Mixture Models

Bùi Tiến Lên

2022

KHOA CÔNG NGHỆ THÔNG TIN
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

Contents

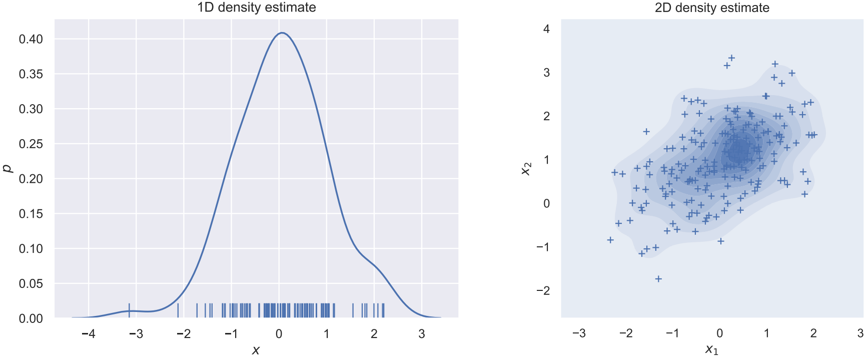
Notation

symbol	meaning
$a, b, c, N \ldots$	scalar number
$\boldsymbol{w}, \boldsymbol{v}, \boldsymbol{x}, \boldsymbol{y} \ldots$	column vector
$\boldsymbol{X}, \boldsymbol{Y} \ldots$	matrix
$\mathbb{R}$	set of real numbers
$\mathbb{Z}$	set of integer numbers
$\mathbb{N}$	set of natural numbers
$\mathbb{R}^D$	set of vectors
$\mathcal{X}, \mathcal{Y}, \ldots$	set
$\mathcal{A}$	algorithm

operator	meaning
$\boldsymbol{w}^\mathsf{T}$	transpose
$\boldsymbol{X}\boldsymbol{Y}$	matrix multiplication
$\boldsymbol{X}^{-1}$	inverse

Unsupervised Problem

- Given data set $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$, find its distribution $p(\boldsymbol{x} \mid \mathcal{D})$



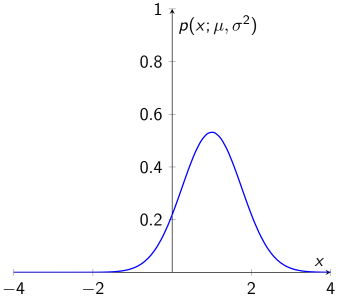
Introduction

- Gaussian Distribution
- Gaussian Model
- Gaussian Mixture Model
- Applications

Univariate Normal Distribution

- A random variable X is normally distributed with parameters (μ, σ^2), denoted as $\mathcal{N}(x; \mu, \sigma^2)$ if its density function is given by

$$p(x; \mu, \sigma^2) = \mathcal{N}(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad (1)$$

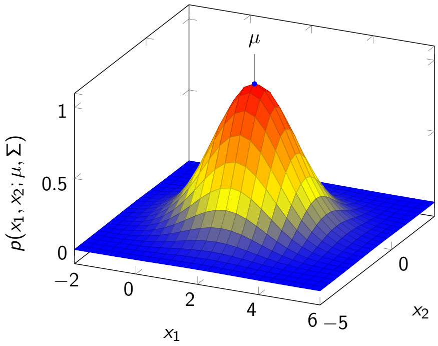


Multivariate Normal Distribution

- A multivariate normal distribution is defined by two parameters:
  - mean vector $\boldsymbol{\mu} \in \mathbb{R}^D$
  - covariance matrix $\Sigma \in \mathbb{R}^{D \times D}$, where Σ is a positive definite matrix.
- The density function is given by

$$p(\boldsymbol{x}; \boldsymbol{\mu}, \Sigma) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu})\right] \quad (2)$$

Multivariate Normal Distribution (cont.)



- A random vector-valued variable $\boldsymbol{x} = (x_1, x_2, ..., x_D)$ is called normally distributed if all linear combinations of its components $x_i, i = 1, ..., D$ is normally distributed.

Multivariate Normal Distribution (cont.)

- In other words:

$$\exists \mu \in \mathbb{R}, \sigma \in \mathbb{R} : \boldsymbol{w} \cdot \boldsymbol{x}^{\mathsf{T}} \sim \mathcal{N}(\mu, \sigma^2), \forall w \in \mathbb{R}^D.$$

- A square matrix A $n \times n$ is called positive definite if

$$\boldsymbol{z}^{\mathsf{T}} A \boldsymbol{z} > 0, \forall \boldsymbol{z} \in \mathbb{R}^n, \boldsymbol{z} \neq \mathbf{0}.$$

What is a Gaussian Model?

