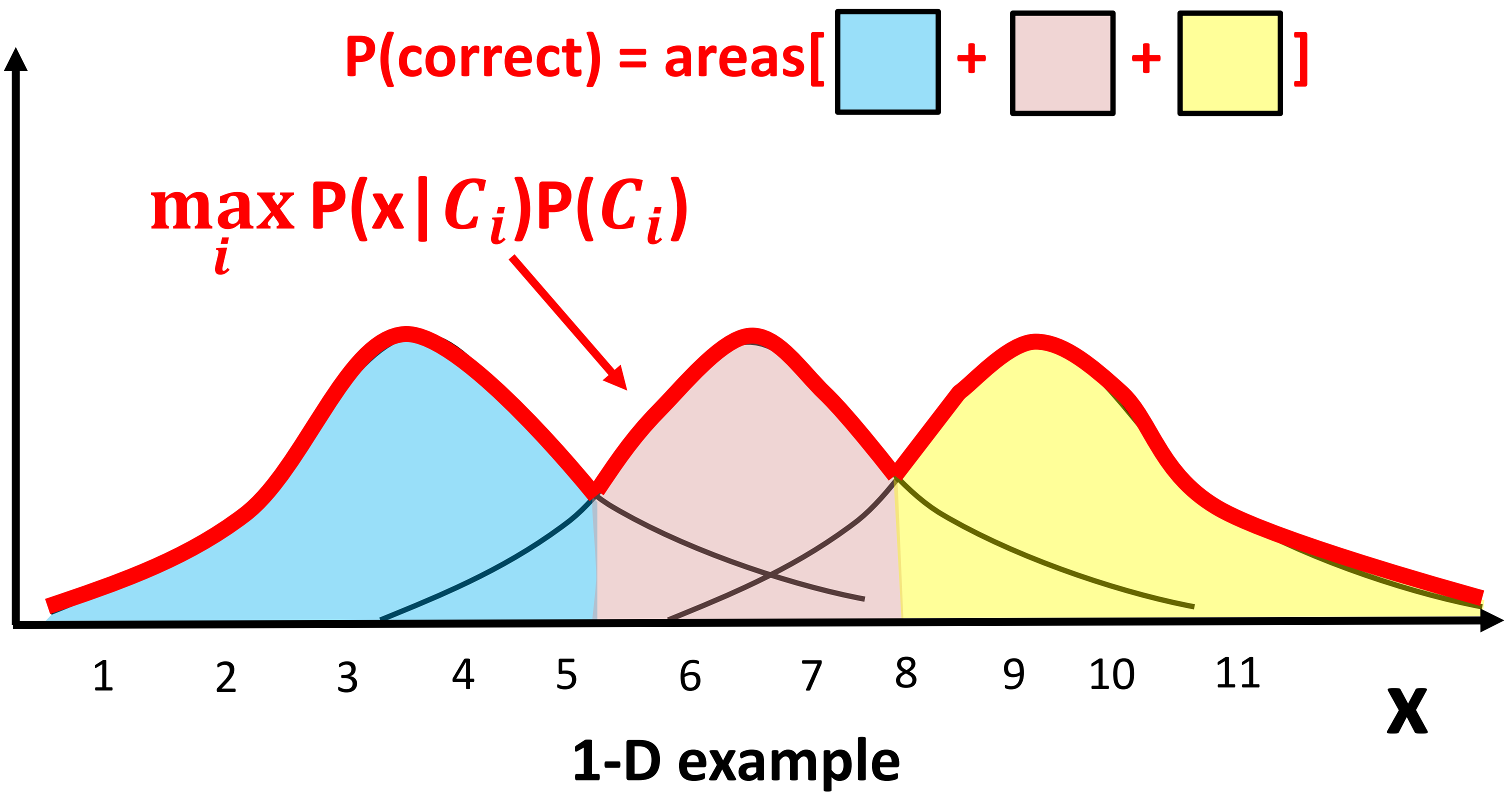
$$P(correct) = \int \max_k P(\underline{X}|C_k)\, P(C_k)\, d\underline{X}$$



Classification Accuracy

- Overall P(correct) is:

$$P(correct) = \int \max_k P(\underline{X}|C_k)\, P(C_k)\, d\underline{X}$$



Classification Accuracy

$$P(correct) = \int \max_k P(\underline{X}|C_k)\, P(C_k)\, d\underline{X}$$

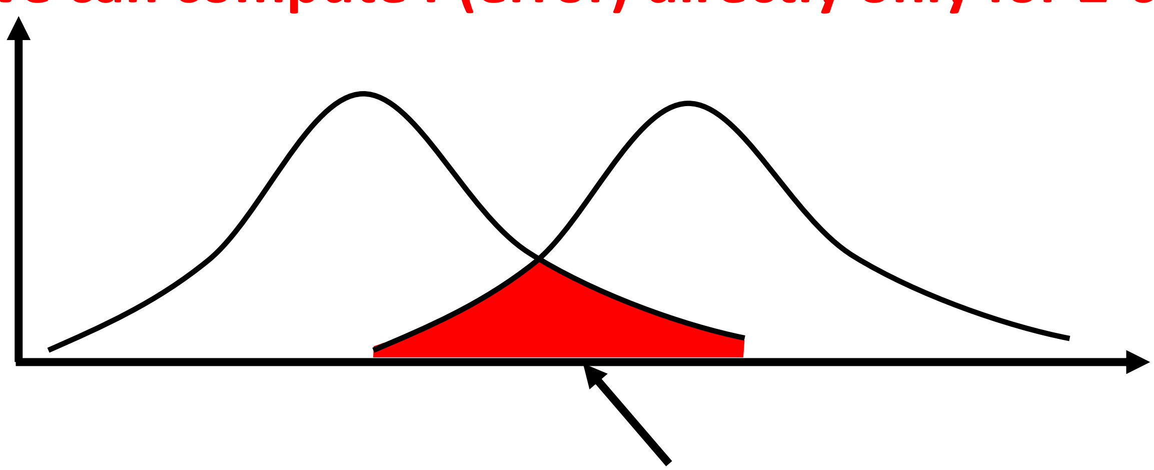
$$P(error) = 1 - P(correct)$$

Classification Accuracy

$$P(correct) = \int \max_k P(\underline{X}|C_k)\, P(C_k)\, d\underline{X}$$

$$P(error) = 1 - P(correct)$$

We can compute P(error) directly only for 2-class case!



area = P(error)

Acknowledgment

- These slides have been created relying on lecture notes of Prof. Dr. Amir Atiya