Distance-based Classification

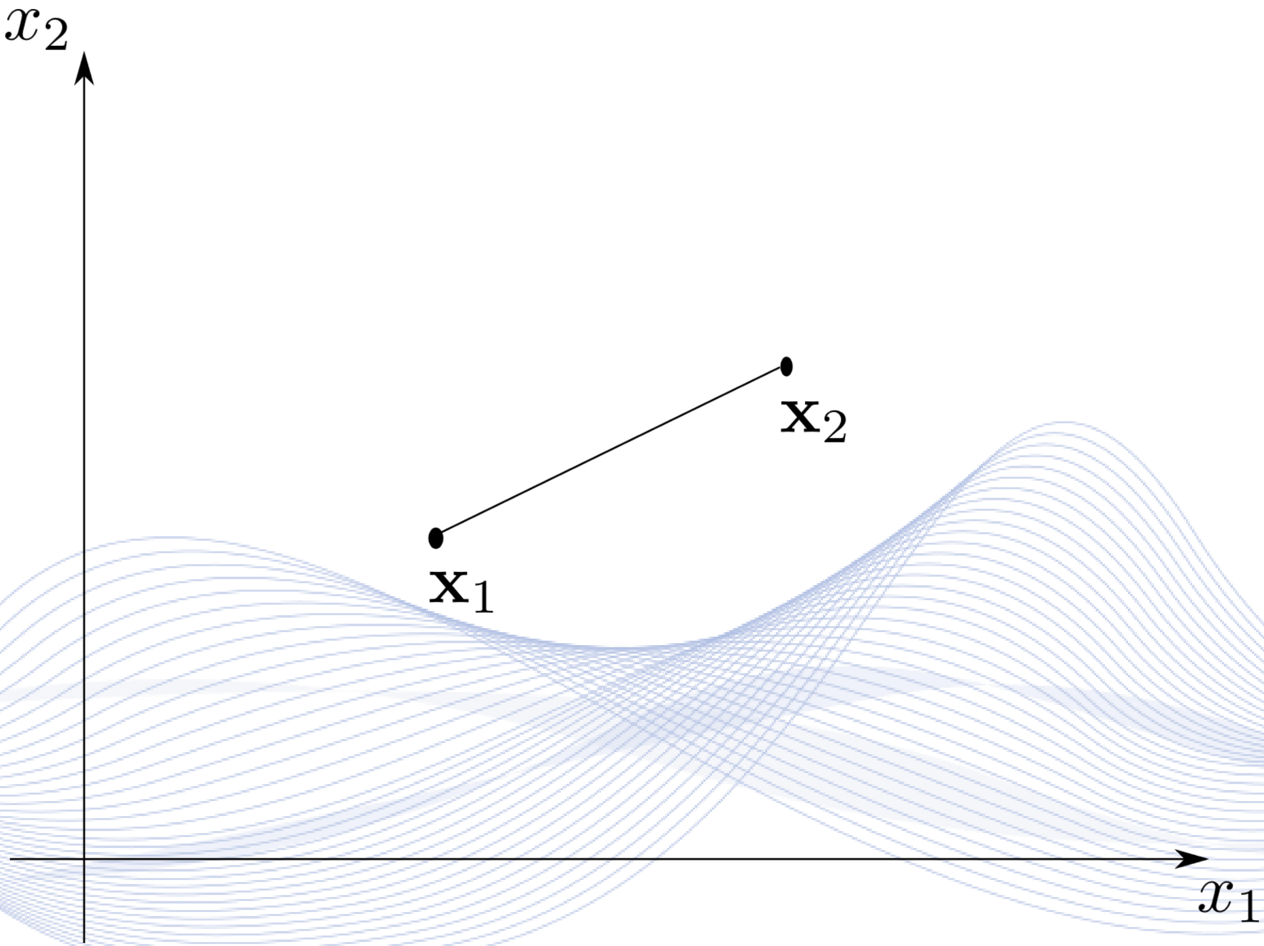
P. Papageorgiou, Prof. Ioannis Pitas
Aristotle University of Thessaloniki
pitas@csd.auth.gr
www.aiia.csd.auth.gr
Version 2.5.3

Artificial Intelligence &
Information Analysis Lab

Outline

- **Nearest neighbor classification**
- Supervised Learning Vector Quantization

Distance Measures



Euclidean distance between two points.

