

Machine Learning Course - CS-433

# Gaussian Mixture Models

Nov 30, 2021

changes by Martin Jaggi 2021, 2020, 2019, changes by Rüdiger Urbanke 2018, changes by  
Martin Jaggi 2017, 2016 ©Mohammad Emtiyaz Khan 2015

Last updated on: November 30, 2021



## Motivation

K-means forces the clusters to be *spherical*, but sometimes it is desirable to have *elliptical* clusters. Another issue is that, in K-means, each example can only belong to one cluster, but this may not always be a good choice, e.g. for data points that are near the “border”. Both of these problems are solved by using Gaussian Mixture Models.

## Clustering with Gaussians

The first issue is resolved by using full covariance matrices  $\Sigma_k$  instead of *isotropic* covariances.

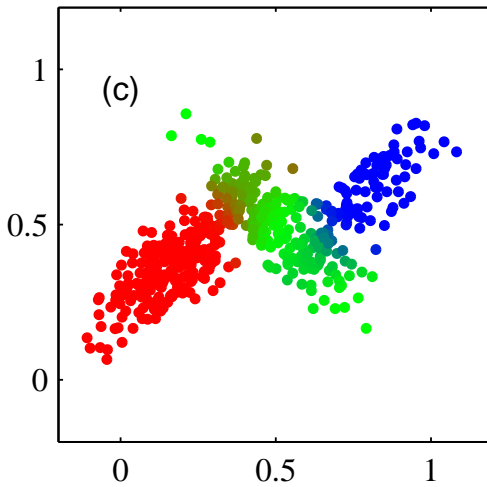
$$p(\mathbf{X}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathbf{z}) = \prod_{n=1}^N \prod_{k=1}^K [\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)]^{z_{nk}}$$

## Soft-clustering

The second issue is resolved by defining  $z_n$  to be a random variable. Specifically, define  $z_n \in \{1, 2, \dots, K\}$  that follows a [multinomial distribution](#).

$$p(z_n = k) = \pi_k \text{ where } \pi_k > 0, \forall k \text{ and } \sum_{k=1}^K \pi_k = 1$$

This leads to [soft-clustering](#) as opposed to having “hard” assignments.



## Gaussian mixture model

Together, the [likelihood](#) and the [prior](#) define the [joint](#) distribution of Gaussian mixture model (GMM):

$$\begin{aligned} p(\mathbf{X}, \mathbf{z} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi}) \\ &= \prod_{n=1}^N p(\mathbf{x}_n \mid z_n, \boldsymbol{\mu}, \boldsymbol{\Sigma}) p(z_n \mid \boldsymbol{\pi}) \\ &= \prod_{n=1}^N \prod_{k=1}^K [\mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)]^{z_{nk}} \prod_{k=1}^K [\pi_k]^{z_{nk}} \end{aligned}$$

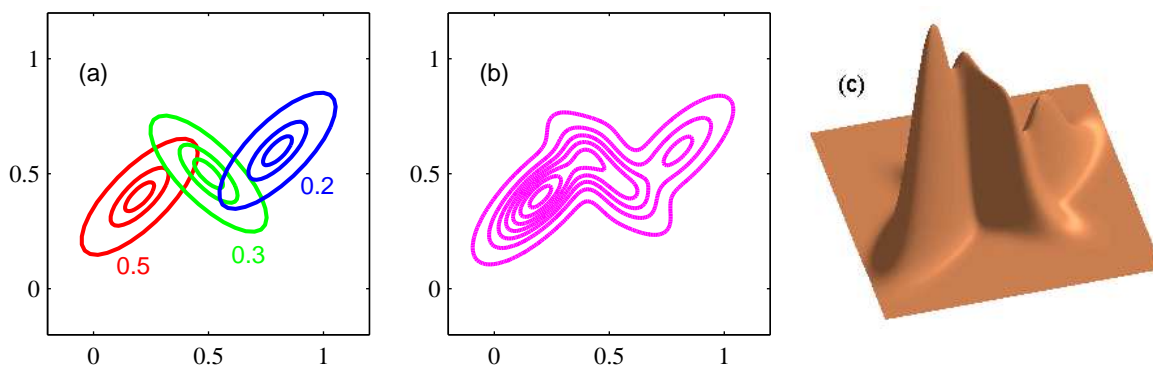
Here,  $\mathbf{x}_n$  are observed data vectors,  $z_n$  are *latent* unobserved variables, and the unknown *parameters* are given by  $\boldsymbol{\theta} := \{\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K, \boldsymbol{\pi}\}$ .

# Marginal likelihood

GMM is a **latent variable model** with  $z_n$  being the unobserved (latent) variables. An advantage of treating  $z_n$  as latent variables instead of *parameters* is that we can *marginalize* them out to get a cost function that does not depend on  $z_n$ , i.e. as if  $z_n$  never existed.

Specifically, we get the following **marginal likelihood** by marginalizing  $z_n$  out from the likelihood:

$$p(\mathbf{x}_n | \boldsymbol{\theta}) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$



Deriving cost functions this way is good for *statistical efficiency*. Without a latent variable model, the number of parameters grows at rate  $\mathcal{O}(N)$ . After marginalization, the growth is reduced to  $\mathcal{O}(D^2K)$  (assuming  $D, K \ll N$ ).

# Maximum likelihood

To get a maximum (marginal) likelihood estimate of  $\boldsymbol{\theta}$ , we maximize the following:

$$\max_{\boldsymbol{\theta}} \sum_{n=1}^N \log \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Is this cost convex?   Identifiable?  
Bounded?