- Input data set $\mathcal{D} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$
- The likelihood of $\boldsymbol{x}$ given a Gaussian model is

$$p(\boldsymbol{x} \mid \boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu})\right]$$

  where D is the dimension of $\boldsymbol{x}$, $\boldsymbol{\mu}$ is the **mean** and Σ is the **covariance matrix** of the Gaussian. Σ is often **diagonal**.
- **Objective**: maximize the likelihood $p(\mathcal{D} \mid \boldsymbol{\mu}, \Sigma)$ of the data $\mathcal{D}$ drawn from the Gaussian model

$$\arg\max_{\boldsymbol{\mu}, \Sigma} p(\mathcal{D} \mid \boldsymbol{\mu}, \Sigma) = \arg\max_{\boldsymbol{\mu}, \Sigma} \prod_{i=1}^{n} p(\boldsymbol{x}_i \mid \boldsymbol{\mu}, \Sigma) \tag{3}$$

Closed-form Solution

- Solve the optimization problem (1), we have

$$\hat{\boldsymbol{\mu}} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i \tag{4}$$

$$\hat{\Sigma} = \frac{1}{n-1} \sum_{i=1}^{n} (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}})(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}})^{\mathsf{T}} \tag{5}$$

What is a Gaussian Mixture Model?

- A Gaussian Mixture Model (GMM) is a **distribution**
- The likelihood given a Gaussian distribution is

$$\mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu})\right] \tag{6}$$

where D is the dimension of $\boldsymbol{x}$, $\boldsymbol{\mu}$ is the **mean** and Σ is the **covariance matrix** of the Gaussian. Σ is often **diagonal**.

- The likelihood given a GMM is

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} w_k \cdot \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_k, \Sigma_k) \tag{7}$$

where K is the number of Gaussians and w_k is the **weight** of Gaussian k, with

$$\sum_{k=1}^{K} w_k = 1 \text{ and } w_k \geq 0 \tag{8}$$

Characteristics of a GMM

- ANNs are **universal approximators of functions**
- GMMs are **universal approximators of densities** (*as long as there are enough Gaussians of course*)
- Even **diagonal GMMs** are universal approximators.
- Full rank GMMs are not easy to handle: number of parameters is the square of the number of dimensions.
- GMMs can be trained by maximum likelihood using an efficient algorithm: **Expectation-Maximization**.

Practical Applications using GMMs

- Biometric person authentication (using voice, face, handwriting, etc):
  - one GMM for the **client**
  - one GMM for **all the others**
  - Bayes decision $\Longrightarrow$ likelihood ratio
- Any highly imbalanced classification task
  - one GMM per class, tuned by maximum likelihood
  - Bayes decision $\Longrightarrow$ likelihood ratio
- Dimensionality reduction
- Quantization

The EM Algorithm

- Introduction
- Graphical Interpretation
- More Formally

Basics of Expectation-Maximization

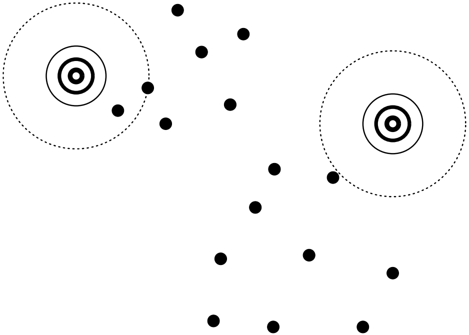
- **Objective**: maximize the likelihood $p(\mathcal{D} \mid \theta)$ of the data $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ drawn from an unknown distribution, given the model parameterized by θ:

$$\theta^* = \arg\max_\theta p(\mathcal{D} \mid \theta) = \arg\max_\theta \prod_{i=1}^{n} p(\boldsymbol{x}_i \mid \theta) \tag{9}$$

- Basic ideas of EM:
  - Introduce a **hidden variable** such that *its knowledge would simplify the maximization of $p(\mathcal{D} \mid \theta)$*
- At each iteration of the algorithm:
  - **E-Step**: **estimate** the distribution of the hidden variable given the data and the current value of the parameters
  - **M-Step**: modify the parameters in order to **maximize** the joint distribution of the data and the hidden variable