K-means Algorithm

- Distances between a feature vector and a class center:
  - ***Mahalanobis distance***:

$$d(\mathbf{x}_i, \mathbf{m}_j) = (\mathbf{x}_i - \mathbf{m}_j)^T \mathbf{A} (\mathbf{x}_i - \mathbf{m}_j).$$

  - $\mathbf{A}$: symmetric, positive definite matrix.
  - ***Euclidean distance***:

$$d(\mathbf{x}_i, \mathbf{m}_j) = (\mathbf{x}_i - \mathbf{m}_j)^T (\mathbf{x}_i - \mathbf{m}_j).$$

Artificial Intelligence & Information Analysis Lab

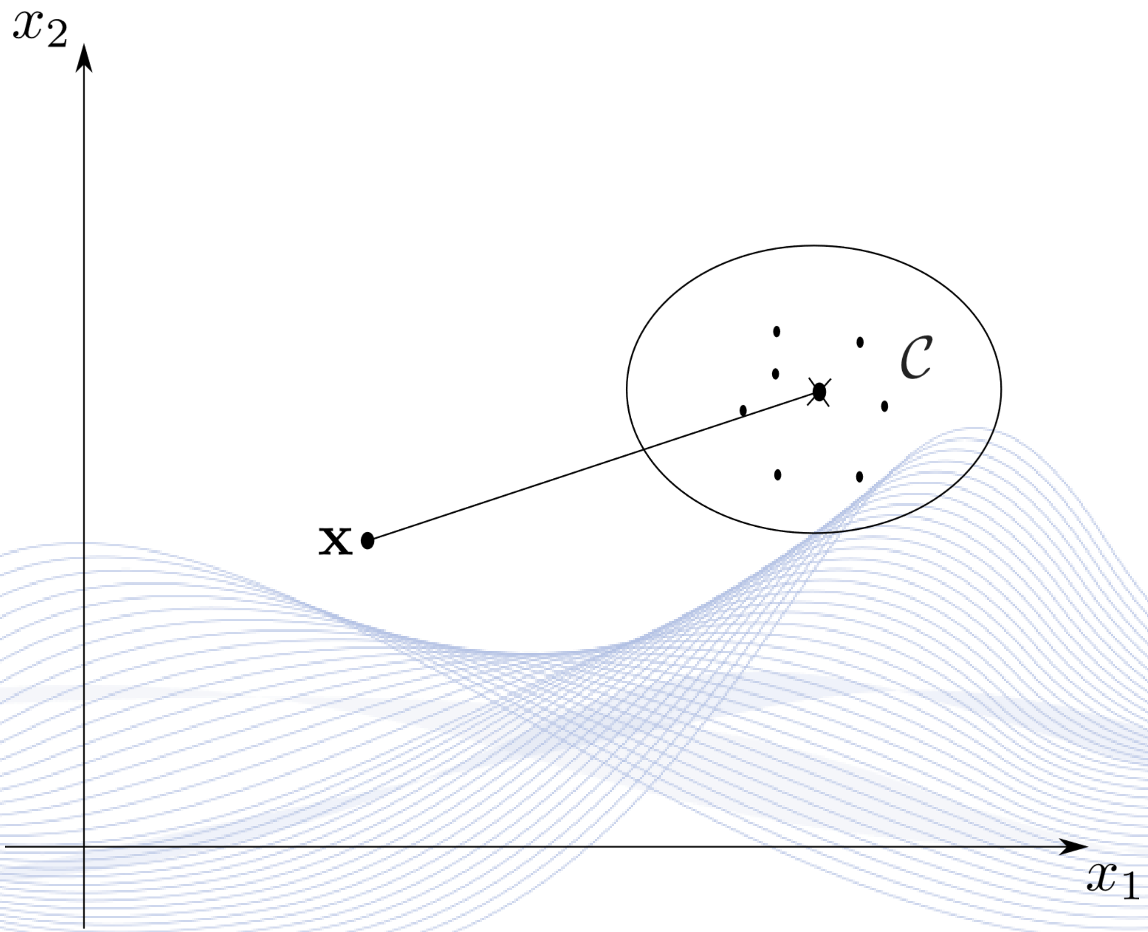
K-means Algorithm

- ***Minkowski distance***:

$$d(\mathbf{x}_i, \mathbf{m}_j) = \left(\sum_{k=1}^{l} |x_{ik} - m_{jk}|^p\right)^{\frac{1}{p}}.$$

- x_{ik}, m_{jk} are the k-th coordinates of $\mathbf{x}_i, \mathbf{m}_j$ respectively.

