Machine Learning I
80-629A

Apprentissage Automatique I
80-629

Unsupervised Learning
— Week #7

Today

- Unsupervised learning
  - Definition and properties
  - A few clustering models
  - Other instantiations of unsupervised learning

Unsupervised Learning

Recall

Experience (E)

- What data does f experience?
  - (Focus on algorithms that experience whole datasets)
  - **Unsupervised.** Examples alone $\{\mathbf{x}_i\}_{i=0}^{n}$
  - **Supervised.** Examples come with labels $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=0}^{n}$

Recall

1. Unsupervised

$$\{x_i\}_{i=0}^{n}$$

- Experience examples alone
- Learn “useful properties of the structure of the data”
  - E.g., clustering, density modeling (p(x)), PCA, FA.

Different tasks

- Finding patterns
  - Clustering $f : X \to \{1, 2, \ldots, K\}$ (K clusters)
  - Dimensionality reduction $f : X^p \to X^k, k << p$
  - Density modelling $f : X \to [0, 1]$
  - ...

Why unsupervised learning?

1. Understand properties of the data
2. Learn useful representations
3. Use the results in a downstream application

- Opportunity: There are lots of unlabelled data

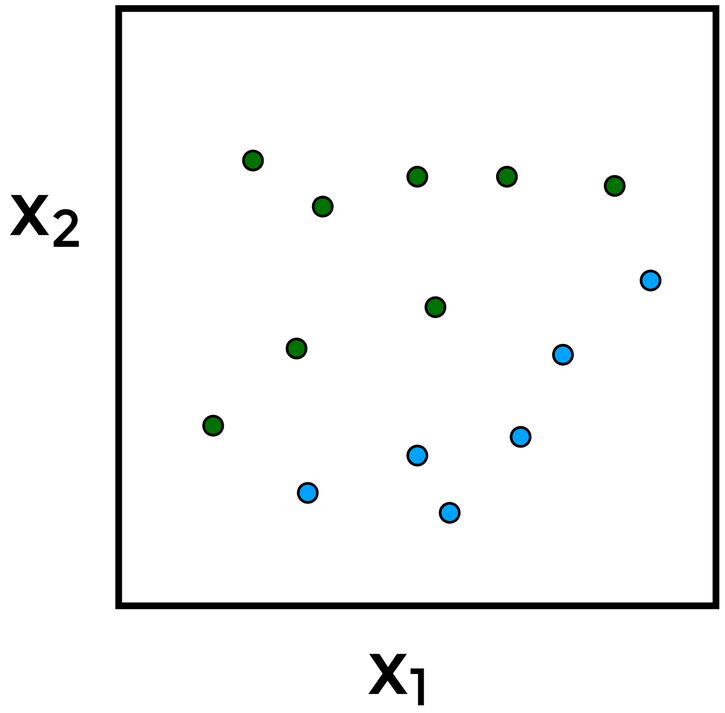
k-means clustering

Clustering

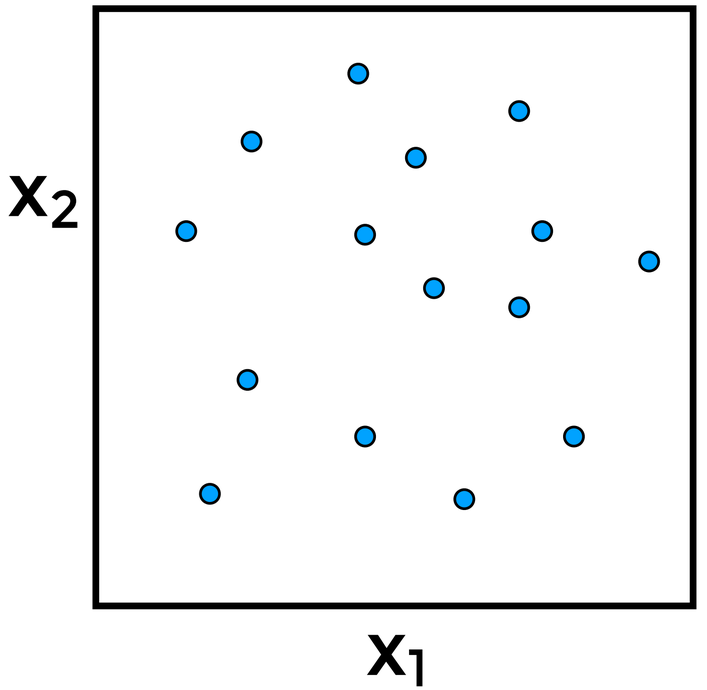
$$f : X \to \{1, \ldots, K\}$$

- Task: Assign each point to one of K clusters
  - Cluster: a set of similar points (a group)
- Alternatively: Divide the space into K regions. Assign points to a cluster based on the region they lie in
  - Similar to classification.

