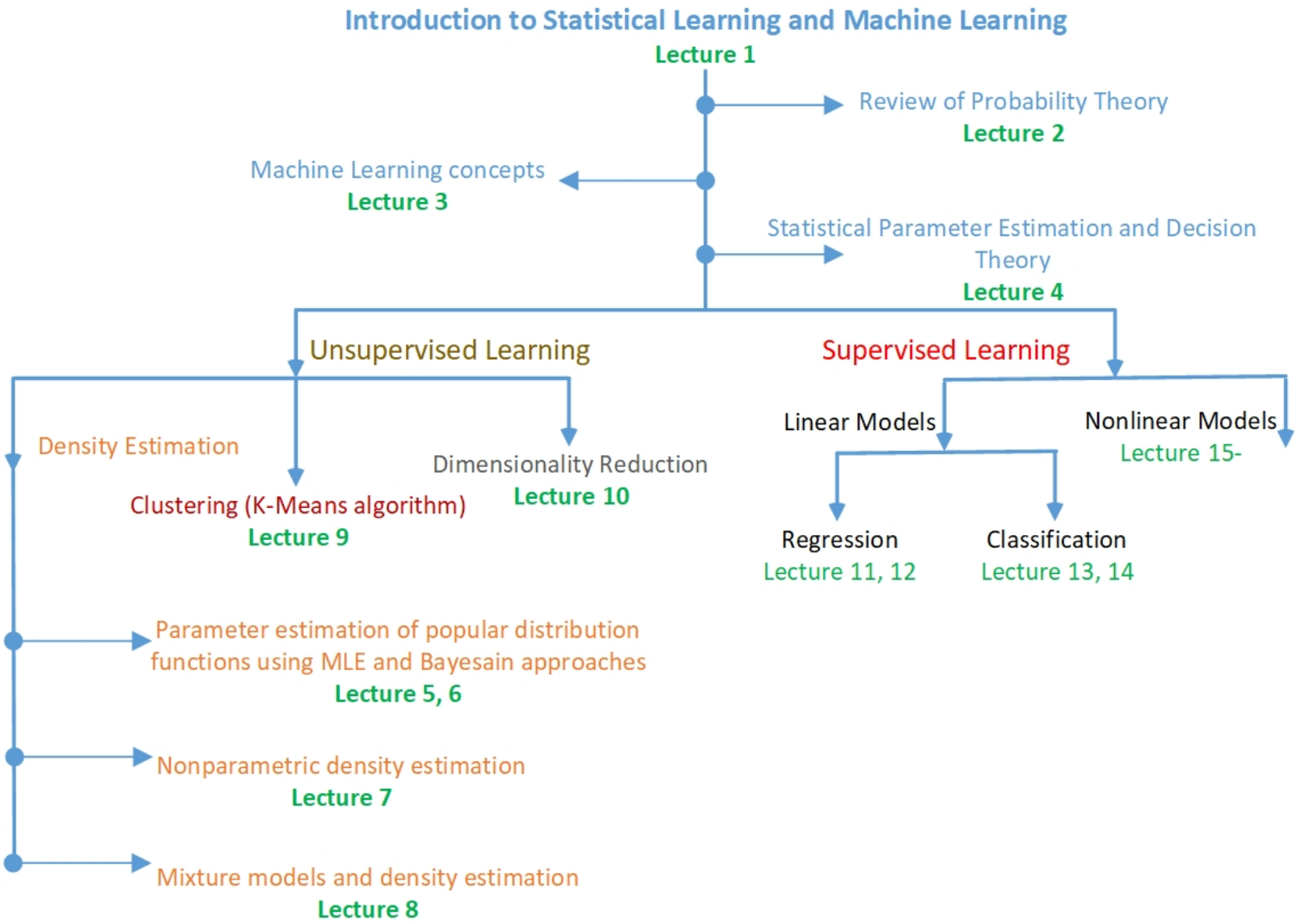
Statistical Learning and Machine Learning