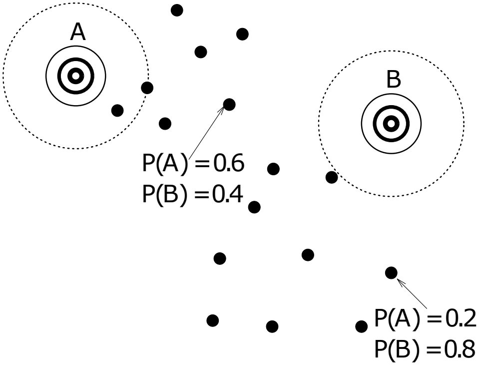
EM for GMM - Graphical View

- Hidden variable: for each point, **which Gaussian generated it**?



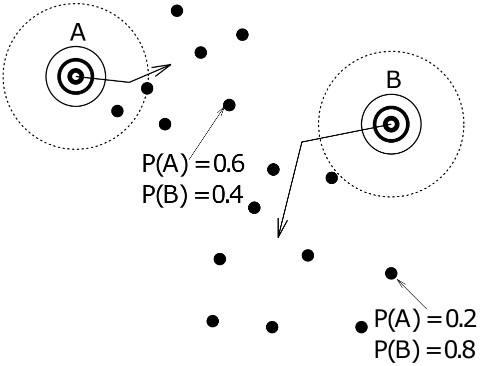
EM for GMM - Graphical View (cont.)

- **E-Step**: for each point, **estimate** the probability that each Gaussian generated it



EM for GMM - Graphical View (cont.)

- **M-Step**: modify the parameters according to the hidden variable to **maximize** the likelihood of the data (and the hidden variable)



EM: More Formally

- Let us call the hidden variable Q and consider the following auxiliary function:

$$A(\theta, \theta^t) = \mathbb{E}_Q \left[\log p(\mathcal{D}, Q \mid \theta) \mid \mathcal{D}, \theta^t\right] \tag{10}$$

- It can be shown that maximizing A

$$\theta^{t+1} = \arg\max_{\theta} A(\theta, \theta^t) \tag{11}$$

always increases the likelihood of the data $p(\mathcal{D} \mid \theta^{t+1})$, and a maximum of A corresponds to a maximum of the likelihood.

Proof of Convergence

- First let us develop the auxiliary function:

$$\begin{aligned} A(\theta, \theta^t) &= \mathbb{E}_Q \left[\log p(\mathcal{D}, Q \mid \theta) \mid \mathcal{D}, \theta^t \right] \\ &= \sum_{q \in Q} P(q \mid \mathcal{D}, \theta^t) \log p(\mathcal{D}, q \mid \theta) \\ &= \sum_{q \in Q} P(q \mid \mathcal{D}, \theta^t) \log \left(P(q \mid \mathcal{D}, \theta) \cdot p(\mathcal{D} \mid \theta) \right) \\ &= \left[\sum_{q \in Q} P(q \mid \mathcal{D}, \theta^t) \log P(q \mid \mathcal{D}, \theta) \right] + \log p(\mathcal{D} \mid \theta) \end{aligned} \tag{12}$$

Proof of Convergence (cont.)

- then if we evaluate it at θ^t

$$A(\theta^t, \theta^t) = \left[\sum_{q \in Q} P(q \mid \mathcal{D}, \theta^t) \log P(q \mid \mathcal{D}, \theta^t)\right] + \log p(\mathcal{D} \mid \theta^t) \tag{13}$$

- the difference between two consecutive log likelihoods of the data can be written as

$$\log p(\mathcal{D} \mid \theta) - \log p(\mathcal{D} \mid \theta^t) =$$
$$A(\theta, \theta^t) - A(\theta^t, \theta^t) + \left[\sum_{q \in Q} P(q \mid \mathcal{D}, \theta^t) \log \frac{P(q \mid \mathcal{D}, \theta^t)}{P(q \mid \mathcal{D}, \theta)}\right] \tag{14}$$

Proof of Convergence (cont.)

- Hence,
    - since the last part of the equation is a **Kullback-Leibler divergence** which is always positive or null,
    - if A increases, the log likelihood of the data also increases
    - Moreover, one can show that when A **is maximum**, the **likelihood of the data** is also at a **maximum**.

EM for Coins

A coin experiment

Estimate the **bias** of two coins:

- Chosen one of the two coins at random.
- Flipped that same coin 10 times.