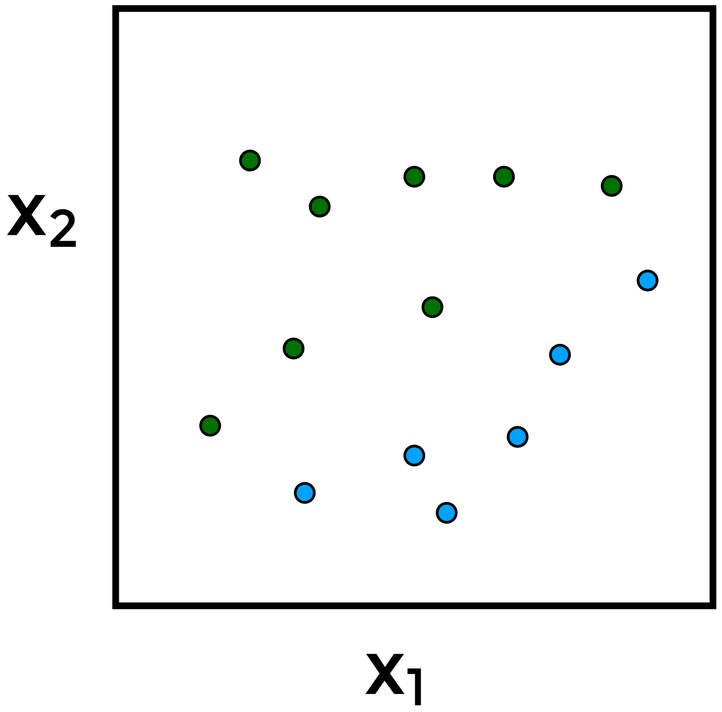
Supervised



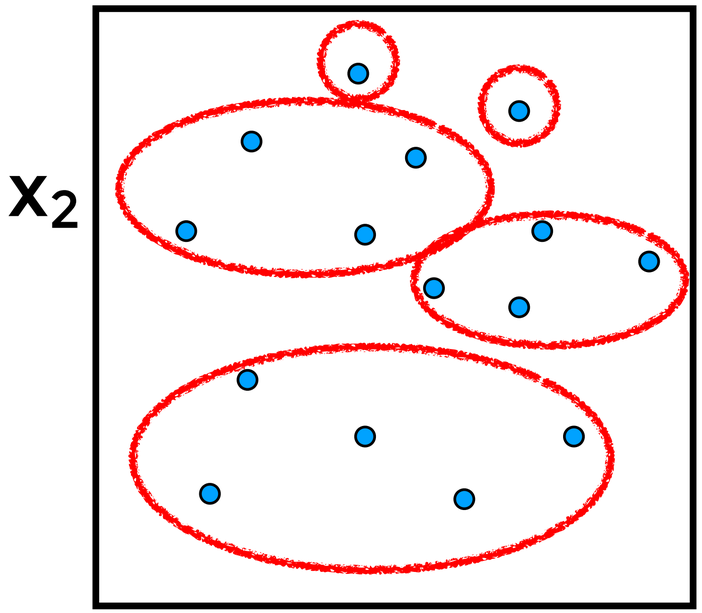
Unsupervised



Supervised



Unsupervised

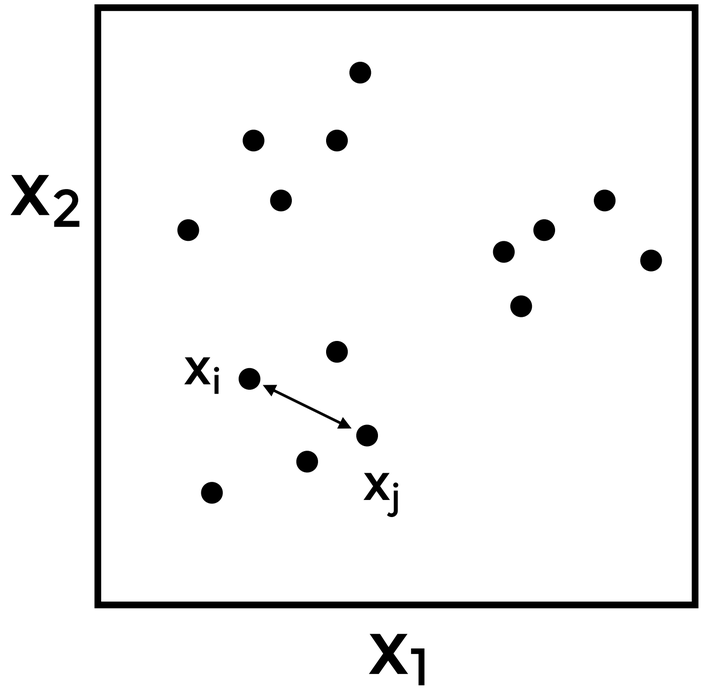


Clustering

Clustering

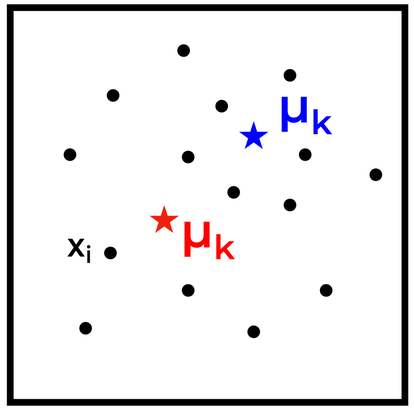
- Desideratum: group similar points together
- Similarity is often defined as being close according to some similarity or (inverse) distance function
  - E.g., in Euclidean space

$$\text{distance}(x_i, x_j) = \| x_i - x_j \|_2^2$$



K-means clustering

- A particular clustering model (and accompanying algorithm)
  - There are K clusters. Each point belongs to a cluster. Clusters have centers: μ
- Objective: Find cluster centers μ_k that minimize the within cluster distance

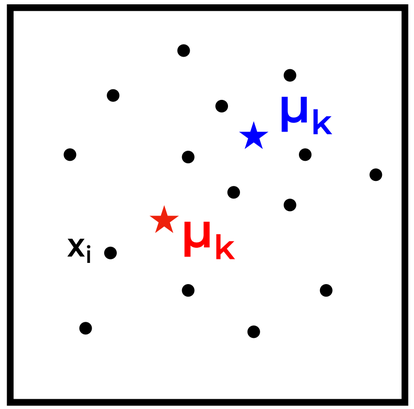


K-means clustering

- A particular clustering model (and accompanying algorithm)
  - There are K clusters. Each point belongs to a cluster. Clusters have centers: μ
- Objective: Find cluster centers μ_k that minimize the within cluster distance

$$\text{Objective} := \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} ||x_i - \mu_k||^2$$

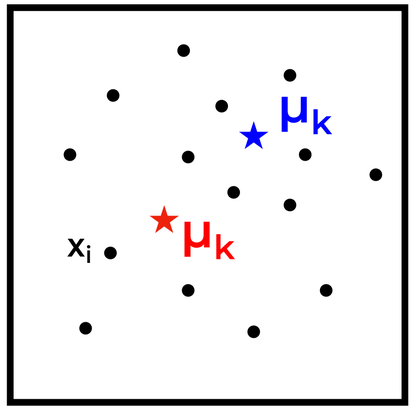
$$r = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ \vdots & \vdots \\ 0 & 1 \end{bmatrix}_{N \times 2}$$



K-means clustering

- A particular clustering model (and accompanying algorithm)
  - There are K clusters. Each point belongs to a cluster. Clusters have centers: μ
- Objective: Find cluster centers μ_k that minimize the within cluster distance

$$\text{Objective} := \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} ||x_i - \mu_k||^2$$



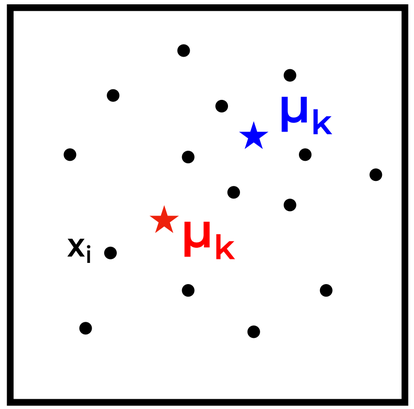
- Algorithm to minimize the objective:
  - Initialize the cluster centers
  - Until convergence:
    1. Update responsibilities: r
    2. Update cluster centers: $\mu_k \quad \forall k$

$$r = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ \vdots & \vdots \\ 0 & 1 \end{bmatrix}_{N \times 2}$$

K-means clustering

- A particular clustering model (and accompanying algorithm)
  - There are K clusters. Each point belongs to a cluster. Clusters have centers: μ
- Objective: Find cluster centers μ_k that minimize the within cluster distance

$$\text{Objective} := \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} ||x_i - \mu_k||^2$$



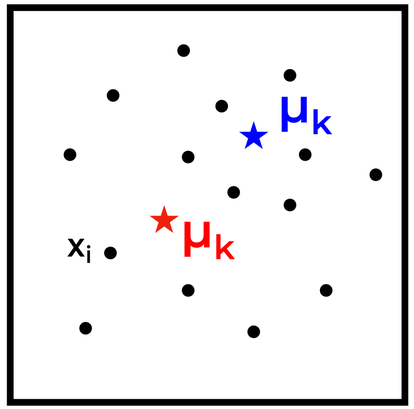
- Algorithm to minimize the objective:
  - Initialize the cluster centers
  - Until convergence:
    1. Update responsibilities: r
    2. Update cluster centers: $\mu_k \quad \forall k$

$$r = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ \vdots & \vdots \\ 0 & 1 \end{bmatrix}_{N \times 2}$$

K-means clustering

- A particular clustering model (and accompanying algorithm)
  - There are K clusters. Each point belongs to a cluster. Clusters have centers: μ
- Objective: Find cluster centers μ_k that minimize the within cluster distance

$$\text{Objective} := \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} ||x_i - \mu_k||^2$$



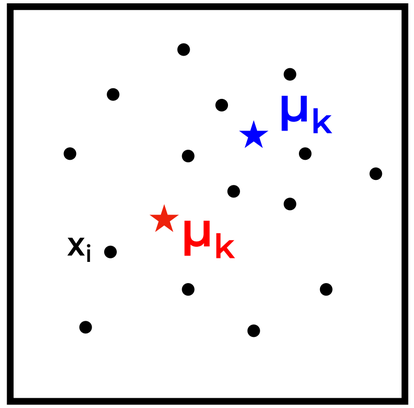
- Algorithm to minimize the objective:
  - Initialize the cluster centers
  - Until convergence:
    1. Update responsibilities: r
    2. Update cluster centers: $\mu_k \quad \forall k$

$$r = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ \vdots & \vdots \\ 0 & 1 \end{bmatrix}_{N \times 2}$$

K-means clustering

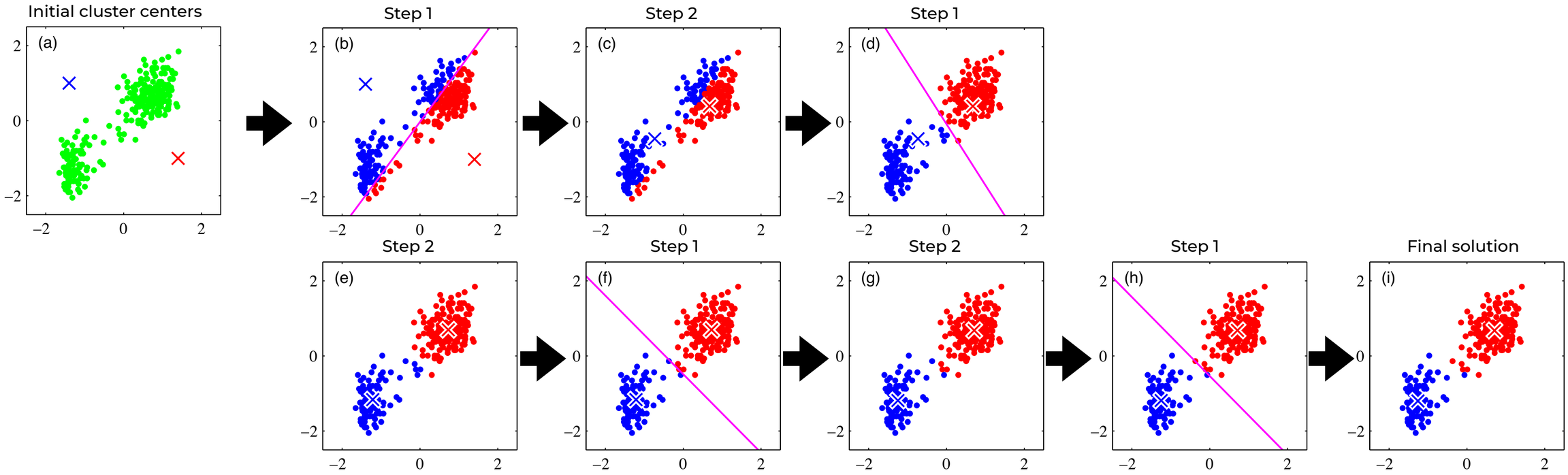
- A particular clustering model (and accompanying algorithm)
  - There are K clusters. Each point belongs to a cluster. Clusters have centers: μ
- Objective: Find cluster centers μ_k that minimize the within cluster distance

$$\text{Objective} := \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} ||x_i - \mu_k||^2$$



