

# KNN Methods

Wasin Siwasarit

Faculty of Economics, Thammasat University

# Battle Plan of Today's Lecture

Nearest Neighbor Methods

Nearest Neighbor Decision Boundary

# Outline

## Nearest Neighbor Methods

### Nearest Neighbor Decision Boundary

By the end of this lecture, you should be able to:

- ▶ Understand the Key Concepts of Nearest Neighbor Methods

By the end of this lecture, you should be able to:

- ▶ Understand the Key Concepts of Nearest Neighbor Methods
- ▶ Understand the Problem of the Curse of Dimensionality

By the end of this lecture, you should be able to:

- ▶ Understand the Key Concepts of Nearest Neighbor Methods
- ▶ Understand the Problem of the Curse of Dimensionality
- ▶ Be able to Code the kNN in Python : Scikit-learn

By the end of this lecture, you should be able to:

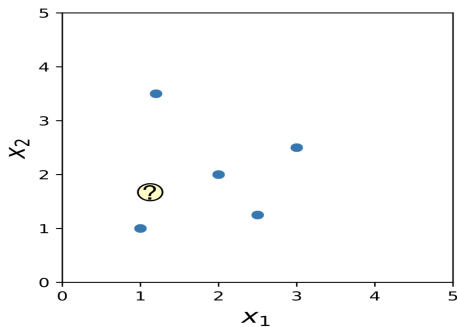
- ▶ Understand the Key Concepts of Nearest Neighbor Methods
- ▶ Understand the Problem of the Curse of Dimensionality
- ▶ Be able to Code the kNN in Python : Scikit-learn
- ▶ Compare Pros and Cons

By the end of this lecture, you should be able to:

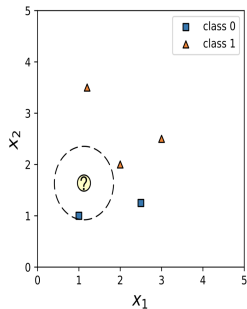
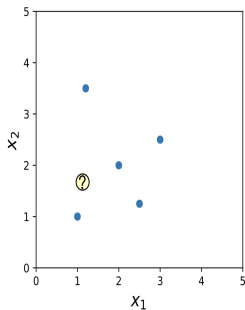
- ▶ Understand the Key Concepts of Nearest Neighbor Methods
- ▶ Understand the Problem of the Curse of Dimensionality
- ▶ Be able to Code the kNN in Python : Scikit-learn
- ▶ Compare Pros and Cons
- ▶ Be able to Improve KNN Performance



# 1-Nearest Neighbor



# 1-Nearest Neighbor



## Training Step

$$\langle \mathbf{x}^{[i]}, y^{[i]} \rangle \in \mathcal{D} \quad (|\mathcal{D}| = n)$$

## Nearest Neighbor Prediction Step

`closest_point := None`

`closest_distance :=  $\infty$`

- for  $i = 1, \dots, n$ :
  - `current_distance :=  $d(\mathbf{x}^{[i]}, \mathbf{x}^{[q]})$`
  - if `current_distance < closest_distance`:
    - `closest_distance := current_distance`
    - `closest_point :=  $\mathbf{x}^{[i]}$`
- return  `$f(\text{closest\_point})$`

`closest_point` is the label of  $\langle \mathbf{x}^{[i]}, f(\mathbf{x}^{[i]}) \rangle$

## Commonly used: Euclidean Distance ( $L^2$ )

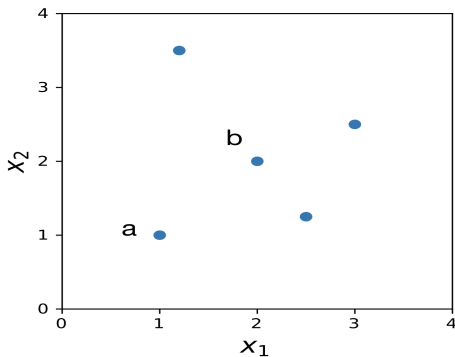
$$d(\mathbf{x}^{[a]}, \mathbf{x}^{[b]}) = \sqrt{\sum_{j=1}^m \left( x_j^{[a]} - x_j^{[b]} \right)^2}$$