Lecture 8 - Mixture Models

September 18, 2021

Course overview and where do we stand



Objectives of the lecture

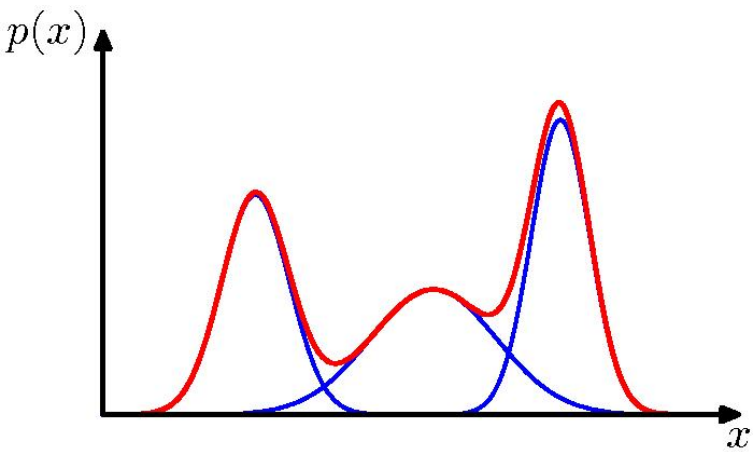
- K-means Clustering
- Mixture of Gaussians

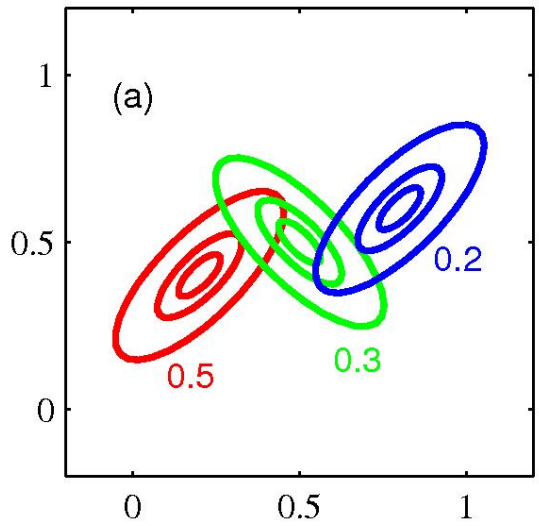
Mixture Models and Clustering

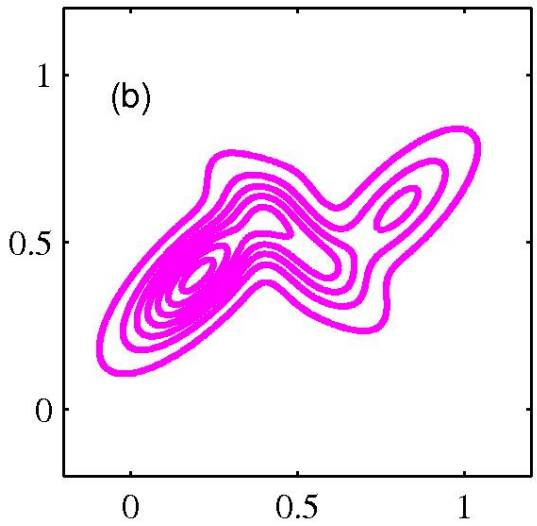
Mixture models are widely used in:

- Pattern Recognition
- Data Mining (Clustering)
- Machine Learning
- Statistical analysis

They allow for using complex distributions, formed by simpler components.



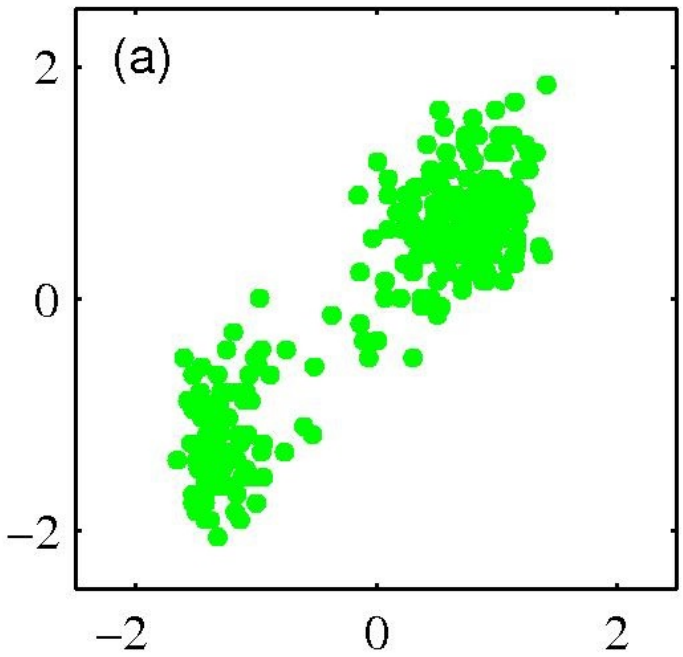




K-Means clustering

Clustering is the task of grouping a set of data points in such a way that:

- group of data points in the same group have inter-point distances which are smaller compared with distances to points outside the cluster/group;



K-Means clustering

Let $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ be a set of N data points $\boldsymbol{x}_n \in \mathbb{R}^D$. Our goal is to group these N data points to K clusters (we assume that K is known).