- Algorithm to minimize the objective:
  - Initialize the cluster centers
  - Until convergence:
    1. Update responsibilities: r
    2. Update cluster centers: $\mu_k \quad \forall k$

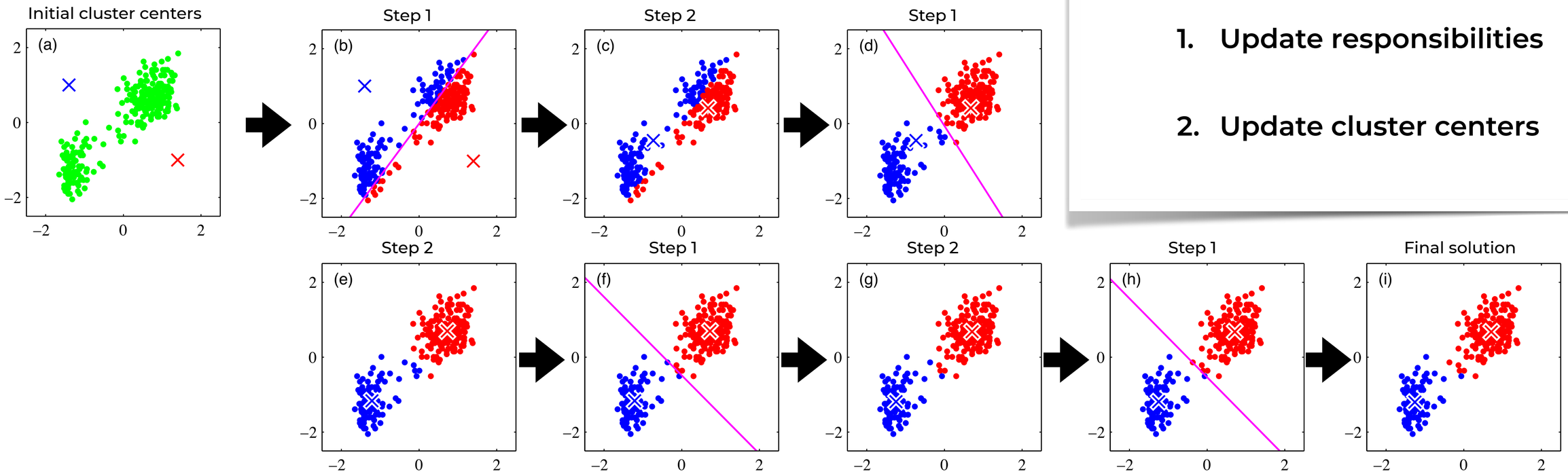
$$r = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ \vdots & \vdots \\ 0 & 1 \end{bmatrix}_{N \times 2}$$



[Figure 9.1 from PRML]

Algorithm

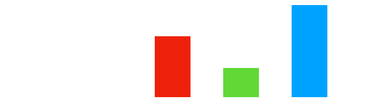
- Initialize the cluster centers
- Until convergence:
  1. Update responsibilities
  2. Update cluster centers



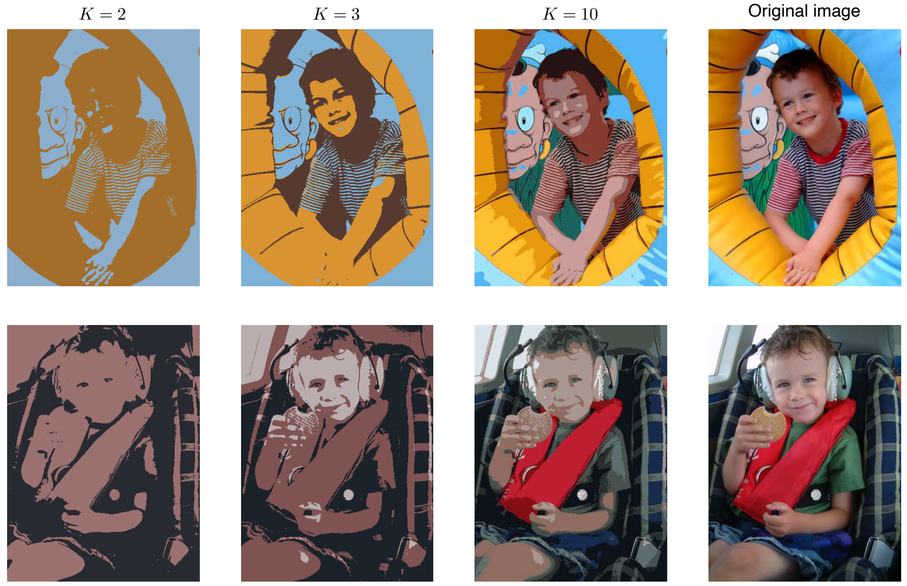
[Figure 9.1 from PRML]

K-means on images

Each pixel is a datum



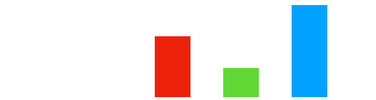
$$\mathbf{x}_i = (\mathbf{r}_i, \mathbf{g}_i, \mathbf{b}_i)$$



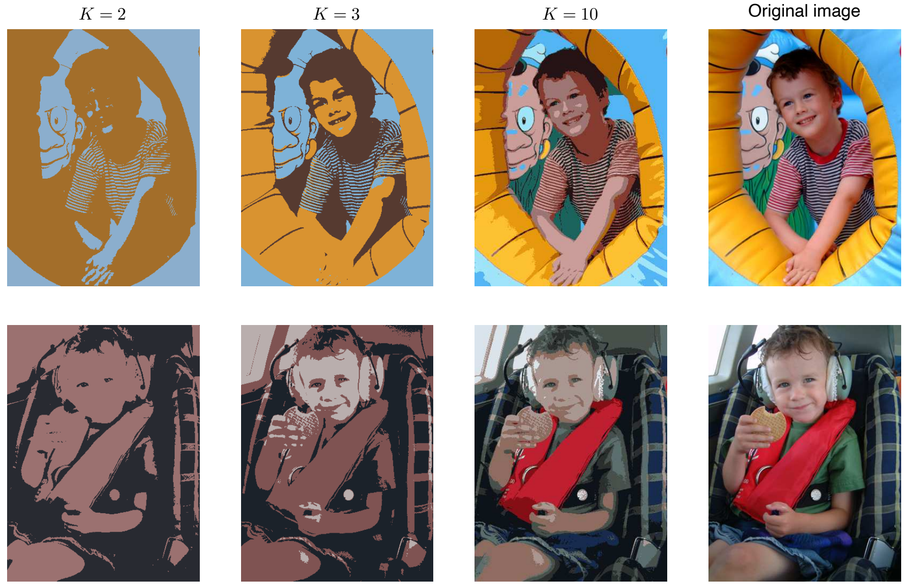
[Figure 9.3 from PRML]

K-means on images

Each pixel is a datum



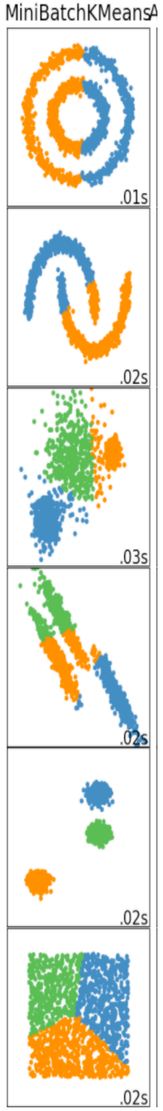
$$\mathbf{x}_i = (\mathbf{r}_i, \mathbf{g}_i, \mathbf{b}_i)$$