# Outline

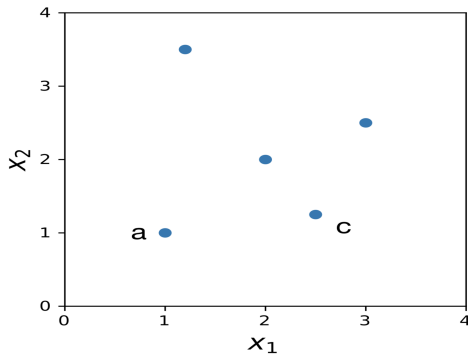
Nearest Neighbor Methods

Nearest Neighbor Decision Boundary

## Decision Boundary Between (a) and (b)

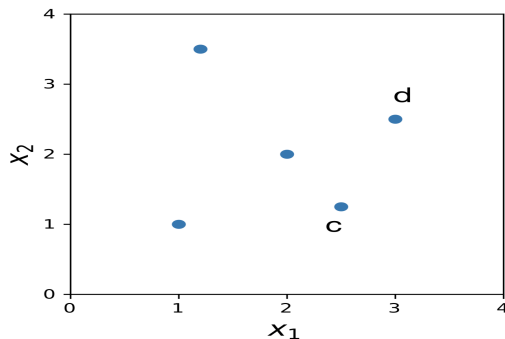


## Decision Boundary Between (a) and (c)

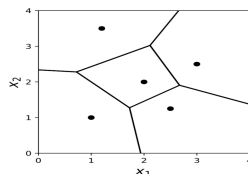
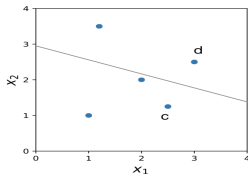
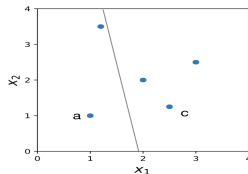
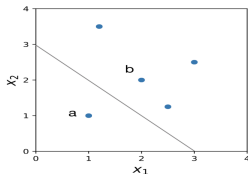




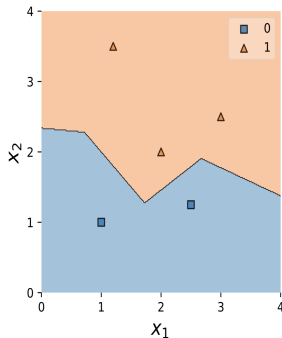
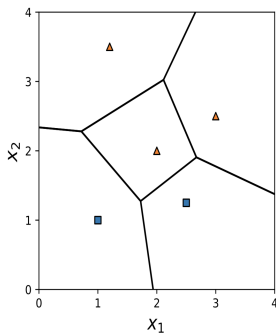
## Decision Boundary Between (b) and (c)



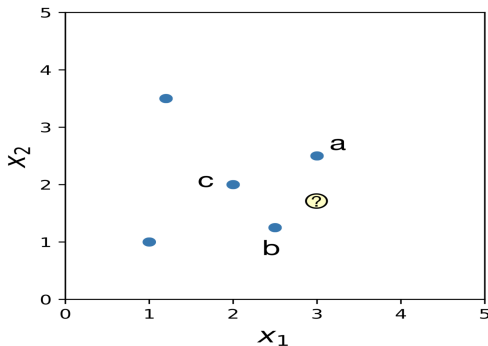
## Decision Boundary 1NN



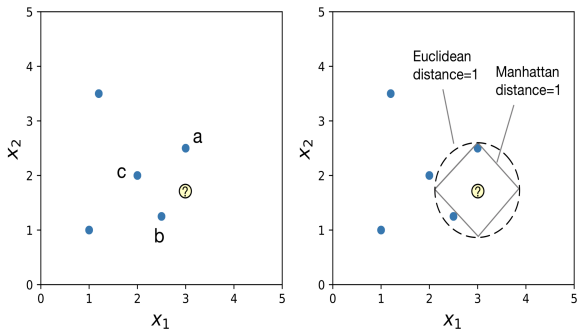
## Decision Boundary 1-NN



## Which Point is Closest?



## Depends on the Distance Measure!



# Continuous Distance Measures

Euclidean

Manhattan

Minkowski: 
$$d(\mathbf{x}^{[a]}, \mathbf{x}^{[b]}) = \left[ \sum_{j=1}^m \left( |x^{[a]} - x^{[b]}| \right)^p \right]^{\frac{1}{p}}$$

Mahalanobis

...