We introduce two types of variables:

- a set of K prototype vectors $\boldsymbol{\mu}_k \in \mathbb{R}^D$, $k = 1, \ldots, K$, each representing (a center of) one cluster
- For each data point $\boldsymbol{x}_n$, a set of binary indicator variables $r_{nk} \in \{0, 1\}$, where $k = 1, \ldots, K$ describing which of the K clusters $\boldsymbol{x}_n$ belongs to:

$$r_{nk} = \begin{cases} 1, & \text{if } \boldsymbol{x}_n \text{ belongs to cluster } k \\ 0, & \text{otherwise} \end{cases}$$

K-Means clustering

The task is to assign each of the N data points to one of the total K number of clusters. This will be achieved by

- finding the prototype vectors $\boldsymbol{\mu}_k,\ k = 1, \ldots, K$
- computing the values of indicator variables, $r_{nk},\ n = 1, \ldots, N$ and $k = 1, \ldots, K$

K-Means Clustering

K-Means finds the vectors $\boldsymbol{\mu}_k,\ k = 1, \ldots, K$ and $\boldsymbol{r}_n,\ n = 1, \ldots, N$ by minimizing the sum of squares of distances of each point to its assigned cluster mean vector

$$\mathcal{J} = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\boldsymbol{x}_n - \boldsymbol{\mu}_k\|^2$$

$\mathcal{J}$ is sometimes called *distortion measure*.

Because $\boldsymbol{\mu}_k$ and $\boldsymbol{r}_n$ are inter-connected, we apply an iterative procedure in which each iteration involves two successive steps:

- Given $\boldsymbol{\mu}_k,\ k = 1, \ldots, K$, we minimize $\mathcal{J}$ w.r.t. $\boldsymbol{r}_n,\ n = 1, \ldots, N$
- Given $\boldsymbol{r}_n,\ n = 1, \ldots, N$, we minimize $\mathcal{J}$ w.r.t. $\boldsymbol{\mu}_k,\ k = 1, \ldots, K$

K-Means Clustering: Update of $\boldsymbol{r}_n$

For known $\boldsymbol{\mu}_k$, $k = 1, \ldots, K$, r_{nk} are updated as:

$$r_{nk} = \begin{cases} 1, & \textit{if k} = \arg\min_j \|\boldsymbol{x}_n - \boldsymbol{\mu}_j\|^2 \\ 0, & \text{otherwise} \end{cases}$$

This means that the data point $\boldsymbol{x}_n$ is assigned to the cluster corresponding to the closest cluster mean vector $\boldsymbol{\mu}_k$.

K-Means Clustering: Update of $\boldsymbol{\mu}_k$

For known $\boldsymbol{r}_n$, $n = 1, \ldots, N$, $\mathcal{J}$ is a quadratic function of $\boldsymbol{\mu}_k$.

It can be minimized by setting $\frac{\theta \mathcal{J}}{\theta \boldsymbol{\mu}_k}$ to zero:

$$2 \sum_{n=1}^{N} r_{nk}(\boldsymbol{x}_n - \boldsymbol{\mu}_k) = 0$$

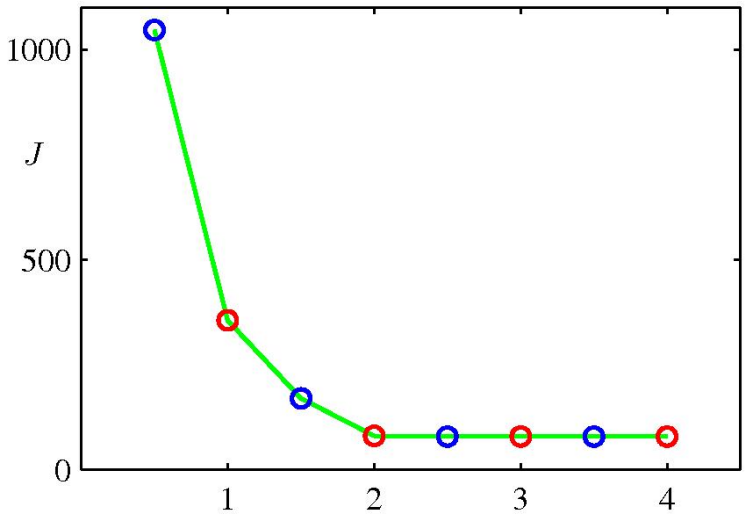
which leads to:

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} r_{nk} \boldsymbol{x}_n}{\sum_{n=1}^{N} r_{nk}}$$

Thus, $\boldsymbol{\mu}_k$ is equal to the mean vector of the data points assigned to cluster k.

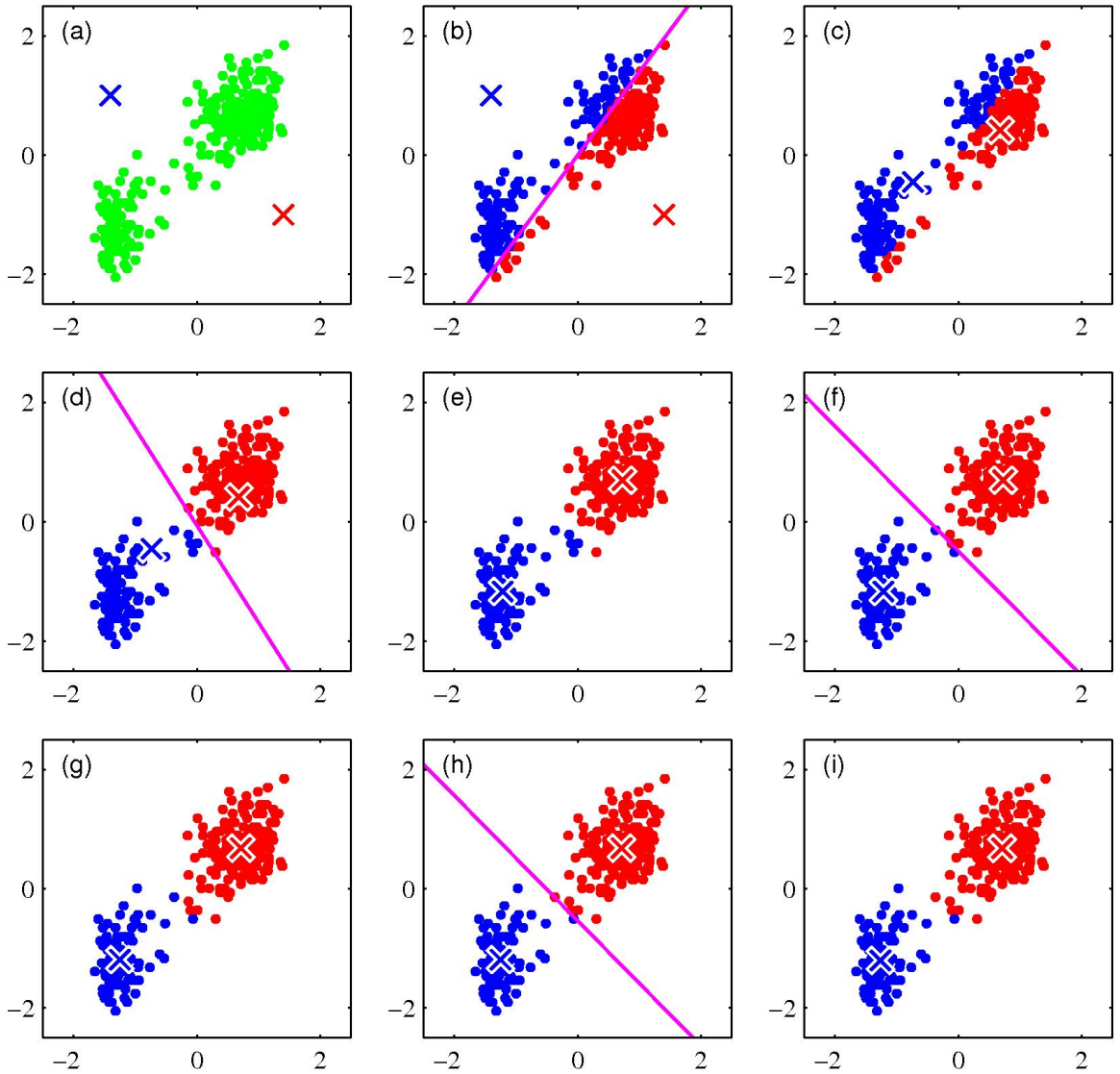
K-Means Clustering

The two processing steps are repeated in turn until there is no change in the assignments (we say that the method converges).



