Sapienza University of Rome

Master in Artificial Intelligence and Robotics

Machine Learning

A.Y. 2025/2026

Prof. Luca Iocchi

14. Unsupervised Learning

Luca Iocchi

with contributions from Valsamis Ntouskos

Overview

- Learning without a teacher
- Gaussian Mixture Model
- K-means algorithm
- Expectation Maximization algorithm
- General EM problem

Reference

C. Bishop. Pattern Recognition and Machine Learning. Chapter 9.

Unsupervised Learning

Input data available $D = \{\mathbf{x}_n\}$, but target values not available.

Unsupervised data clustering: finding multiple classes from data.

Modelling input data useful when combined with supervised learning.

Gaussian Mixture Model

Gaussian Mixture Model (GMM)

Mixed probability distribution P formed by k different Gaussian distributions

$$P(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$
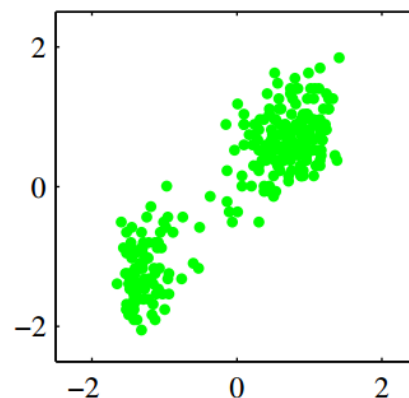
Model parameters:

- π_k, prior probability
- $\boldsymbol{\mu}_k$, mean
- $\boldsymbol{\Sigma}_k$, covariance matrix

Unsupervised learning algorithms determine mixed probability distributions from data.

Generating data from mixture of Gaussians

Each instance $\mathbf{x}_n$ generated by

1. Choosing Gaussian k according to prior probabilities $[\pi_1, \ldots, \pi_K]$
2. Generating an instance at random according to that Gaussian, using $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$
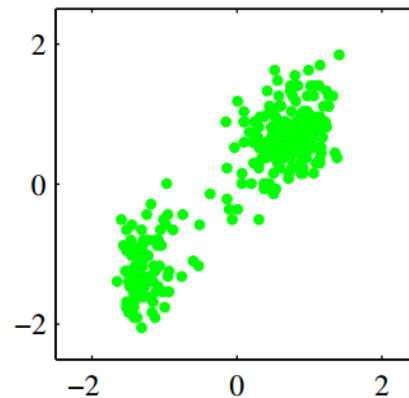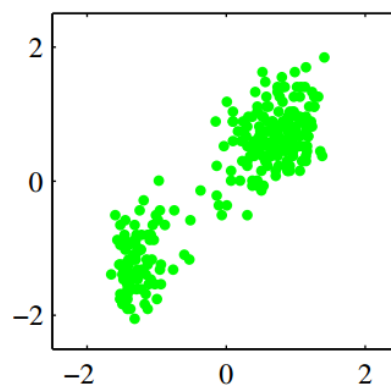
Learning with unsupervised data

Given $D = \{\mathbf{x}_n\}$,
find the parameters of the model.

For mixture of Gaussians, find

- $\pi_1, \ldots, \pi_K$
- $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K$
- $\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K$

K-means

Computing K means of data generated from K Gaussian distributions.

Input: $D = \{\mathbf{x}_n\}$, value K Output: $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K$

K-means

Step 1. Begin with a decision on the value of k = number of clusters

Step 2. Put any initial partition that classifies the data into k clusters. You may assign the training samples randomly, or systematically as follows

1. Take the first k training samples as single-element clusters
2. Assign each of the remaining (N-k) training samples to the cluster with the nearest centroid. After each assignment, recompute the centroid of the new cluster.

K-means

Step 3. Take each sample in sequence and compute its distance from the centroid of each of the clusters. If a sample is not currently in the cluster with the closest centroid, switch this sample to that cluster and update the centroid of the two clusters involved in the switch.

Step 4. Repeat step 3 until convergence is achieved, that is until a pass through the training sample causes no new assignments.