How can you provide a reasonable estimate of each coin bias? Let's refer to these coins as coin A and coin B and their bias as θ_A and θ_B.

We see which coin is flipped

Maximum likelihood

Coin	Tosses	Coin A	Coin B
B	H T T T H H T H T H		5 H, 5 T
A	H H H H T H H H H H	9 H, 1 T	
A	H T H H H H H T H H	8 H, 2 T	
B	H T H T T T H H T T		4 H, 6 T
A	T H H H T H H H T H	7 H, 3 T	
		24 H, 6 T	9 H, 11 T

5 sets, 10 tosses per set

$$\hat{\theta}_A = \frac{24}{24+6} = 0.80$$

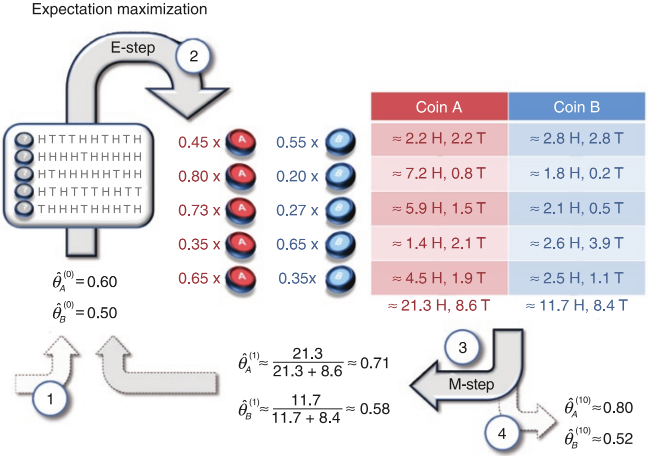
$$\hat{\theta}_B = \frac{9}{9+11} = 0.45$$

We don't see which coin is flipped

Using EM algorithm

1. EM starts with an initial guess of the parameters.
2. In the E-step, a probability distribution over possible completions is computed using the current parameters. The counts shown in the table are the expected numbers of heads and tails according to this distribution.
3. In the M-step, new parameters are determined using the current completions.
4. After several repetitions of the E-step and M-step, the algorithm converges.

We don’t see which coin is flipped (cont.)



Loop	θ_A	θ_B
0	0.60	0.50
1	0.71	0.58
2	0.75	0.57
3	0.77	0.55
4	0.78	0.53
5	0.79	0.53
6	0.79	0.52
7	0.80	0.52
8	0.80	0.52
9	0.80	0.52
10	0.80	0.52

EM for GMMs

- Hidden Variable
- Auxiliary Function
- E-Step
- M-Step

EM for GMM: Hidden Variable

- For GMM, the hidden variable Q will describe **which Gaussian generated each example**.
- If Q was observed, then it would be simple to maximize the likelihood of the data: simply estimate the parameters Gaussian by Gaussian
- Moreover, we will see that we can **easily estimate** Q
- Let us first write the mixture of Gaussian model for one x_i:

$$p(x_i \mid \boldsymbol{\theta}) = \sum_{k=1}^{K} P(k \mid \boldsymbol{\theta}) p_k(x_i|\boldsymbol{\theta}) \tag{15}$$

- Let us now introduce the following **indicator variable**:

$$q_{i,k} = \begin{cases} 1 & \text{if Gaussian } k \text{ emitted } x_i \\ 0 & \text{otherwise} \end{cases}$$

EM for GMM: Auxiliary Function

- We can now write the joint likelihood of all the $\mathcal{D}$ and q:

$$p(\mathcal{D}, Q \mid \boldsymbol{\theta}) = \prod_{i=1}^{n} \prod_{k=1}^{K} P(k \mid \boldsymbol{\theta})^{q_{i,k}} p(x_i \mid k, \boldsymbol{\theta})^{q_{i,k}} \tag{16}$$

- which in log gives

$$\log p(\mathcal{D}, Q \mid \boldsymbol{\theta}) = \sum_{i=1}^{n} \sum_{k=1}^{K} q_{i,k} \log P(k \mid \boldsymbol{\theta}) + q_{i,k} \log p(x_i \mid k, \boldsymbol{\theta}) \tag{17}$$

EM for GMM: Auxiliary Function (cont.)