Because each update reduces the value of $\mathcal{J}$, convergence is assured.

Graphical illustration of K-Means Clustering



K-Means: Sequential Updates

- Instead of updating the data point assignments to clusters at once for the entire data set, we can sequentially update the assignment of each (newly arrived) data point $\boldsymbol{x}_n$ at a time.
- After assigning $\boldsymbol{x}_n$ to the cluster of the nearest prototype $\boldsymbol{\mu}_k$, the prototype is updated as follows:

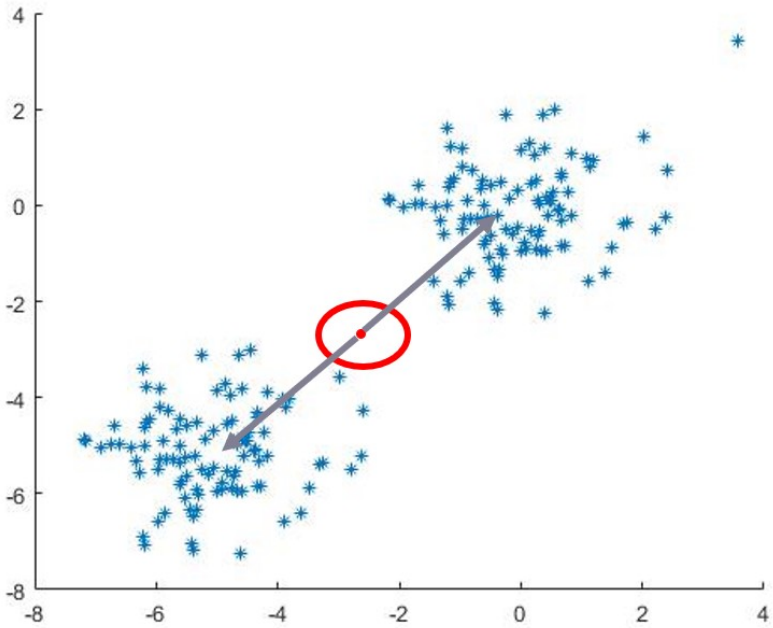
$$\boldsymbol{\mu}_k^{new} = \boldsymbol{\mu}_k^{old} + \eta_n(\boldsymbol{x}_n - \boldsymbol{\mu}_k^{old})$$

where η_n is the learning rate parameter.

K-Means Clustering

Properties:

- Assumes that the number of clusters K is known (defined by the user)
- Easy to implement
- Fast to compute: The computational cost of each step is $O(KN)$
- For data points that lie roughly at the midway between cluster mean vectors, hard assignment to the nearest cluster may not be the best way



K-Medoids clustering

K-Means clustering uses the the Euclidean distance measure. Other *dissimilarity measures* $\mathcal{V}(\boldsymbol{x}, \boldsymbol{x}')$ between two vectors $\boldsymbol{x}$ and $\boldsymbol{x}'$ can also be used, with the following advantages:

- robustness to outliers
- suitable for cases where one or ore variables in $\boldsymbol{x}$ denote class or labels