- We cluster pixels by color
  - I.e., pixels of similar color will be belong to the same cluster
- We re-draw the image by replacing the color of each pixel by the color of its cluster

[Figure 9.3 from PRML]

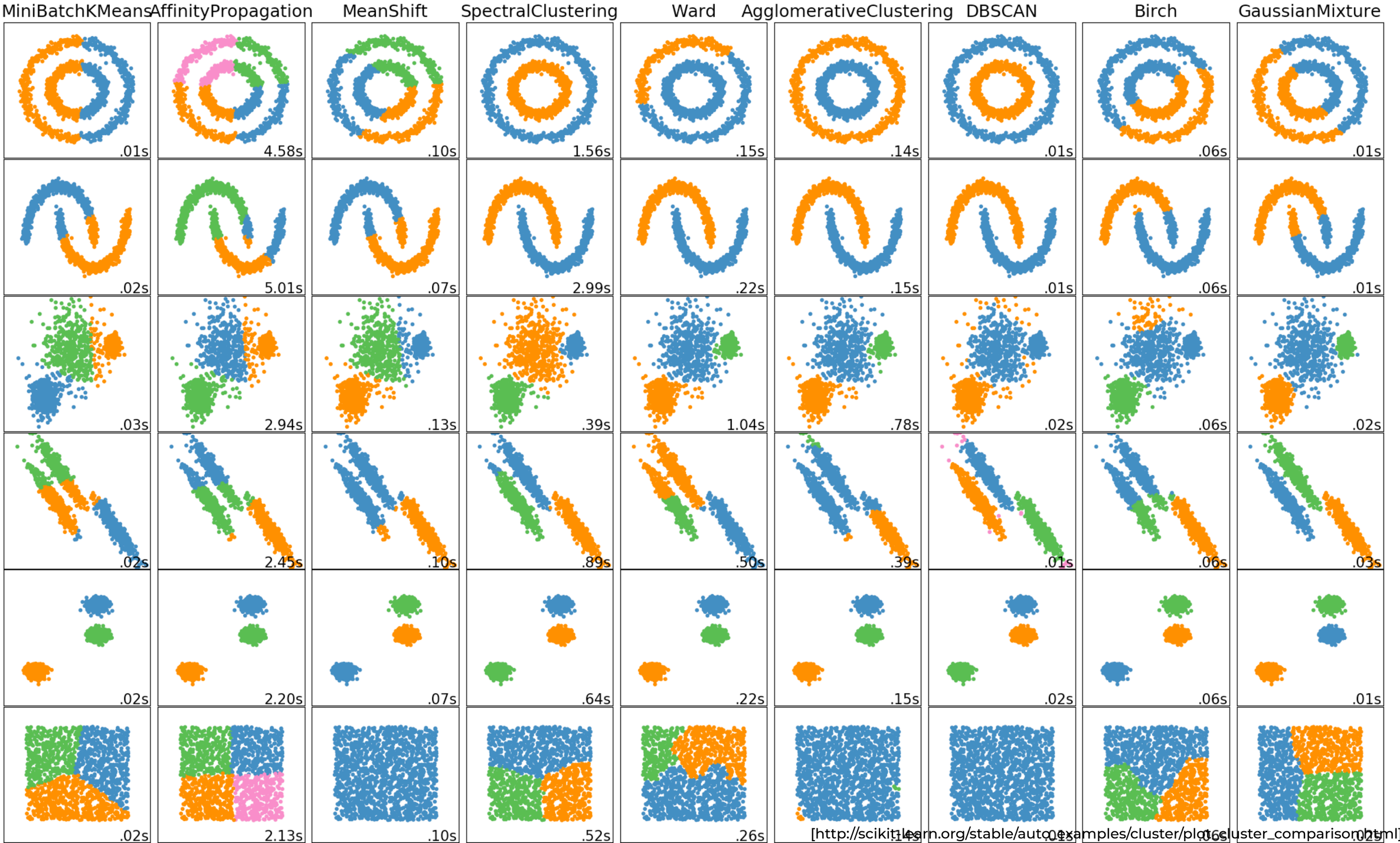
A few failure cases of k-means



- Only works with squared Euclidean distance
  - What about non-continuous data?

    K-medoids Objective $:= \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} d(x_i, \mu_k)$
  - Not robust to outliers
- Tends to result in relatively uniform cluster sizes
- Hard assignments

[http://scikit-learn.org/stable/auto_examples/cluster/plot_cluster_comparison.html]



[http://scikit-learn.org/stable/auto_examples/cluster/plot_cluster_comparison.html]

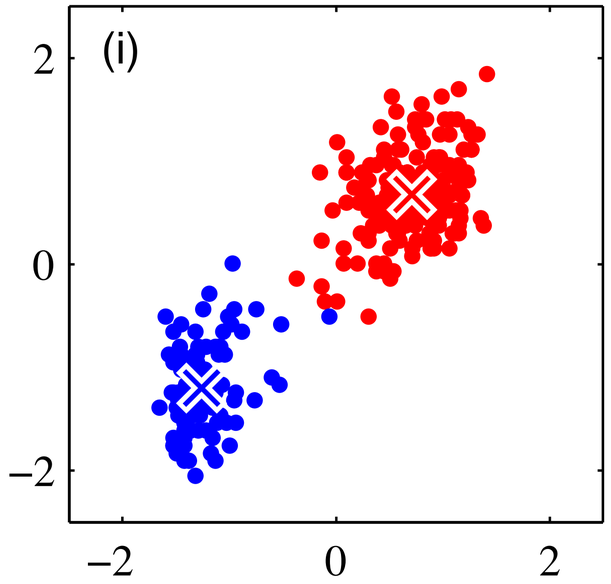
Unsupervised learning

- In supervised learning you have:
  - Clear metric (e.g., mean squared error, accuracy, etc.)
  - Procedures for comparing different models
- Unsupervised learning
  - Unclear what the right metric is, domain dependant
  - How to compare different models?
- Data dimensions can have a big impact on solution
  - You should carefully select the input

Gaussian Mixture for Probabilistic Clustering

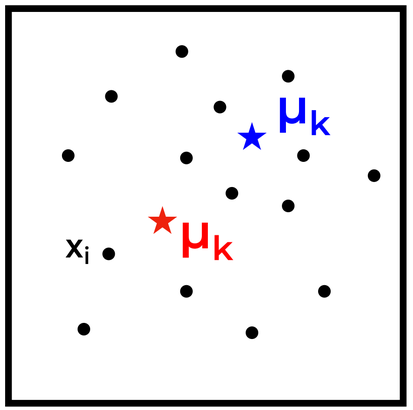
Motivation

Clustering found
using K-means



A probabilistic approach to k-means clustering

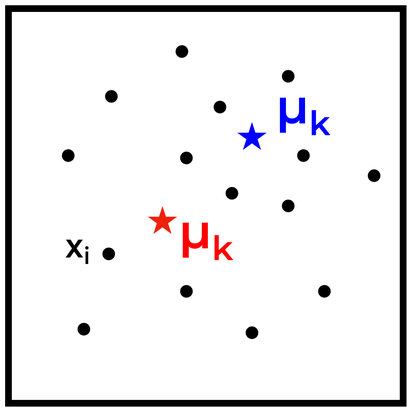
K-means Clustering



$$\text{Objective} := \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} \parallel x_i - \mu_k \parallel^2$$

A probabilistic approach to k-means clustering

K-means Clustering

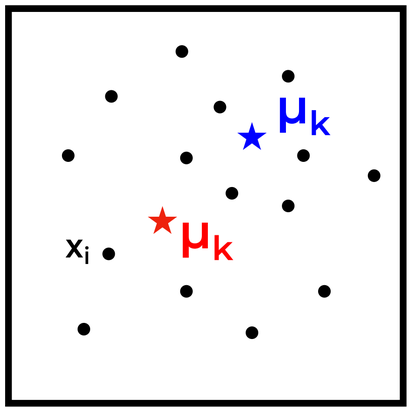


Responsibility center

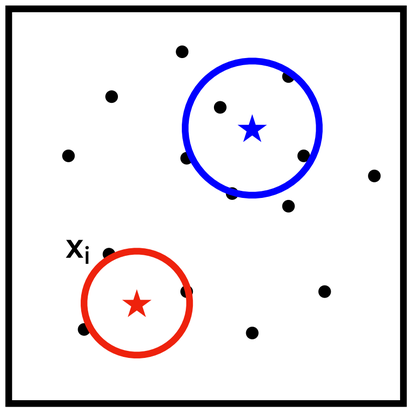
$$\text{Objective} := \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} \| x_i - \mu_k \|^2$$