- Let us now write the corresponding auxiliary function:

$$
\begin{aligned}
A(\boldsymbol{\theta}, \boldsymbol{\theta}^t) &= \mathbb{E}_Q \left[\log p(\mathcal{D}, Q \mid \boldsymbol{\theta}) \mid \mathcal{D}, \boldsymbol{\theta}^t\right] \qquad (18) \\
&= \mathbb{E}_Q \left[\sum_{i=1}^{n} \sum_{k=1}^{K} q_{i,k} \log P(k \mid \boldsymbol{\theta}) + q_{i,k} \log p(x_i \mid k, \boldsymbol{\theta}) \mid \mathcal{D}, \boldsymbol{\theta}^t\right] \\
&= \sum_{i=1}^{n} \sum_{k=1}^{K} \mathbb{E}_Q[q_{i,k} \mid \mathcal{D}, \boldsymbol{\theta}^t] \log P(k \mid \boldsymbol{\theta}) + \mathbb{E}_Q[q_{i,k} \mid \mathcal{D}, \boldsymbol{\theta}^t] \log p(x_i \mid k, \boldsymbol{\theta})
\end{aligned}
$$

E-Step

- Hence, the **E-Step** estimates the posterior:

$$\mathbb{E}_Q[q_{i,k} \mid \mathcal{D}, \boldsymbol{\theta}^t] = 1 \cdot P(q_{i,k} = 1 \mid \mathcal{D}, \boldsymbol{\theta}^t) + 0 \cdot P(q_{i,k} = 0 \mid \mathcal{D}, \boldsymbol{\theta}^t) \tag{19}$$

$$= P(k \mid x_i, \boldsymbol{\theta}^t) = \frac{p(x_i \mid k, \boldsymbol{\theta}^t) P(k \mid \boldsymbol{\theta}^t)}{p(x_i \mid \boldsymbol{\theta}^t)}$$

M-Step

- **M-step** finds the parameters $\boldsymbol{\theta} = \{\mu, \sigma^2, w\}$ that maximizes A, hence searching for
$$\frac{\partial A}{\partial \theta} = 0$$
for each parameter (means μ_k, variances σ_k^2, and weights w_k).
- Note however that $\{w_k\}_{k=1}^{K}$ should sum to 1.

M-Step for Means

$$A(\boldsymbol{\theta}, \boldsymbol{\theta}^t) = \sum_{i=1}^{n} \sum_{k=1}^{K} \mathbb{E}_Q[q_{i,k} \mid \mathcal{D}, \boldsymbol{\theta}^t] \log P(k \mid \boldsymbol{\theta}) + \mathbb{E}_Q[q_{i,k} \mid \mathcal{D}, \boldsymbol{\theta}^t] \log p(x_i \mid k, \boldsymbol{\theta}) \tag{20}$$

$$\begin{aligned}
\frac{\partial A}{\partial \mu_k} &= \sum_{i=1}^{n} \frac{\partial A}{\partial \log p(x_i \mid k, \theta)} \frac{\partial \log p(x_i \mid k, \theta)}{\partial \mu_k} \\
&= \sum_{i=1}^{n} P(k \mid x_i, \theta^t) \frac{\partial \log p(x_i \mid k, \theta)}{\partial \mu_k} \\
&= \sum_{i=1}^{n} P(k \mid x_i, \theta^t) \frac{(x_i - \mu_k)}{\sigma_k^2} = 0
\end{aligned}$$

M-Step for Means (cont.)

- removing constant terms in the sum

$$\sum_{i=1}^{n} P(k \mid x_i, \theta^t) x_i - \sum_{i=1}^{n} P(k \mid x_i, \theta^t) \mu_k = 0$$

$$\hat{\mu}_k = \frac{\sum_{i=1}^{n} P(k \mid x_i, \theta^t) x_i}{\sum_{i=1}^{n} P(k \mid x_i, \theta^t)}$$

M-Step for Variances

$$A(\boldsymbol{\theta}, \boldsymbol{\theta}^t) = \sum_{i=1}^{n} \sum_{k=1}^{K} \mathbb{E}_Q[q_{i,k} \mid \mathcal{D}, \boldsymbol{\theta}^t] \log P(k \mid \boldsymbol{\theta}) + \mathbb{E}_Q[q_{i,k} \mid \mathcal{D}, \boldsymbol{\theta}^t] \log p(x_i \mid k, \boldsymbol{\theta})$$