Distance Measures



Distance between point and set (set center).

Distance Measures

Distance Functions between a Point and a Set (class)

- Distance $d'(\mathbf{x}, \mathcal{C})$ between vector $\mathbf{x}$ and class $\mathcal{C}$:
  - Distance to class center (vector) $\mathbf{m}$: $d'(\mathbf{x}, \mathcal{C}) = d(\mathbf{x}, \mathbf{m})$.
  - Max Distance function: $d'(\mathbf{x}, \mathcal{C}) = \max_{\mathbf{y} \in \mathcal{C}} d(\mathbf{x}, \mathbf{y})$.
  - Min Distance function: $d'(\mathbf{x}, \mathcal{C}) = \min_{\mathbf{y} \in \mathcal{C}} d(\mathbf{x}, \mathbf{y})$.
  - Average Distance function: $d'(\mathbf{x}, \mathcal{C}) = \frac{1}{|\mathcal{C}|} \sum_{\mathbf{y} \in \mathcal{C}} d(\mathbf{x}, \mathbf{y})$.
- $|\mathcal{C}|$: set $\mathcal{C}$ cardinality.

Distance Measures

Class center:

- Representative vector of a data vector set:
  - ***Arithmetic mean vector***:

$$\mathbf{m} = \frac{1}{|\mathcal{C}|} \sum_{\mathbf{x} \in \mathcal{C}} \mathbf{x}.$$

  - Sensitive to outliers.

Distance Measures

- ***Vector median***:

$$\sum_{\mathbf{y}\in\mathcal{C}} d(\mathbf{m}_v,\mathbf{y}) \leq \sum_{\mathbf{y}\in\mathcal{C}} d(\mathbf{z},\mathbf{y}),\ \mathbf{m}_v \in \mathcal{C},\ \forall\mathbf{z} \in \mathcal{C}.$$

- ***Median center***:

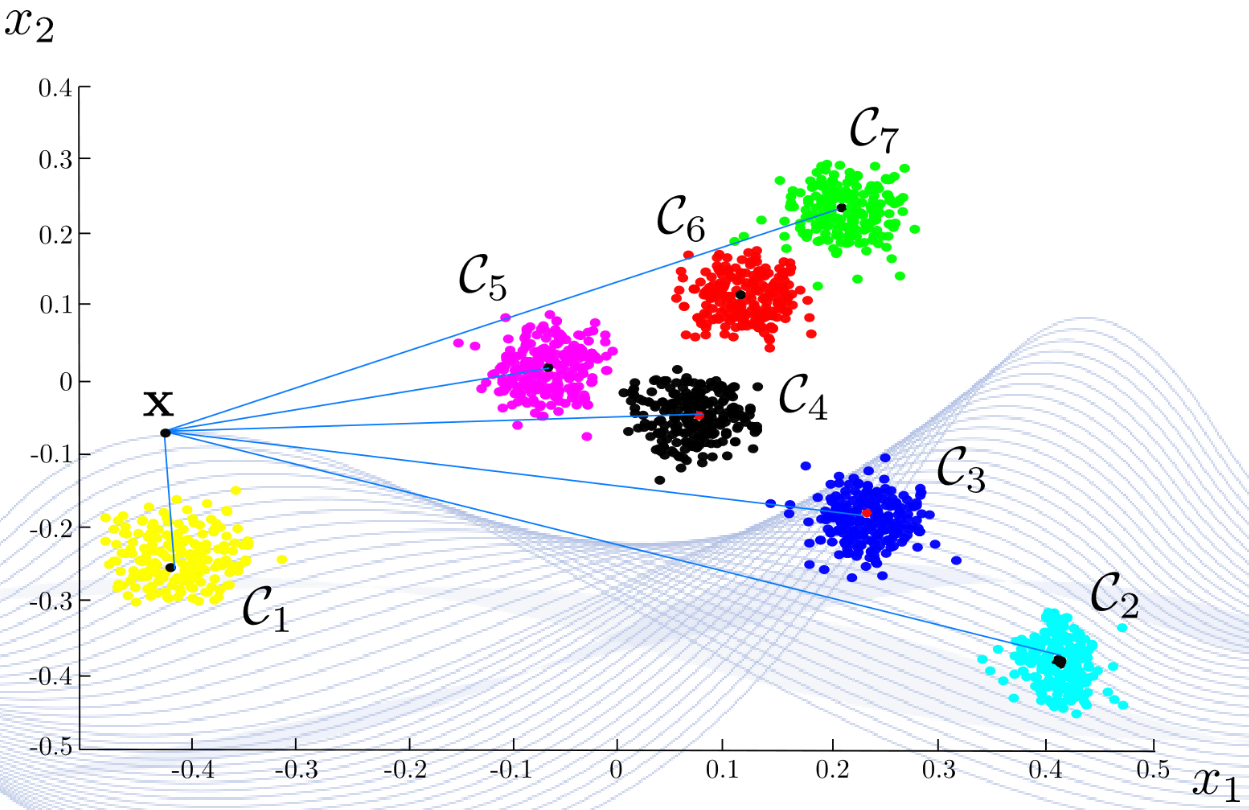
$$\mathrm{med}(d(\mathbf{m}_m,\mathbf{y})|\mathbf{y} \in \mathcal{C}) \leq \mathrm{med}(d(\mathbf{z},\mathbf{y})|\mathbf{y} \in \mathcal{C}),\ \mathbf{m}_m \in \mathcal{C},\ \forall\mathbf{z} \in \mathcal{C}.$$