A probabilistic approach to k-means clustering

K-means Clustering



Soft K-means Clustering

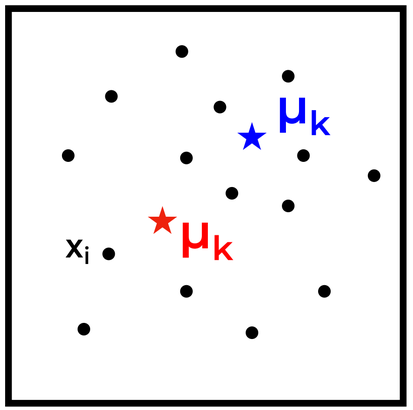


Responsibility center

$$\text{Objective} := \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} \| x_i - \mu_k \|^2$$

A probabilistic approach to k-means clustering

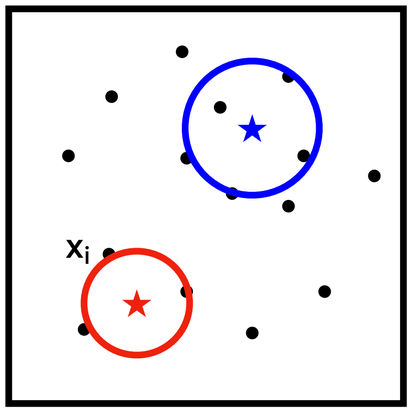
K-means Clustering



Responsibility center

$$\text{Objective} := \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} \| x_i - \mu_k \|^2$$

Soft K-means Clustering



- Responsibilities are continuous [0, 1]
  - Each cluster has a responsibility: π_k
- Each cluster models data using a Gaussian: $\mathcal{N}(x_i \mid \mu_k, \Sigma_k)$

Soft K-means Clustering

- **Each cluster gives a probability to each datum:** $\mathcal{N}(\mathbf{x_i} \mid \boldsymbol{\mu_k}, \Sigma_\mathbf{k})$

Soft K-means Clustering

- Each cluster gives a probability to each datum: $\mathcal{N}(x_i \mid \mu_k, \Sigma_k)$
- We can write the complete model as:

$$P(x_i \mid \text{parameters}) = \sum_{k=1}^{K} \pi_k P(x_i \mid \mu_k, \Sigma_k)$$

Soft K-means Clustering

- Each cluster gives a probability to each datum: $\mathcal{N}(x_i \mid \mu_k, \Sigma_k)$
- We can write the complete model as:

$$P(x_i \mid \text{parameters}) = \sum_{k=1}^{K} \pi_k P(x_i \mid \mu_k, \Sigma_k)$$

$$0 \leq \pi \leq 1$$

$$\sum_k \pi_k = 1$$

Soft K-means Clustering

- Each cluster gives a probability to each datum: $\mathcal{N}(x_i \mid \mu_k, \Sigma_k)$
- We can write the complete model as:

$$P(x_i \mid \text{parameters}) = \sum_{k=1}^{K} \pi_k P(x_i \mid \mu_k, \Sigma_k)$$

$$0 \leq \pi \leq 1$$

$$\sum_k \pi_k = 1$$

$$P(x_i \mid \text{parameters}) = \sum_{k=1}^{K} P(z_i = k) P(x_i \mid \mu_k, \Sigma_k)$$

Soft K-means Clustering

- Each cluster gives a probability to each datum: $\mathcal{N}(x_i \mid \mu_k, \Sigma_k)$
- We can write the complete model as:

$$P(x_i \mid \text{parameters}) = \sum_{k=1}^{K} \pi_k P(x_i \mid \mu_k, \Sigma_k)$$

$$0 \leq \pi \leq 1$$

$$\sum_k \pi_k = 1$$

$$P(x_i \mid \text{parameters}) = \sum_{k=1}^{K} P(z_i = k) P(x_i \mid \mu_k, \Sigma_k)$$

Posterior: $P(z_i = k \mid x_i) \propto P(z_i = k) P(x_i \mid \mu_k, \Sigma_k)$

Soft K-means Clustering

- Each cluster gives a probability to each datum: $\mathcal{N}(x_i \mid \mu_k, \Sigma_k)$
- We can write the complete model as:

$$P(x_i \mid \text{parameters}) = \sum_{k=1}^{K} \pi_k P(x_i \mid \mu_k, \Sigma_k)$$

$$0 \leq \pi \leq 1$$

$$\sum_k \pi_k = 1$$

$$P(x_i \mid \text{parameters}) = \sum_{k=1}^{K} P(z_i = k) P(x_i \mid \mu_k, \Sigma_k)$$

Posterior: $P(z_i = k \mid x_i) \propto P(z_i = k) P(x_i \mid \mu_k, \Sigma_k)$

- Model is known as a **Gaussian Mixture Model**

Soft K-means Clustering

- Each cluster gives a probability to each datum: $\mathcal{N}(x_i \mid \mu_k, \Sigma_k)$
- We can write the complete model as:

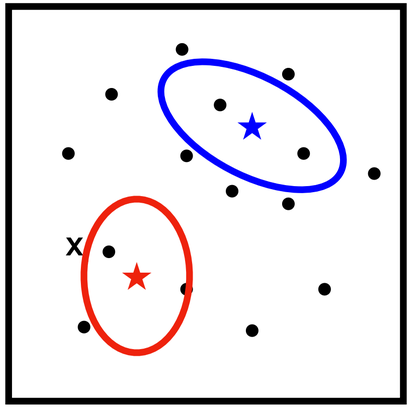
$$P(x_i \mid \text{parameters}) = \sum_{k=1}^{K} \pi_k P(x_i \mid \mu_k, \Sigma_k)$$

$$0 \leq \pi \leq 1 \qquad \sum_k \pi_k = 1$$

$$P(x_i \mid \text{parameters}) = \sum_{k=1}^{K} P(z_i = k) P(x_i \mid \mu_k, \Sigma_k)$$

Posterior: $P(z_i = k \mid x_i) \propto P(z_i = k) P(x_i \mid \mu_k, \Sigma_k)$

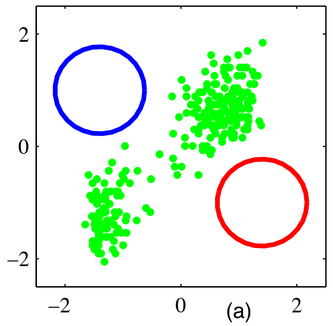
- Model is known as a **Gaussian Mixture Model**
- Advantages over k-means:
  - Soft Clustering
  - Clusters don't have to be spherical

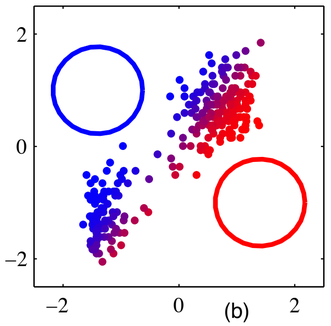


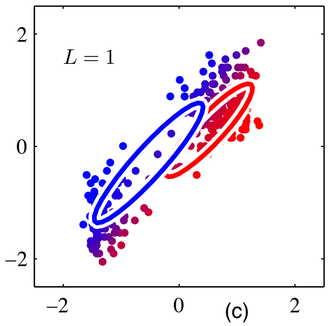
A probabilistic approach to k-means clustering

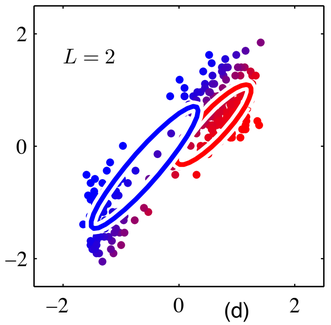
- Estimation of the parameters: Maximum likelihood estimate
  - Could do it jointly (“a la” neural network)
  - Often in two separate steps (similar as the non-probabilistic version)
    - This leads to the Expectation-Maximization (EM) algorithm