$$\begin{aligned}
\frac{\partial A}{\partial \sigma_k^2} &= \sum_{i=1}^{n} \frac{\partial A}{\partial \log p(x_i \mid k, \theta)} \frac{\partial \log p(x_i \mid k, \theta)}{\partial \sigma_k^2} \\
&= \sum_{i=1}^{n} P(k \mid x_i, \theta^t) \frac{\partial \log p(x_i \mid k, \theta)}{\partial \sigma_k^2} \\
&= \sum_{i=1}^{n} P(k \mid x_i, \theta^t) \left(\frac{(x_i - \mu_k)^2}{2\sigma_k^4} - \frac{1}{2\sigma_k^2} \right) = 0
\end{aligned}$$

$$\hat{\sigma}_k^2 = \frac{\sum_{i=1}^{n} P(k \mid x_i, \theta^t)(x_i - \hat{\mu}_k)^2}{\sum_{i=1}^{n} P(k \mid x_i, \theta^t)}$$

M-Step for Weights

- We have the constraint that all weights w_k should be positive and sum to 1:

$$\sum_{k=1}^{K} w_k = 1$$

- Incorporating it into the system:

$$J(\theta, \theta^t) = A(\theta, \theta^t) + \left(1 - \sum_{k=1}^{K} w_k\right) \lambda_k$$

where λ_k are Lagrange multipliers.

M-Step for Weights (cont.)

- So we need to derive J with respect to w_k and to set it to 0.

$$\frac{\partial J}{\partial w_k} = \frac{\partial J}{\partial A(\theta, \theta^t)} \frac{\partial A(\theta, \theta^t)}{\partial w_k} - \lambda_k$$
$$= 1 \cdot \left(\sum_{i=1}^{n} P(k \mid x_i, \theta^t) \cdot \frac{1}{w_k} \right) - \lambda_k = 0$$

$$\hat{w}_k = \frac{\sum_{i=1}^{n} P(k \mid x_i, \theta^t)}{\lambda_k}$$

- and incorporating the probabilistic constraint, we get

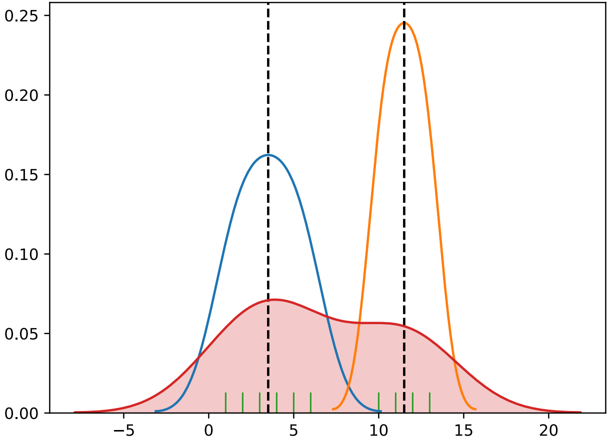
$$\hat{w}_k = \frac{\sum_{i=1}^{n} P(k \mid x_i, \theta^t)}{\sum_{j=1}^{K} \sum_{i=1}^{n} P(j \mid x_i, \theta^t)} = \frac{1}{n} \sum_{i=1}^{n} P(k \mid x_i, \theta^t)$$

Update Rules

$$
\begin{aligned}
&\text{Means} && \hat{\mu}_k = \frac{\sum_{i=1}^{n} P(k \mid x_i, \theta^t) x_i}{\sum_{i=1}^{n} P(k \mid x_i, \theta^t)} \\
&\text{Variances} && \hat{\sigma}_k^2 = \frac{\sum_{i=1}^{n} P(k \mid x_i, \theta^t)(x_i - \hat{\mu}_k)^2}{\sum_{i=1}^{n} P(k \mid x_i, \theta^t)} \\
&\text{Weights} && \hat{w}_k = \frac{1}{n} \sum_{i=1}^{n} P(k \mid x_i, \theta^t)
\end{aligned}
\tag{21}
$$

Example

- Data $\mathcal{D} = \{1, 2, 3, 4, 5, 6, 10, 11, 12, 13\}$ generated by two Gaussians $\mathcal{N}(\mu_1, \sigma_1)$ and $\mathcal{N}(\mu_2, \sigma_2)$



Example

- Estimate the most likely Gaussians $\mathcal{N}(\mu_1, \sigma_1)$ and $\mathcal{N}(\mu_2, \sigma_2)$ from $\mathcal{D}$