K-means convergence

The convergence will always occur if the following conditions are satisfied:

1. For each switch in step 2, the sum of distances from each training sample to that training sample's group centroid is decreased.

2. There are only finitely many partitions of the training examples into k clusters.

K-means example
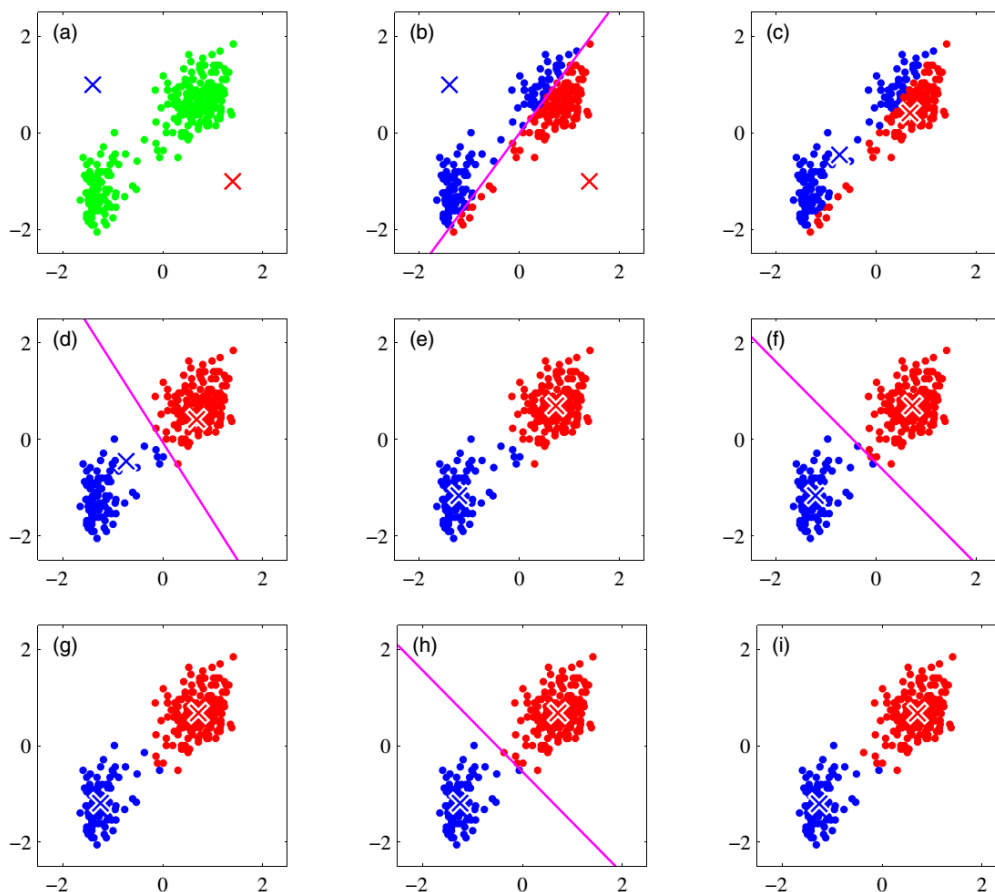
Remarks on K-means

- The number of clusters K must be determined before hand.
- Sensitive to initial condition (local optimum) when a few data available.
- Not robust to outliers. Very far data from the centroid may pull the centroid away from the real one.
- The result is a circular cluster shape because it is based on distance.

Remarks on K-means

Some solutions:

- use K-means clustering only if there are many data available
- use median instead of mean
- define better *distance* functions

Gaussian Mixture Model

$$P(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Introduce new variables $z_k \in \{0, 1\}$, probability of using Gaussian k.

$\mathbf{z} = (z_1, \ldots, z_K)^T$ 1-out-of-K encoding
(only one component is 1, all the others are 0).