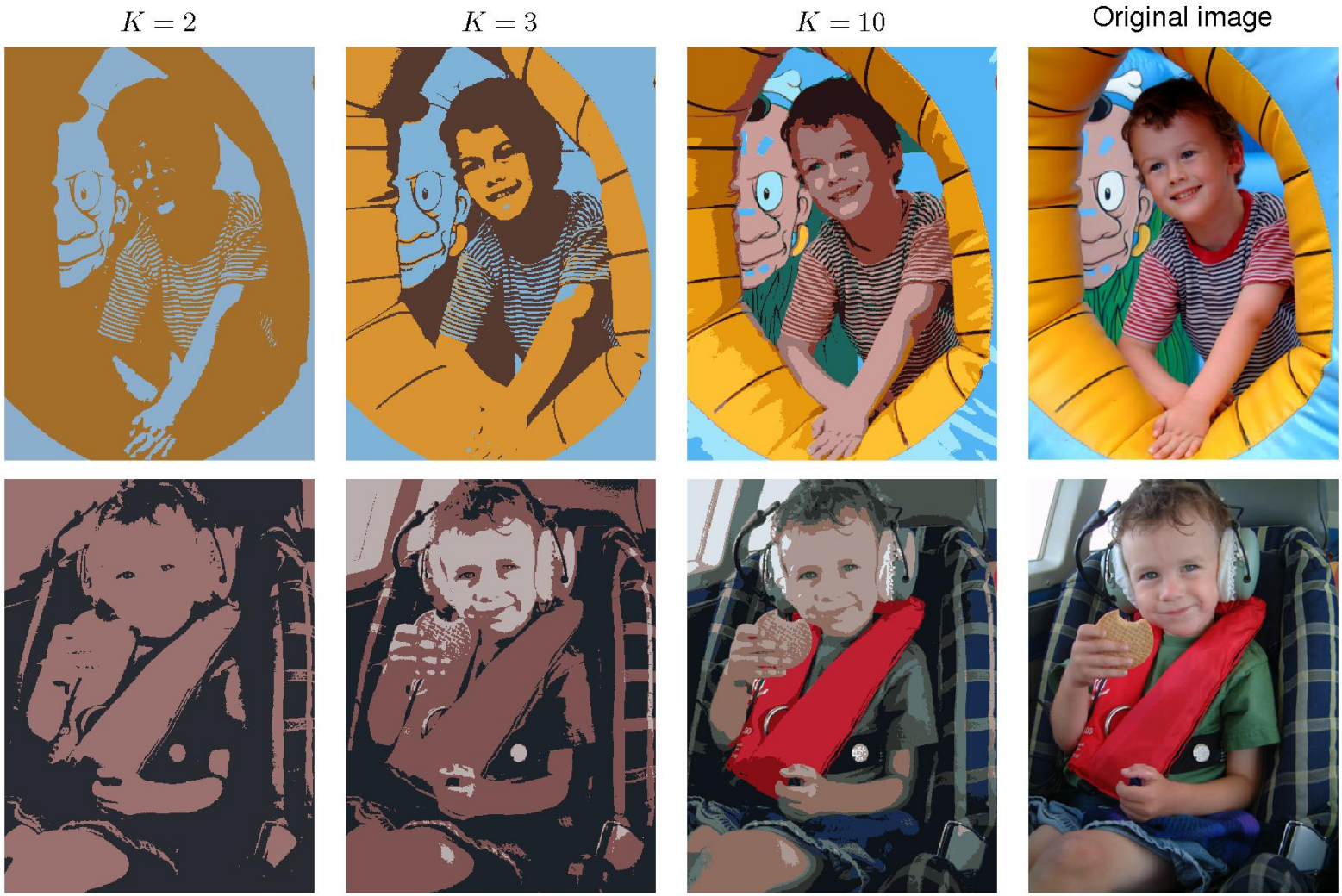
Using this measure, $\boldsymbol{\mu}_k,\ k = 1, \ldots, K$ and $\boldsymbol{r}_n,\ n = 1, \ldots, N$ can be found by minimizing:

$$\tilde{\mathcal{J}} = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \mathcal{V}(\boldsymbol{x}_n, \boldsymbol{\mu}_k).$$

The E step (assignment to cluster) remains the same but the M step (updating the prototype vector) becomes complex.

Clustering and image segmentation

Goal: Partition image into regions of homogeneous visual appearance



Mixture of Gaussians

Reminder: When using a superposition of K Gaussians, the resulting distribution has the form:

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \Sigma_k) \tag{5}$$

and is called mixture of Gaussian.

The parameters π_k are called *mixing coefficients* and satisfy:

$$0 \leq \pi_k \leq 1, \qquad \sum_{k=1}^{K} \pi_k = 1.$$

Thus, the mixing coefficients have the form of probabilities ($\pi_k = p(k)$).