Loop	μ_1	σ_1	μ_2	σ_2
0	2	1	11	1
1	3.495413364585706	1.7060277624010254	11.48493211841284	1.152919810380393
2	3.5012090905616713	1.710016971284593	11.500336746451783	1.1180916981781446
3	3.5013264210256705	1.7102090502730485	11.500411717617954	1.1179889840886608
4	3.5013290606968535	1.7102137326222728	11.500412673611843	1.1179885261658662
5	3.5013291208365427	1.7102138400004387	11.50041269402295	1.1179885191522907
6	3.501329122208362	1.7102138424510698	11.500412694486043	1.1179885189985401
7	3.5013291222396563	1.710213842506978	11.500412694496601	1.1179885189950438
8	3.5013291222403704	1.7102138425082534	11.500412694496841	1.1179885189949643
9	3.501329122240387	1.7102138425082827	11.500412694496847	1.1179885189949623
10	3.501329122240387	1.7102138425082831	11.500412694496848	1.1179885189949623
11	3.501329122240387	1.7102138425082831	11.500412694496848	1.1179885189949623

Practical Issues

- Initialization
- Capacity Control
- Adaptation

Initialization

- EM is an iterative procedure that is very sensitive to initial conditions!
- Start from trash $\rightarrow$ end up with trash.
- Hence, we need a good and fast initialization procedure.
- Often used: K-Means.
- Other options: hierarchical K-Means, Gaussian splitting.

Capacity Control

- How to control the **capacity** with GMMs?
  - selecting the number of Gaussians
  - constraining the value of the variances to be far from 0 (small variances $\Longrightarrow$ large capacity)
- Use cross-validation on the desired criterion (Maximum Likelihood, classification...)

Adaptation Techniques

- In some cases, you have access to only a few examples coming from the target distribution...
- ... but many coming from a nearby distribution!
- How can we profit from the big nearby dataset?
- Solution: use adaptation techniques.
- The most well known and used for GMMs: the Maximum A Posteriori adaptation.

MAP Adaptation

- Normal maximum likelihood training for a dataset $\mathcal{D}$:

$$\theta^* = \arg\max_\theta p(\mathcal{D} \mid \theta) \tag{22}$$

- Maximum A Posteriori (MAP) training:

$$\begin{aligned}\theta^* &= \arg\max_\theta p(\theta \mid \mathcal{D}) \\ &= \arg\max_\theta \frac{p(\mathcal{D} \mid \theta) p(\theta)}{p(\mathcal{D})} \\ &= \arg\max_\theta p(\mathcal{D} \mid \theta) p(\theta)\end{aligned} \tag{23}$$

where $p(\theta)$ represents your prior belief about the distribution of the parameters θ.

Implementation

- Which kind of prior distribution for $p(\theta)$?
- Two objectives:
  - constraining θ to reasonable values
  - keep the EM algorithm tractable
- Use **conjugate priors**:
  - Dirichlet distribution for weights
  - Gaussian densities for means and variances

What is a Conjugate Prior?

- A conjugate prior is chosen such that the corresponding **posterior** belongs to the same functional family as the prior.
- So we would like that $p(X \mid \theta)p(\theta)$ is distributed according to the same **family** as $p(\theta)$ and tractable.

Example

- Likelihood is Gaussian

$$p(X \mid \theta) = K_1 \exp\left(-\frac{(x_1 - \mu_1)^2}{2\sigma_1^2}\right)$$

- Prior is Gaussian

$$p(\theta) = K_2 \exp\left(-\frac{(x_2 - \mu_2)^2}{2\sigma_2^2}\right)$$

- Posterior is Gaussian

$$\begin{aligned} p(X \mid \theta)p(\theta) &= K_1 K_2 \exp\left(-\frac{(x_1 - \mu_1)^2}{2\sigma_1^2} - \frac{(x_2 - \mu_2)^2}{2\sigma_2^2}\right) \\ &= K_3 \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right) \end{aligned}$$

Conjugate Prior of Multinomials

- Multinomial distribution:

$$p(X_1 = x_1, \ldots, X_K = x_K \mid \theta) = \binom{n}{x_1 \cdots x_K} \prod_{k=1}^{K} \theta_k^{x_k} \tag{24}$$

  where x_k are nonnegative integers and $\sum_{k=1}^{K} x_k = n$
- Dirichlet distribution with parameter u:

$$P(\theta \mid u) = \frac{1}{Z(u)} \prod_{k=1}^{K} \theta_k^{u_k - 1} \tag{25}$$