Let's define $P(z_k = 1) = \pi_k$, thus

$$P(\mathbf{z}) = \prod_{k=1}^{K} \pi_k^{z_k}$$

($z_k = 1$ only for one value of k, 0 otherwise)

Gaussian Mixture Model

For a given value of $\mathbf{z}$:

$$P(\mathbf{x}|z_k = 1) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Thus

$$P(\mathbf{x}|\mathbf{z}) = \prod_{k=1}^{K} \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_k}$$

Gaussian Mixture Model

Joint distribution: $P(\mathbf{x}, \mathbf{z}) = P(\mathbf{x}|\mathbf{z})P(\mathbf{z})$ (chain rule).

Marginalization $P(\mathbf{x}) = \sum_{\mathbf{z}} P(\mathbf{x}, \mathbf{z}) = \sum_{\mathbf{z}} P(\mathbf{z})P(\mathbf{x}|\mathbf{z})$

When $\mathbf{z}$ are variables with 1-out-of-K encoding and $P(z_k = 1) = \pi_k$

$$P(\mathbf{x}) = \sum_{\mathbf{z}} P(\mathbf{z})P(\mathbf{x}|\mathbf{z}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

GMM distribution $P(\mathbf{x})$ can be seen as the marginalization of a distribution $P(\mathbf{x}, \mathbf{z})$ over variables $\mathbf{z}$.

Gaussian Mixture Model

Given observations $D = \{(\mathbf{x}_n)_{n=1}^{N}\}$, each data point $\mathbf{x}_n$ is associated to the corresponding variable $\mathbf{z}_n$ which is unknown.

Note: $z_{nk} = 1$ denotes $\mathbf{x}_n$ sampled from Gaussian k

$\mathbf{z}_n$ are called **latent variables**.

Analysis of latent variables allows for a better understanding of input data (e.g., dimensionality reduction).

Gaussian Mixture Model

Let's define the posterior

$$\gamma(z_k) \equiv P(z_k = 1|\mathbf{x}) = \frac{P(z_k = 1)\,P(\mathbf{x}|z_k = 1)}{P(\mathbf{x})}$$
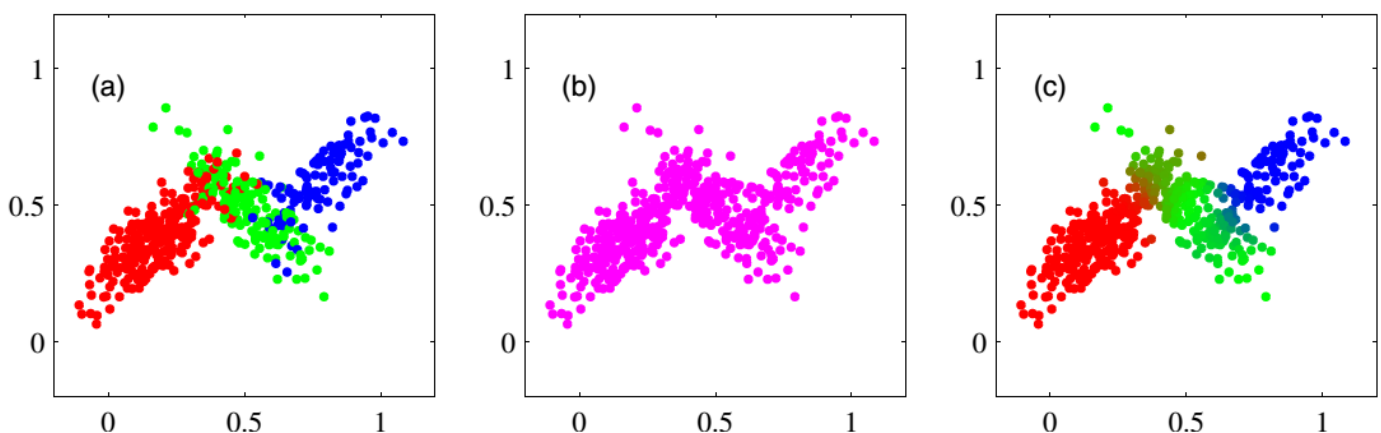
$$\gamma(z_k) = \frac{\pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

Note:

π_k: prior probability of z_k

$\gamma(z_k)$: posterior probability after observation of $\mathbf{x}$.