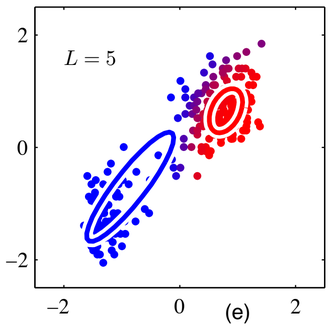
Initial cluster centers



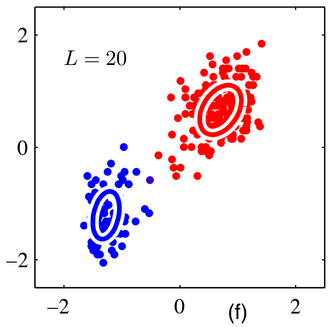






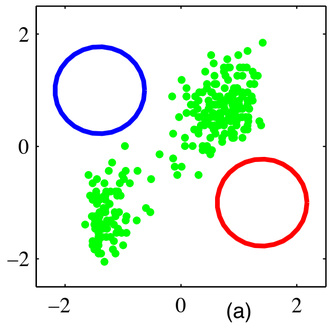


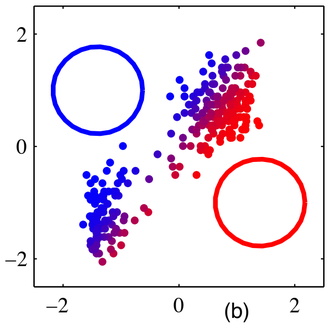
Final solution

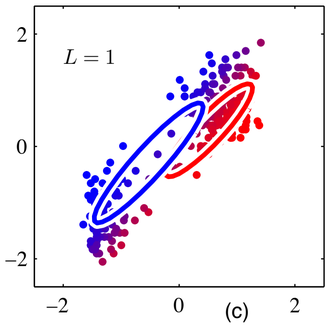


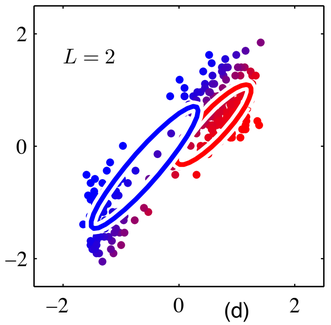
[Figure 9.8 from PRML]

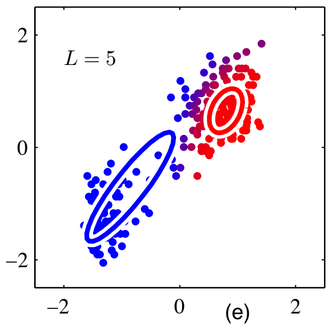
Initial cluster centers



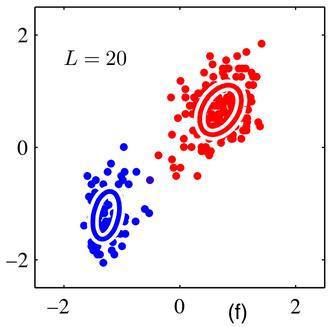




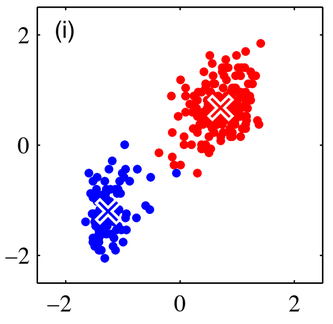




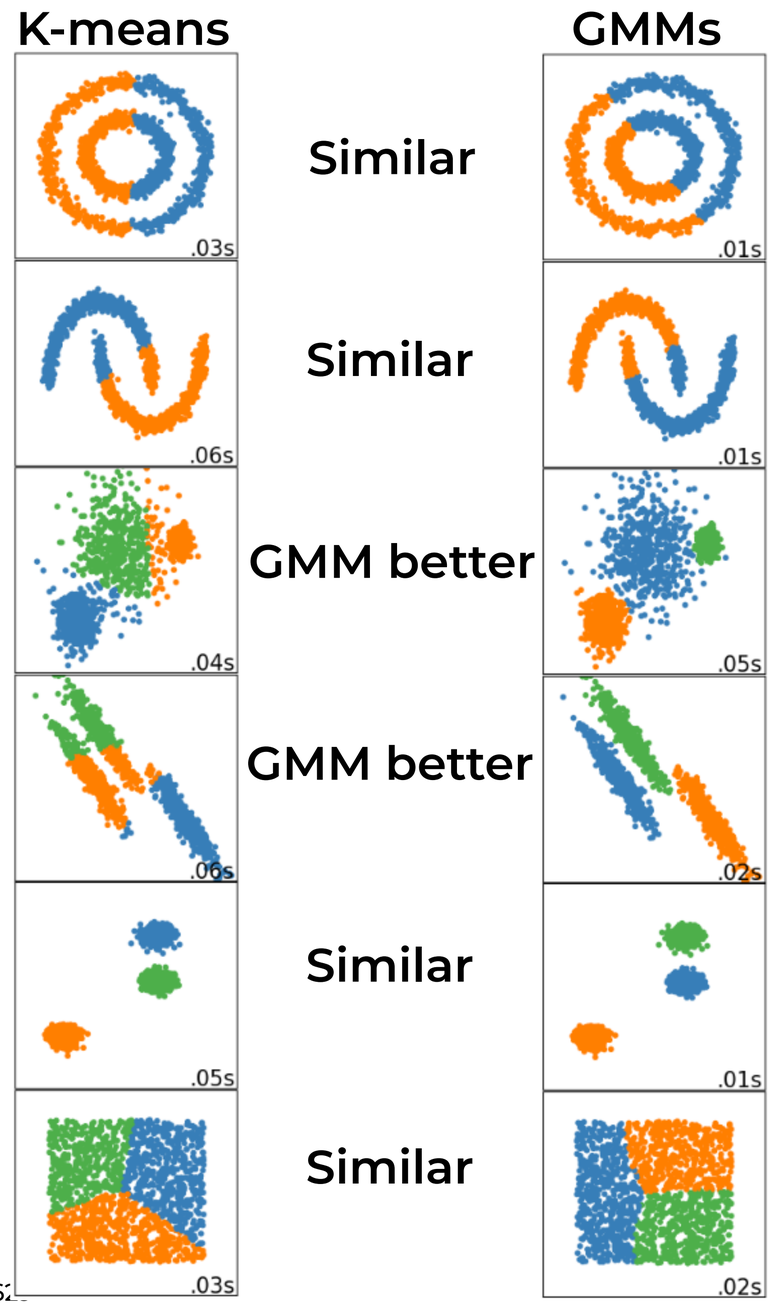
Final solution

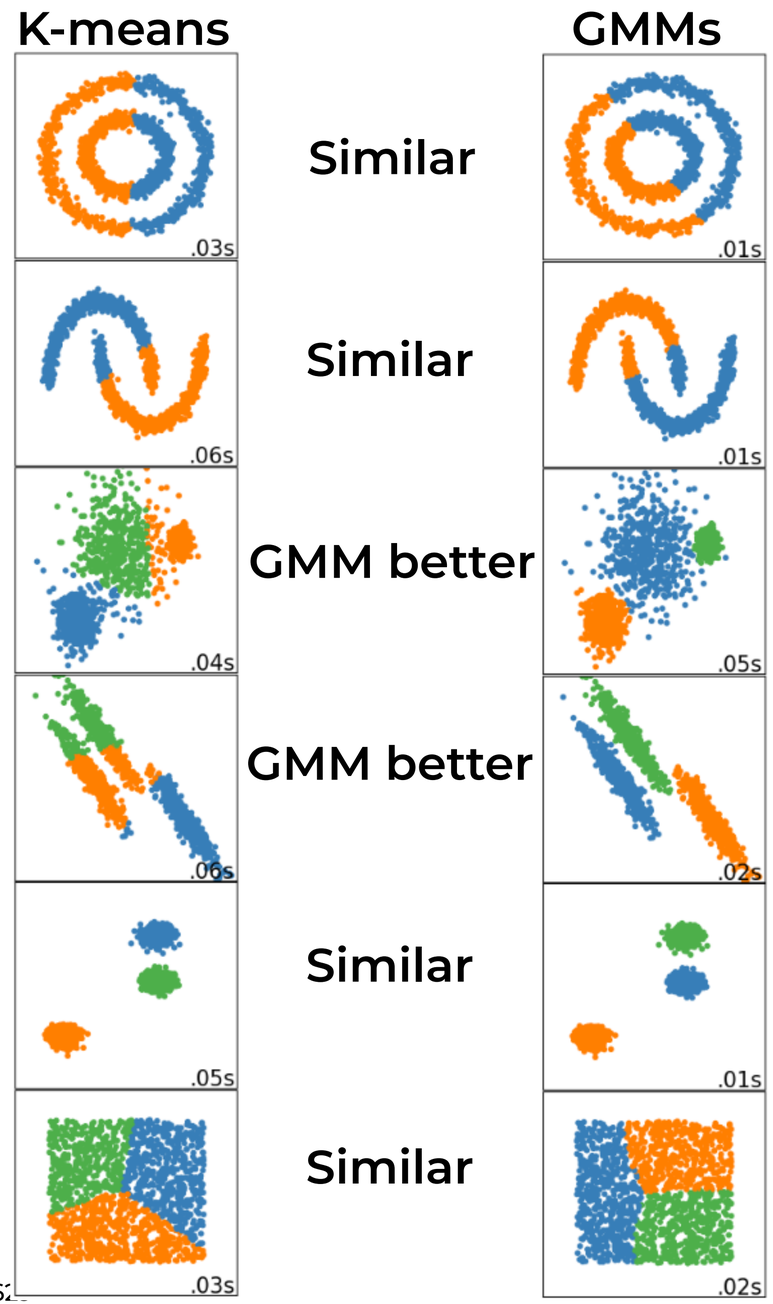


k-means
Final solution



[Figure 9.8 from PRML]