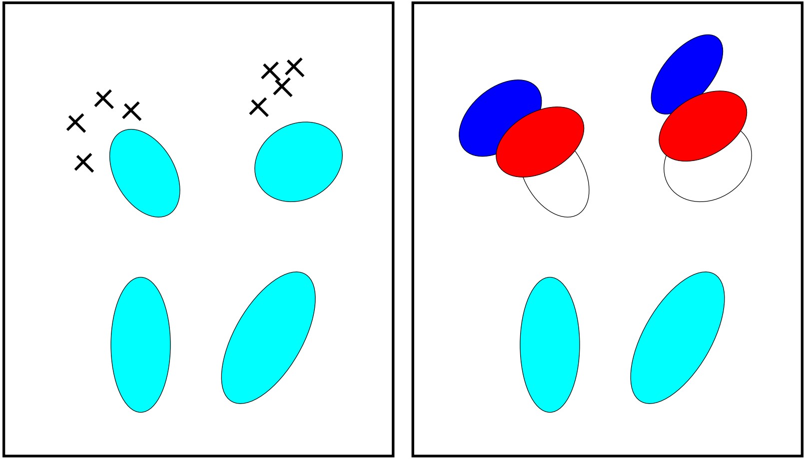
  where $\theta_1, \ldots, \theta_K \geq 0$ and $\sum_{k=1}^{K} \theta_k = 1$ and $u_1, \ldots, u_K \geq 0$
- Conjugate prior $=$ dirichlet with parameter $x + u$:

$$P(X, \theta \mid u) = \frac{1}{Z} \prod_{k=1}^{K} \theta_k^{x_k + u_k - 1} \tag{26}$$

Examples of Conjugate Priors

likelihood $p(\mathcal{D} \mid \theta)$	**conjugate prior** $p(\theta)$	**posterior** $p(\theta \mid \mathcal{D})$
Gaussian	Gaussian	Gaussian
Binomial	Beta	Beta
Poisson	Gamma	Gamma
Multinomial	Dirichlet	Dirichlet

Simple Implementation for MAP-GMMs



Simple Implementation

- Train a generic **prior** model p with large amount of available data

$$\Longrightarrow \{w_k^p, \mu_k^p, \sigma_k^p\}$$

- One hyper-parameter: $\alpha \in [0, 1]$: faith on prior model
- Weights:

$$\hat{w}_k = \left[\alpha w_k^p + (1-\alpha)\sum_{i=1}^{n} P(k \mid x_i)\right]\gamma$$

where γ is a normalization factor (so that $\sum_k w_k = 1$)

- Means:

$$\hat{\mu}_k = \alpha \mu_k^p + (1-\alpha)\frac{\sum_{i=1}^{n} P(k \mid x_i)x_i}{\sum_{i=1}^{n} P(k \mid x_i)}$$

Simple Implementation (cont.)

- Variances:

$$\hat{\sigma}_k = \alpha(\sigma_k^p + \mu_k^p \mu_k^p) + (1-\alpha)\frac{\sum_{i=1}^n P(k \mid x_i) x_i x_i}{\sum_{i=1}^n P(k \mid x_i)} - \mu_k^p \mu_k^p$$

Adapted GMMs for Person Authentication

- Person authentication task:

$$\text{accept access if } P(S_i \mid \mathcal{D}) > P(\bar{S}_i \mid \mathcal{D})$$

  with S_i a client, $\bar{S}_i$ all the other persons, and $\mathcal{D}$ an access attributed to S_i.
- Using Bayes theorem, this becomes:

$$\frac{P(\mathcal{D} \mid S_i)}{P(\mathcal{D} \mid \bar{S}_i)} > \frac{P(\bar{S}_i)}{P(S_i)} = \Delta_{S_i} \approx \Delta$$

- $P(\mathcal{D} \mid \bar{S}_i)$ is trained on a large dataset.
- $P(\mathcal{D} \mid S_i)$ is MAP adapted from $P(\mathcal{D} \mid \bar{S}_i)$.
- Δ is found on a separate validation set to optimize a given criterion.

References

Goodfellow, I., Bengio, Y., and Courville, A. (2016).
Deep learning.
MIT press.

Lê, B. and Tô, V. (2014).
Cở sở trí tuệ nhân tạo.
Nhà xuất bản Khoa học và Kỹ thuật.

Russell, S. and Norvig, P. (2021).
Artificial intelligence: a modern approach.
Pearson Education Limited.