Gaussian Mixture Model example

a) $P(\mathbf{x}, \mathbf{z})$ with 3 latent variables $\mathbf{z}$ (red, green, blue)
b) $P(\mathbf{x})$ marginalized distribution
c) $\gamma(z_{n,k})$ posterior distribution

Expectation Maximization (EM)

Given data set $D = \{(x_n)_{n=1}^N\}$ and GMM

$$P(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

determine $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k$

Note: generalization of K-means algorithm

Expectation Maximization (EM)

Maximum likelihood

$$\operatorname*{argmax}_{\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\Sigma}} \ln P(\mathbf{X}|\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\Sigma})$$

At maximum:

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}_n$$

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T$$

$$\pi_k = \frac{N_k}{N}, \quad \text{with } N_k = \sum_{n=1}^{N} \gamma(z_{nk})$$

Expectation Maximization (EM)

- **E step**
  Given $\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$, compute $\gamma(z_{nk})$

- **M step**
  Given $\gamma(z_{nk})$, compute $\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$

Expectation Maximization (EM)

- Initialize $\pi_k^{(0)}, \boldsymbol{\mu}_k^{(0)}, \boldsymbol{\Sigma}_k^{(0)}$
- Repeat until termination condition $t = 0, \ldots, T$
  - **E step**

$$\gamma(z_{nk})^{(t+1)} = \frac{\pi_k^{(t)} \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})}{\sum_{j=1}^{K} \pi_j^{(t)} \mathcal{N}(\mathbf{x}_k; \boldsymbol{\mu}_j^{(t)}, \boldsymbol{\Sigma}_j^{(t)})}$$
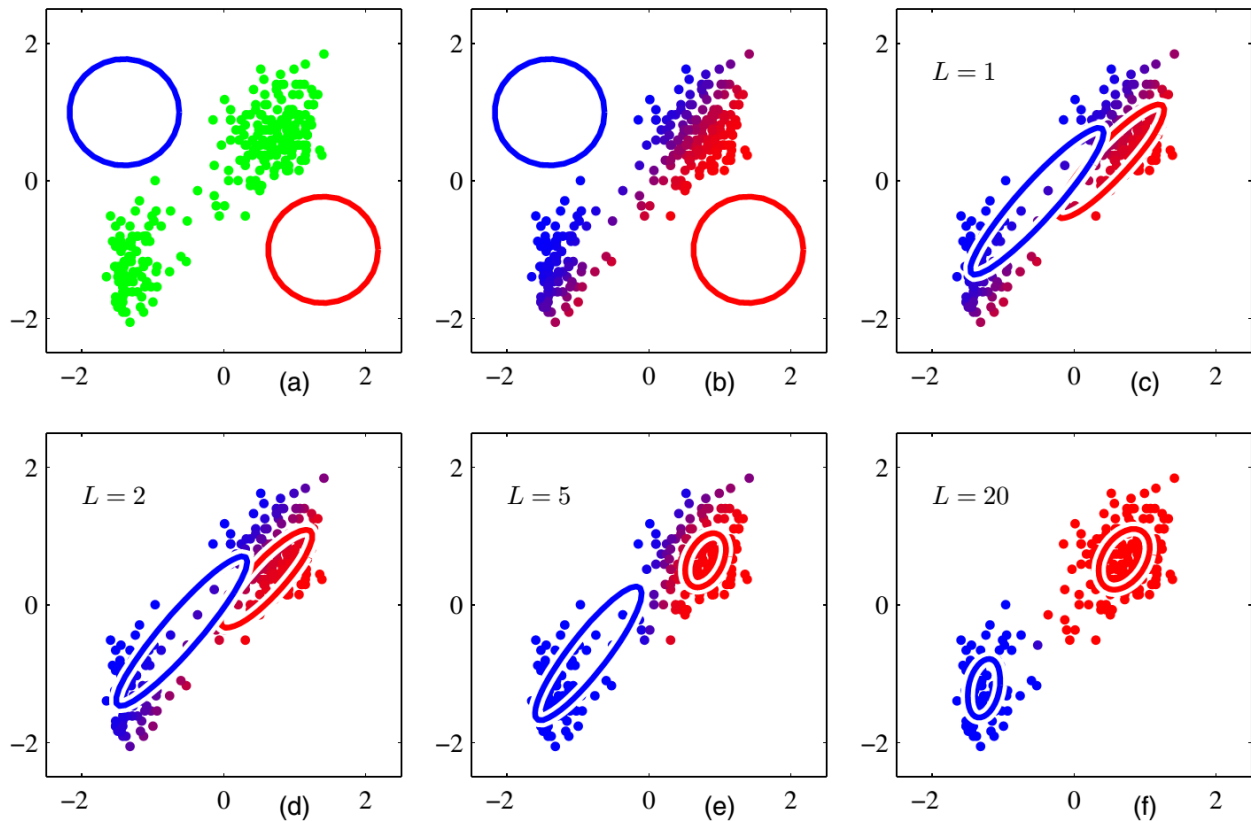
  - **M step**