K-means
GMMs
Similar
.03s
.01s
Similar
.06s
.01s
GMM better
.04s
.05s
GMM better
.06s
.02s
Similar
.05s
.01s
Similar
.03s
.02s



Comparing K-means to GMMs

- GMMs learns covariance matrix
  - Per cluster variance
  - Covariance terms
- GMMs has many more parameters
  - Covariance matrix (MxM)

Unsupervised Learning Beyond Clustering

Different tasks

- Finding patterns
  - Clustering $f : X \to \{1, 2, \ldots, K\}$ (K clusters)
  - Dimensionality reduction $f : X^p \to X^k, k << p$
  - Density modelling $f : X \to [0, 1]$
  - ...

Different tasks

- Finding patterns
  - Clustering $f : X \to \{1, 2, \ldots, K\}$ (K clusters)
  - Dimensionality reduction $f : X^p \to X^k, k << p$
  - Density modelling $f : X \to [0, 1]$
  - ...

Autoencoders ⟶

Autoencoders

- A type of neural network for dimensionality reduction
  - Non-linear PCA

Autoencoders

- A type of neural network for dimensionality reduction
  - Non-linear PCA
  - Intuition: let's learn to copy the data

$$\mathbf{x} = \mathbf{f}(\mathbf{x})$$

Autoencoders

- A type of neural network for dimensionality reduction
  - Non-linear PCA
  - Intuition: let's learn to copy the data

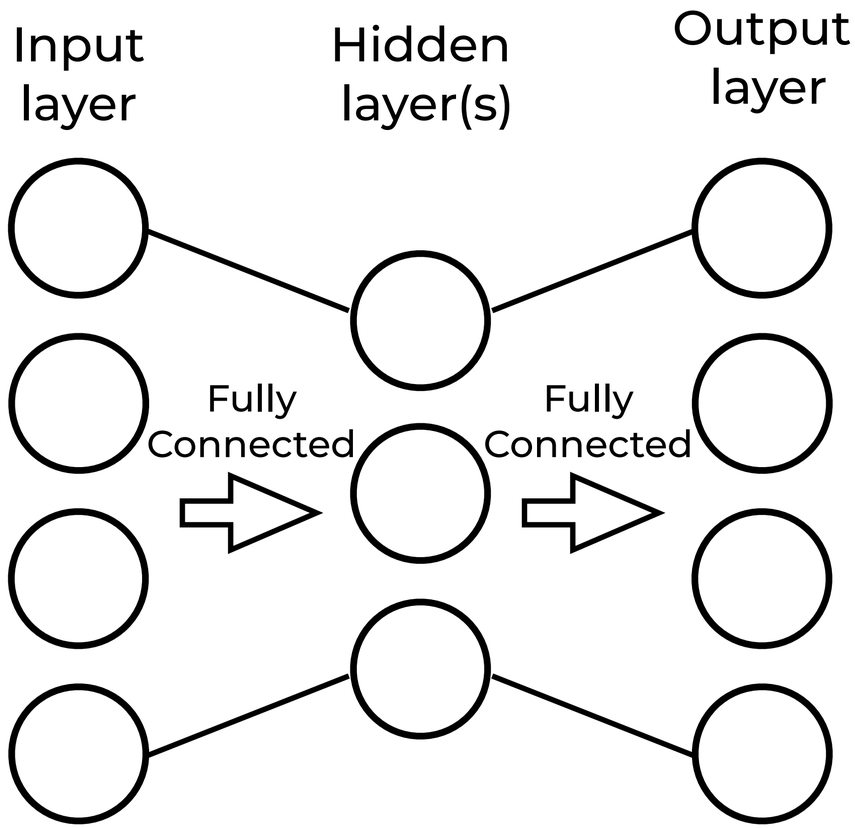
$$\mathbf{x} = \mathbf{f}(\mathbf{x})$$

  - We force a "bottleneck"

$$\mathbf{z} = \mathbf{f}_1(\mathbf{x})$$
$$\mathbf{x} = \mathbf{f}_2(\mathbf{z})$$
$$\dim(\mathbf{z}) < \dim(\mathbf{x})$$