Mixture of Gaussians: Latent variables

Let $\boldsymbol{z}$ be a binary random variable having a 1-of-K representation satisfying:

$$z_k \in \{0, 1\}, \qquad \sum_{k=1}^{K} z_k = 1.$$

$\boldsymbol{z}$ is associated with the marginal distribution $p(\boldsymbol{x})$, which is specified as:

$$p(z_k = 1) = \pi_k$$

Reminder: Since $\boldsymbol{z}$ is a 1-of-K vector:

$$p(\boldsymbol{z}) = \prod_{k=1}^{K} \pi_k^{z_k}$$

Mixture of Gaussians: Latent variables

The conditional distribution of $\boldsymbol{x}$ for the k-th Gaussian is:

$$p(\boldsymbol{x}|z_k = 1) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \Sigma_k)$$

and with respect to $\boldsymbol{z}$ is:

$$p(\boldsymbol{x}|\boldsymbol{z}) = \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \Sigma_k)^{z_k}$$

We can now write $p(\boldsymbol{x})$:

$$p(\boldsymbol{x}) = \sum_{z} p(\boldsymbol{x}, \boldsymbol{z}) = \sum_{z} p(\boldsymbol{z}) p(\boldsymbol{x}|\boldsymbol{z}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \Sigma_k) \qquad (6)$$

Compare (6) with (5)!

Mixture of Gaussians: Latent variables

The conditional probability of $\boldsymbol{z}$ given $\boldsymbol{x}$ is (using Bayes' theorem):

$$\begin{aligned} \gamma(z_k) \equiv p(z_k = 1|\boldsymbol{x}) &= \frac{p(z_k = 1)p(\boldsymbol{x}|z_k = 1)}{\sum_{j=1}^{K} p(z_j = 1)p(\boldsymbol{x}|z_j = 1)} \\ &= \frac{\pi_k \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_j, \Sigma_j)} \end{aligned} \tag{7}$$

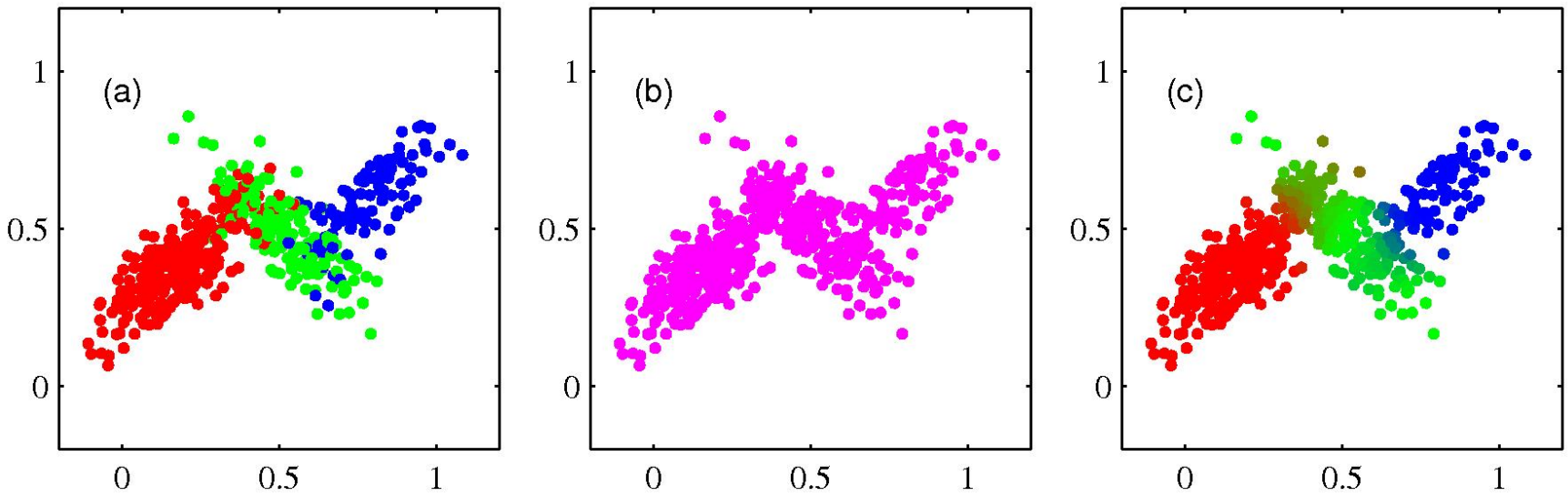
Thus, we consider π_k as the prior probability of $z_k = 1$ and $\gamma(z_k)$ as the corresponding posterior probability after observing $\boldsymbol{x}$ (also called *responsibility* of component k for 'explaining' $\boldsymbol{x}$).