$$\boldsymbol{\mu}_k^{(t+1)} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})^{(t+1)} \mathbf{x}_n$$

$$\boldsymbol{\Sigma}_k^{(t+1)} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})^{(t+1)} (\mathbf{x}_n - \boldsymbol{\mu}_k^{(t+1)})(\mathbf{x}_n - \boldsymbol{\mu}_k^{(t+1)})^T$$

$$\pi_k^{(t+1)} = \frac{N_k}{N}, \quad with \ N_k = \sum_{n=1}^{N} \gamma(z_{nk})^{(t+1)}$$

EM example

Remarks on EM Algorithm

- Converges to local maximum likelihood
- Provides estimates of the latent variables variables z_{nk}
- Extended version of K-means (probabilistic assignment to a cluster z_{nk})
- Can be generalized to other distributions (not only Gaussians)

General EM Problem

Given:

- Observed data $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$
- Unobserved latent values $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$
- Parametrized probability distribution $P(\mathbf{Y}|\boldsymbol{\theta})$, where
    - $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$ is the full data $\mathbf{y}_n = \langle \mathbf{x}_n, \mathbf{z}_n \rangle$
    - $\boldsymbol{\theta}$ are the parameters

Determine:

- $\boldsymbol{\theta}^*$ that (locally) maximizes $E[\ln P(\mathbf{Y}|\boldsymbol{\theta})]$

Many uses:

- Unsupervised clustering
- Bayesian Networks
- Hidden Markov Models

General EM Method

Define likelihood function $Q(\boldsymbol{\theta}'|\boldsymbol{\theta})$ defined on variables $\mathbf{Y} = \mathbf{X} \cup \mathbf{Z}$, using observed $\mathbf{X}$ and current parameters $\boldsymbol{\theta}$ to estimate $\mathbf{Z}$

EM Algorithm:

Estimation (E) step: Calculate $Q(\boldsymbol{\theta}'|\boldsymbol{\theta})$ using current hypothesis $\boldsymbol{\theta}$ and observed data $\mathbf{X}$ to estimate probability distribution over $\mathbf{Y}$

$$Q(\boldsymbol{\theta}'|\boldsymbol{\theta}) \leftarrow E[\ln P(\mathbf{Y}|\boldsymbol{\theta}')|\boldsymbol{\theta}, \mathbf{X}]$$

Maximization (M) step: Replace hypothesis $\boldsymbol{\theta}$ by the hypothesis $\boldsymbol{\theta}'$ that maximizes this Q function

$$\boldsymbol{\theta} \leftarrow \underset{\boldsymbol{\theta}'}{\operatorname{argmax}} \, Q(\boldsymbol{\theta}'|\boldsymbol{\theta})$$

Summary

- Unsupervised learning useful to deal with unknown variables
- Clustering when labeled data are not available
- EM algorithm is a general method to estimate likelihood for mixed distributions including observed and latent variables
- Concepts to be extended to continuous latent variables