  - med: median operator.

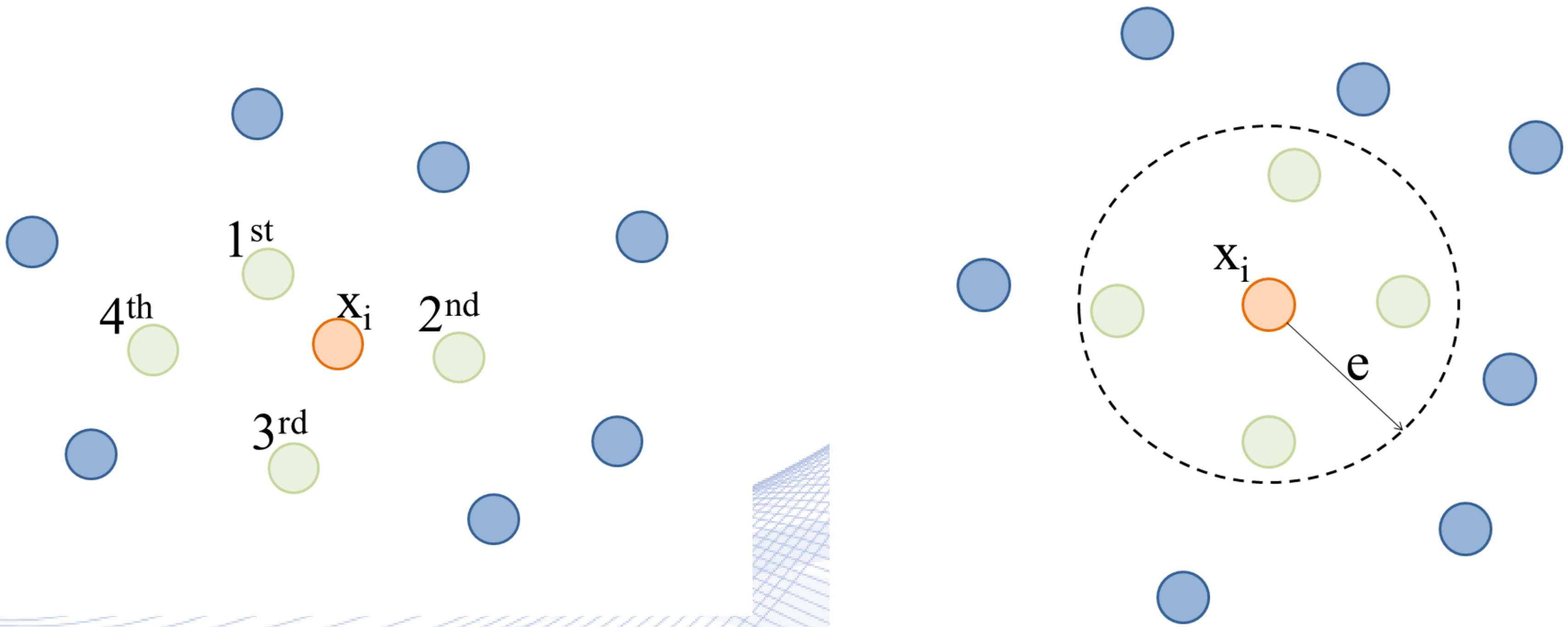
Nearest class classification

- A data point $\mathbf{x}$ is to be classified to one of the classes $\mathcal{C}_i, i = 1, \ldots, m$.
- A data class $\mathcal{C}_i$ is represented by a labeled data set: $\mathcal{C}_i = \{\mathbf{x}_{1i}, \mathbf{x}_{2i}, \ldots, \mathbf{x}_{Ni}\}$.
- Classify $\mathbf{x}$ to the closest class $\mathcal{C}$, by minimizing a distance $d'(\mathbf{x}, \mathcal{C})$.

Nearest class classification



Nearest neighbor graphs



a) k-nearest neighbor graph; b) e-neighborhood graph.

k-Nearest neighbor classification

- A data point $\mathbf{x}$ is to be classified to one of the classes $\mathcal{C}_i, i = 1, \ldots, m$.
- A data class $\mathcal{C}_i$ is represented by a labeled data set:

$$\mathcal{C}_i = \{\mathbf{x}_{1i}, \mathbf{x}_{2i}, \ldots, \mathbf{x}_{Ni}\}.$$

- Classify $\mathbf{x}$ to the class $\mathcal{C}$, whose data vectors are most common in the k –neighborhood of $\mathbf{x}$.

Outline

- Nearest neighbor classification
- **Supervised Learning Vector Quantization**

Supervised Learning Vector Quantization

- A data point $\mathbf{x}$ is to be classified to one of the classes $\mathcal{C}_i, i = 1, \dots, m$.
- A data class $\mathcal{C}_i$ is represented by a labeled data set: $\mathcal{C}_i = \{\mathbf{x}_{1i}, \mathbf{x}_{2i}, \dots, \mathbf{x}_{Ni}\}$.
- Each class is represented by a class center $\mathbf{m}_i, i = 1, \dots, m$.

Supervised Learning Vector Quantization

Supervised LVQ training

- $\mathbf{x}$: vector to be assigned to a class.