## Discrete Distance Measures

Hamming: 
$$d(\mathbf{x}^{[a]}, \mathbf{x}^{[b]}) = \sum_{j=1}^m \left| x^{[a]} - x^{[b]} \right|$$

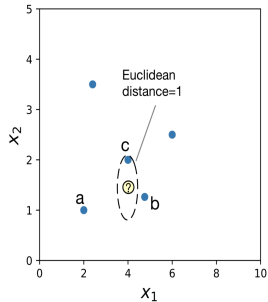
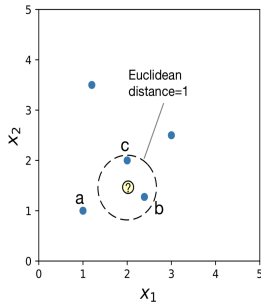
Jaccard/Tanimoto

Cosine similarity

Dice

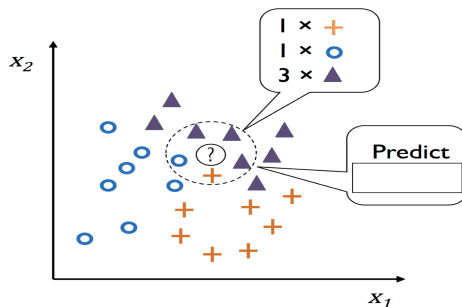
...

# Feature Scaling






## k-Nearest Neighbors



**A**

y:  

Majority vote: 

Purity vote: 

**B**

y:   

Majority vote: None

Purity vote: 

# kNN for Classification

$$\mathcal{D}_k = \{\langle \mathbf{x}^{[1]}, f(\mathbf{x}^{[1]}) \rangle, \dots, \langle \mathbf{x}^{[k]}, f(\mathbf{x}^{[k]}) \rangle\} \quad \mathcal{D}_k \subseteq \mathcal{D}$$

# kNN for Classification

$$\mathcal{D}_k = \{\langle \mathbf{x}^{[1]}, f(\mathbf{x}^{[1]}) \rangle, \dots, \langle \mathbf{x}^{[k]}, f(\mathbf{x}^{[k]}) \rangle\} \quad \mathcal{D}_k \subseteq \mathcal{D}$$

$$h(\mathbf{x}^{[q]}) = \arg \max_{y \in \{1, \dots, t\}} \sum_{i=1}^k \delta(y, f(\mathbf{x}^{[i]}))$$

$$\delta(a, b) = \begin{cases} 1, & \text{if } a = b, \\ 0, & \text{if } a \neq b. \end{cases}$$

# kNN for Classification

$$\mathcal{D}_k = \{\langle \mathbf{x}^{[1]}, f(\mathbf{x}^{[1]}) \rangle, \dots, \langle \mathbf{x}^{[k]}, f(\mathbf{x}^{[k]}) \rangle\} \quad \mathcal{D}_k \subseteq \mathcal{D}$$

$$h(\mathbf{x}^{[q]}) = \arg \max_{y \in \{1, \dots, t\}} \sum_{i=1}^k \delta(y, f(\mathbf{x}^{[i]}))$$

$$\delta(a, b) = \begin{cases} 1, & \text{if } a = b, \\ 0, & \text{if } a \neq b. \end{cases}$$

$$h(\mathbf{x}^{[t]}) = \mathbf{mode}(\{f(\mathbf{x}^{[1]}), \dots, f(\mathbf{x}^{[k]})\})$$

## kNN for Regression

$$\mathcal{D}_k = \{ \langle \mathbf{x}^{[1]}, f(\mathbf{x}^{[1]}) \rangle, \dots, \langle \mathbf{x}^{[k]}, f(\mathbf{x}^{[k]}) \rangle \} \quad \mathcal{D}_k \subseteq \mathcal{D}$$

$$h(\mathbf{x}^{[t]}) = \frac{1}{k} \sum_{i=1}^k f(\mathbf{x}^{[i]})$$

## Categories

- eager vs lazy;
- batch vs online;
- parametric vs nonparametric;
- discriminative vs generative.