Mixture of Gaussians: Latent variables

Technique to generate random samples described by the Gaussian mixture model:

1. Generate a random vector $\hat{\boldsymbol{z}}$ using $p(\boldsymbol{z})$ (multinomial distribution)
2. Generate $\boldsymbol{x}$ using $p(\boldsymbol{x}|\hat{\boldsymbol{z}}) = \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \Sigma_k)^{\hat{z}_k}$

Use $p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})$, $p(\boldsymbol{x})$ and $\gamma(\boldsymbol{z})$ for visualization.



Mixture of Gaussians: Parameter estimation using MLE

Given a set of i.i.d. data points $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, we want to estimate the parameters of the mixture of Gaussians:

$$\{\boldsymbol{\mu}_k, \ \Sigma_k \ \pi_k\}, \quad k = 1, \ldots, K$$

The log-likelihood function is:

$$\begin{aligned} \ln p(\mathcal{D}|\boldsymbol{\pi}, \boldsymbol{\mu}, \Sigma) &= \ln \prod_{n=1}^{N} p(\boldsymbol{x}_n) \\ &= \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_k, \Sigma_k) \right\} \end{aligned}$$

Maximization of the log-likelihood function is more complex than in the case of a single Gaussian. Why?

Mixture of Gaussians: Parameter estimation using MLE

Problem of singularity

- Consider the case where the covariance matrices of Gaussian mixture model are diagonal: $\Sigma_k = \sigma_k^2 \boldsymbol{I}$
- Further consider that $\boldsymbol{\mu}_j = \boldsymbol{x}_n$
- The data point will contribute the following term to the likelihood function:

$$\mathcal{N}(\boldsymbol{x}_n | \boldsymbol{x}_n, \sigma_j^2 \boldsymbol{I}) = \frac{1}{\sqrt{2\pi}} \frac{1}{\sigma_j}$$

  For $\sigma_j \to 0$, the above term will o to infinity and thus maximization of the likelihood function will not make sense.

- Remedy: Detecting the singularity and resetting the mean $\boldsymbol{\mu}_j$ to random value and resetting the covariance to some large value.

Mixture of Gaussians: Parameter estimation using MLE

We set the derivative of the log-likelihood function w.r.t. each parameter equal to zero.

$$
\begin{aligned}
0 &= \frac{\partial \ln p(\mathcal{D}|\boldsymbol{\pi}, \boldsymbol{\mu}, \Sigma)}{\partial \boldsymbol{\mu}_k} \\
0 &= -\sum_{n=1}^{N} \frac{\pi_k \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_k, \Sigma_k)}{\sum_{j=1}^{N} \pi_j \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_j, \Sigma_j)} \Sigma_k(\boldsymbol{x}_n - \boldsymbol{\mu}_k) \qquad (8)
\end{aligned}
$$

By multiplying with Σ_k^{-1} (assuming Σ_k is nonsingular matrix):

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

with $N_k = \sum_{n=1}^{N} \gamma(z_{nk})$ being the effective number of data points assigned to cluster k.

Mixture of Gaussians: Parameter estimation using MLE

We set the derivative of the log-likelihood function w.r.t. Σ_k equal to zero:

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

Thus, Σ_k is the covariance matrix of a single Gaussian fitted to all data, with each data point weighted by the corresponding responsibility value and divided with the effective number of points.

Mixture of Gaussians: Parameter estimation using MLE

To optimize w.r.t. π_k, we need to also consider the constraint that $\sum_{k=1}^{K} \pi_k = 1$. To do so, we use a Lagrange multiplier and maximize for:

$$\ln p(\mathcal{D}|\boldsymbol{\pi}, \boldsymbol{\mu}, \Sigma) + \lambda \left(\sum_{k=1}^{K} \pi_k - 1 \right)$$

We set the derivative of the above Lagrangian function equal to zero and:

$$\sum_{n=1}^{N} \frac{\mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_k, \Sigma_k)}{\sum_{j=1}^{K} \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_j, \Sigma_j)} + \lambda = 0$$

leading to $\pi_k = N_k/N$ (π_k is the average responsibility of the component to explain the data).

Mixture of Gaussians: Expectation-Maximization