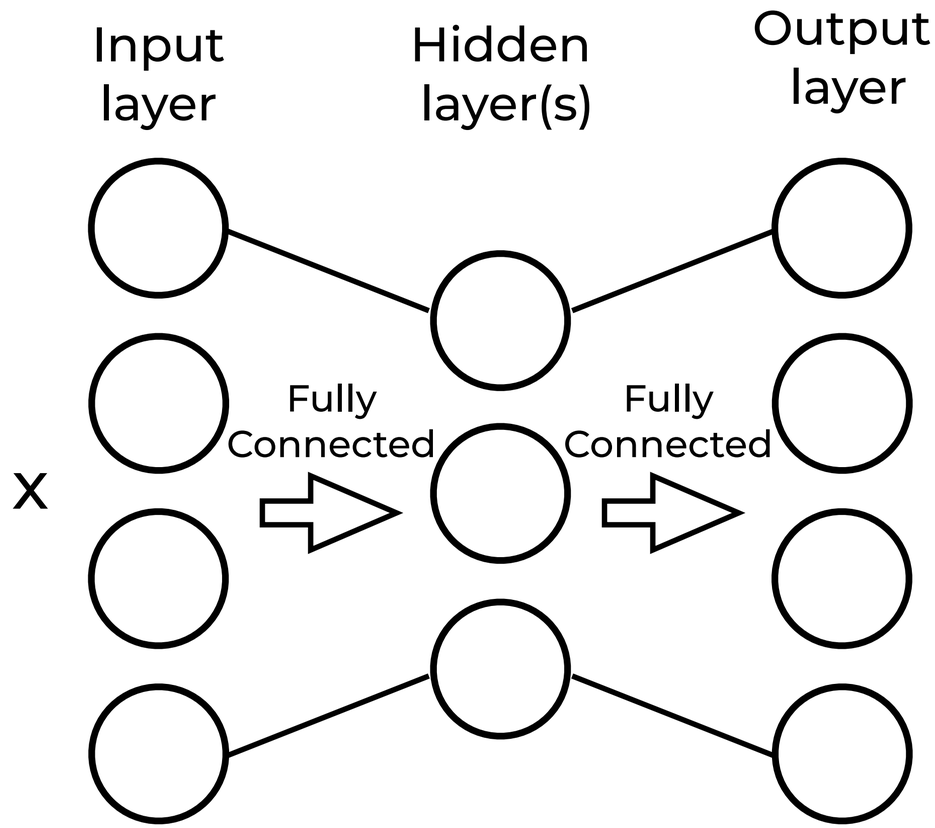
Autoencoders

- **A neural network architecture for unsupervised learning**



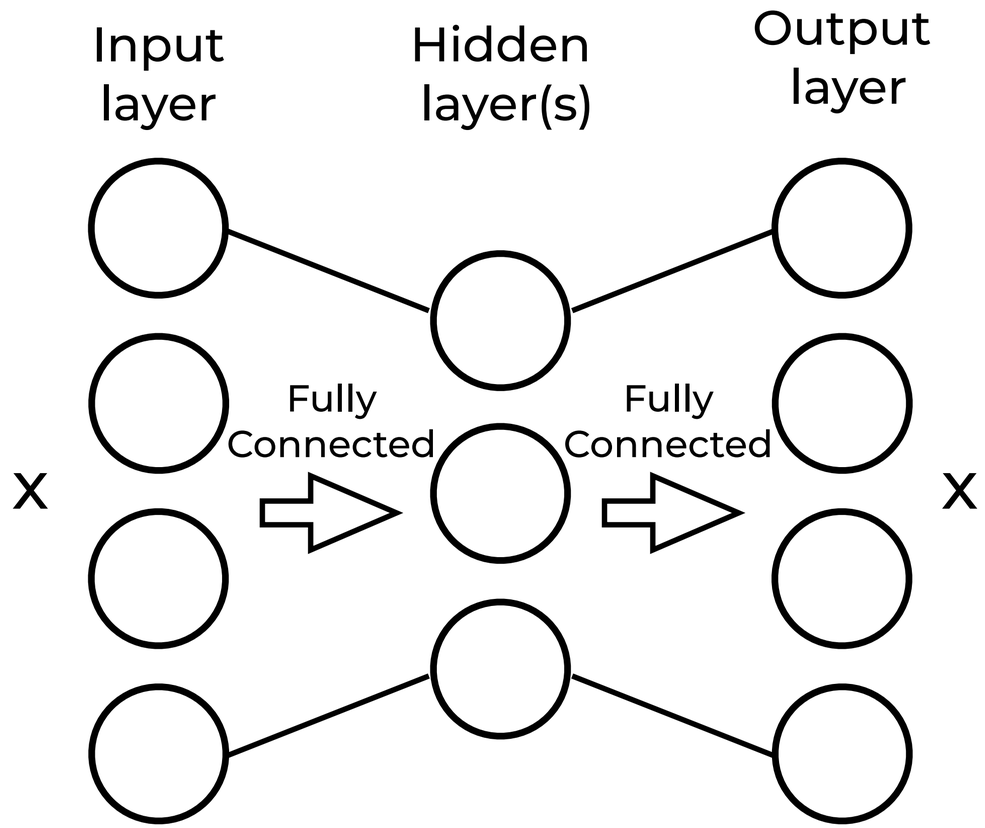
Autoencoders

- **A neural network architecture for unsupervised learning**



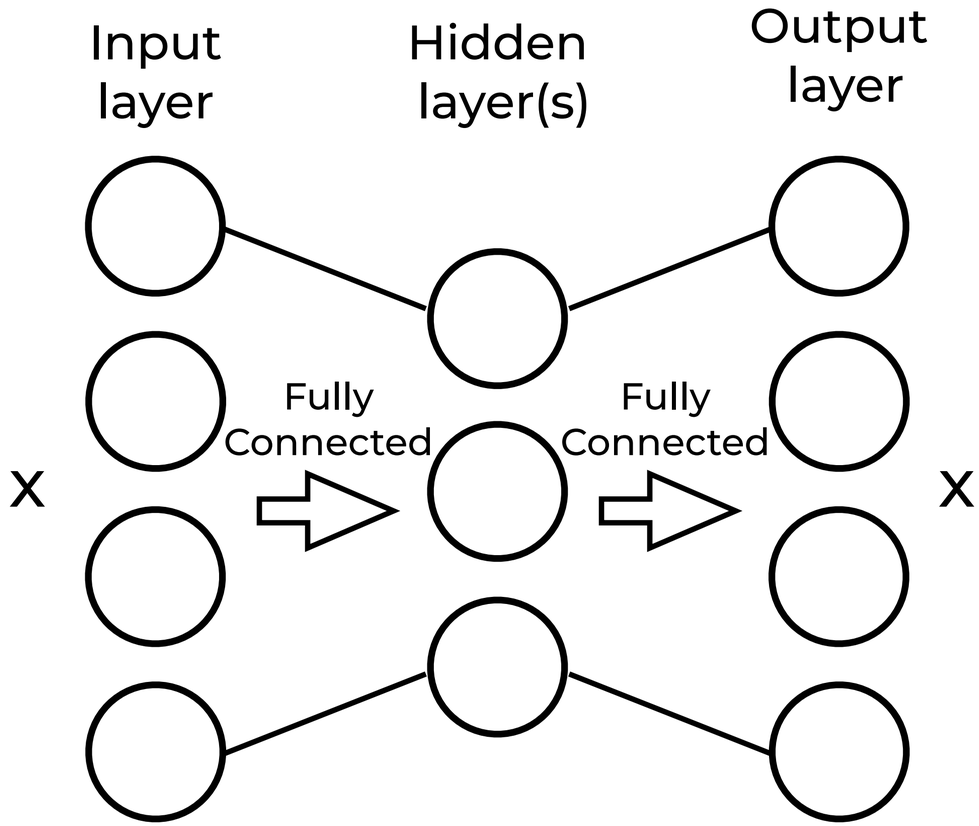
Autoencoders

- **A neural network architecture for unsupervised learning**



Autoencoders

- **A neural network architecture for unsupervised learning**



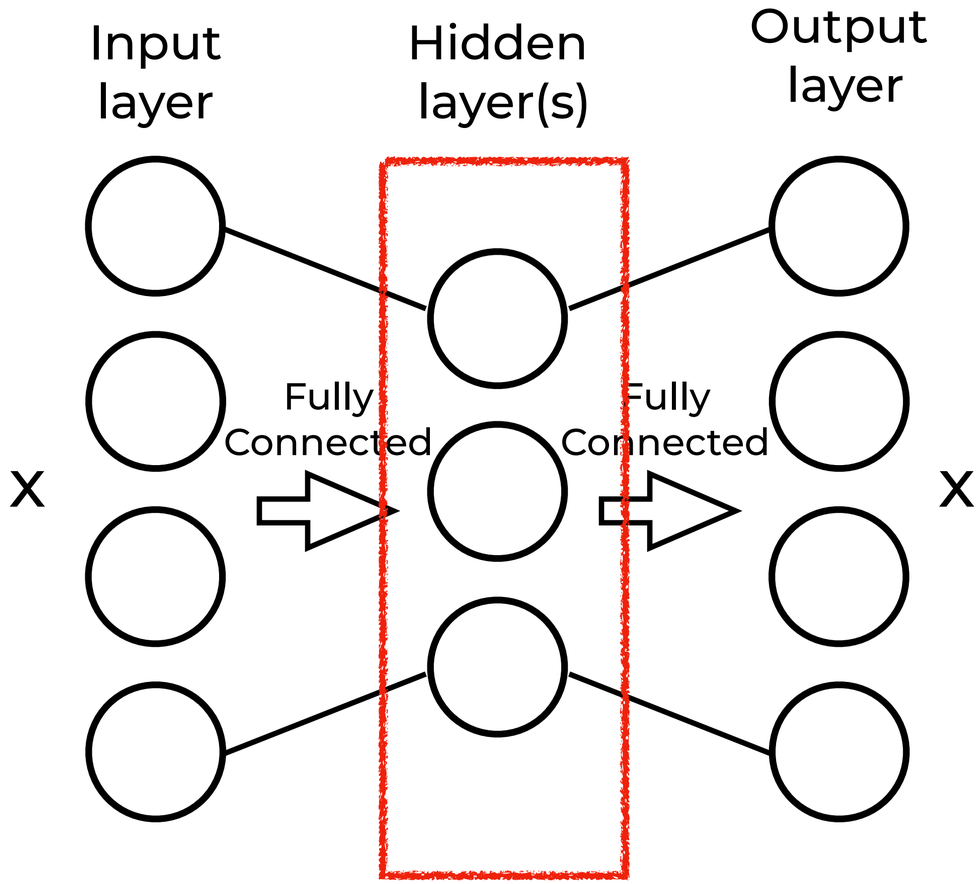
Objective:
How well the network predicts X ?

$$\textbf{Loss} := \sum_{i=1}^{N} (x_i - \hat{x}_i)^2$$

$$= \sum_{i=1}^{N} (x_i - f_2(f_1(x)))^2$$

Autoencoders

- A neural network architecture for unsupervised learning



Lower dimensional representation

Objective:
How well the network predicts X ?

$$\text{Loss} := \sum_{i=1}^{N} (x_i - \hat{x}_i)^2$$

$$= \sum_{i=1}^{N} (x_i - f_2(f_1(x)))^2$$

Unsupervised learning as supervised learning

- Examples:
  - Auto-encoders “predict” their inputs
  - Language models “predict” the next word

Unsupervised learning as supervised learning

- Examples:
  - Auto-encoders “predict” their inputs
  - Language models “predict” the next word
- Create a target from the x’s (or a subset)
  - Find a task to ensure that you learn something useful
  - *Self-supervised learning*

Unsupervised learning takeaways

- Most data (in the world) is unlabeled
- Useful tasks: clustering, density estimation, dimensionality reduction
- K-means and Gaussian mixtures (GMMs)
- Performance is harder to judge
  - Note that all our examples were in 2D
- Can be used in downstream tasks