- Employ Euclidean distance.
- Find the optimal class centers $\mathbf{m}_i, i = 1, \dots, m$.
- Find the closest class center $\mathbf{m}_k$:

$$d(\mathbf{x}, \mathbf{m}_k) = \min_i \{d(\mathbf{x}, \mathbf{m}_i)\}, \forall i \neq k \, .$$

Supervised Learning Vector Quantization

- ***Winning class center updating***:

$$\mathbf{m}_k(t+1) = \mathbf{m}_k(t) + a(t)[\mathbf{x} - \mathbf{m}_k(t)]$$

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t), \qquad \text{for } i \neq k,$$

- $0 \leq a(t) \leq 1.$

Supervised Learning Vector Quantization

- Incremental algorithm: data may come on the fly.
- For the first steps, $a(t)$ value shall be close to 1.
- Depending on total number of steps, $a(t)$ decreases:
  - Linear, exponential decrease.

- When $a(t)$ falls below the threshold, the algorithm freezes.

Supervised Learning Vector Quantization

Competition during training:

- If $\mathcal{C}_i$ is the closest cluster to $\mathbf{x}$, but $\mathcal{C}_j$ is the correct cluster ($\mathcal{C}_j \neq \mathcal{C}_i$):

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) - a(t)[\mathbf{x}(t) - \mathbf{m}_i(t)],$$

$$\mathbf{m}_j(t+1) = \mathbf{m}_j(t) + a(t)[\mathbf{x}(t) - \mathbf{m}_j(t)].$$

  - For all other clusters: $\mathbf{m}_k(t+1) = \mathbf{m}_k(t)$.

LVQ testing:

- Classify $\mathbf{x}$ to the closest class $\mathcal{C}_k$, by minimizing $d(\mathbf{x}, \mathbf{m}_k)$.

Q & A

Thank you very much for your attention!

More material in
http://icarus.csd.auth.gr/cvml-web-lecture-series/

Contact: Prof. I. Pitas
pitas@csd.auth.gr