Systematic Trading Strategies with Machine Learning Algorithms

Introduction to Unsupervised Learning Techniques
Optional Lecture Notes

24 April 2025

Contents

1 Clustering methods

1.1 Motivation

Given a data set $X = (x_1, \dots, x_n) \in \mathbb{R}^{n \times p}$ where n is the number of observation and p is the number of features, we want to separate these data into K classes (clusters), i.e.we want to learn :

- the centroid (center) of each cluster $\{c_1, \dots, c_K\} \in \mathbb{R}^{p \times K}$
- an assignation function $\Psi : \{x_1, \dots, x_n\} \in \mathbb{R}^{n \times p} \to \{1, \dots, K\}$, meaning "sample x_i belongs to class $\Psi(x_i)$ ".

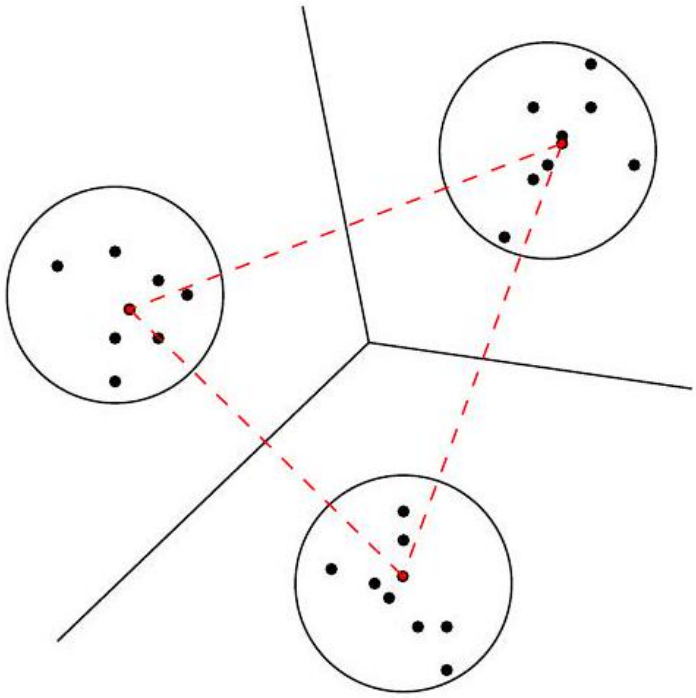

Figure 1: A simple representation of the situation $(n = 25, p = 2, K = 3)$

1.2 The K-means Clustering Algorithm

Definition 1.1. *The algorithm 1 is called the K-means algorithm. It's an iterative algorithm that provides an assignment function Ψ^* and the associated centroids $c_1^*, \ldots, c_K^*$.*

Algorithm 1 The K-means Algorithm

Require: A data set $X = \{x_1, \ldots, x_n\}$ $(x_i \in \mathbb{R}^p)$
Ensure: An assignment function Ψ^* and the associated centroids $c_1^*, \ldots, c_K^*$.

```
Initialization: Choose c_1, ..., c_K in X at random
repeat
    Assignment step:
    for i = 1 ... n do
        Ψ(x_i) ← arg min_{k∈{1,...,K}} ||x_i − c_k||^2
    end for
    Re-estimation step:
    for k = 1 ... K do
        c_j ← 1 / (Σ_{i=1}^n 1(Ψ(x_i)=k)) Σ_{i=1}^n 1 (Ψ(x_i) = k) x_i
    end for
until convergence
return Ψ*, c_1^*, ..., c_K^*
```

Definition 1.2. *Let Ψ be an assignation function and $c = (c_1, \ldots, c_K)$ be the corresponding centroids. We define the distortion $J(\Psi, c)$ as follows:*

$$J(\Psi, c) = \frac{1}{n} \sum_{i=1}^{n} \left\| x_i - c_{\Psi(x_i)} \right\|^2$$

Theorem 1.2.1. *The K-means algorithm 1 monotonically decreases the distortion*

Exercise:

For each iteration t of the algorithm 1, we define the distortion at time t as :

$$J(\Psi^{(t)}, c^{(t)}) = \frac{1}{n} \sum_{i=1}^{n} \left\| x_i - c^{(t)}_{\Psi^{(t)}(x_i)} \right\|^2$$

Show that for all t:

$$J(\Psi^{(t)}, c^{(t)}) \geq J(\Psi^{(t+1)}, c^{(t+1)})$$

Solution:

We have:

$$\Psi^{t+1}(x_i) = \underset{k \in \{1,\dots,K\}}{\arg\min} \left\| x_i - c_k^{(t)} \right\|^2$$

So,

$$\begin{aligned} J(\Psi^{(t)}, c^{(t)}) &= \frac{1}{n} \sum_{i=1}^{n} \left\| x_i - c^{(t)}_{\Psi^{(t)}(x_i)} \right\|^2 \\ &\geq \frac{1}{n} \sum_{i=1}^{n} \left\| x_i - c^{(t)}_{\Psi^{(t+1)}(x_i)} \right\|^2 \\ &= \frac{1}{n} \sum_{i=1}^{n} \sum_{k=1}^{K} z_i^k \left\| x_i - c_k^{(t)} \right\|^2 \end{aligned} \tag{1}$$

where $z = (z_i^k)_{(i,k) \in \{1,\dots,n\} \times \{1,\dots,K\}}$ is such that:

$$\forall (i,k) \in \{1,\dots,n\} \times \{1,\dots,K\} \quad z_i^k = \mathbf{1}\left(\Psi^{(t+1)}(x_i) = k\right)$$

We define:

$$\forall c = (c_1, \dots, c_K) \in \mathbb{R}^{K \times p} \quad \mathcal{L}(c) := \frac{1}{n} \sum_{i=1}^{n} \sum_{k=1}^{K} z_i^k \left\| x_i - c_k \right\|^2$$

It's straightforward that:

$$\mathcal{L}(c^{(t)}) = \frac{1}{n} \sum_{i=1}^{n} \left\| x_i - c^{(t)}_{\Psi^{(t+1)}(x_i)} \right\|^2 \tag{2}$$

and

$$\mathcal{L}(c^{(t+1)}) = \frac{1}{n}\sum_{i=1}^{n}\left\|x_i - c^{(t+1)}_{\Psi^{(t+1)}(x_i)}\right\|^2 \tag{3}$$

We wish to minimize $\mathcal{L}$ w.r.t c.

We have:

$$\forall k \in \{1,\dots,K\} \quad \nabla_{c_k}\mathcal{L}(c) = \frac{1}{n}\sum_{i=1}^{n} z_i^k \nabla_{c_k} \underbrace{\left(\|x_i - c\|^2\right)}_{g\circ f(c)} \tag{4}$$

where:

$$f : c \mapsto x_i - c \quad \text{and} \quad g : y \mapsto \|y\|^2$$

We have

$$\begin{aligned} d(g\circ f)_a(h) &= dg_{f(a)}\left(df_a(h)\right) \\ &= dg_{f(a)}(-h) \\ &= \langle 2f(a), -h\rangle \\ &= \langle -2(x_i - a), h\rangle \end{aligned}$$

Thus,

$$\nabla_{c_k} g\circ f(c) = -2(x_i - c) \tag{5}$$

From 4 and 5, we conclude that:

$$\forall k \in \{1,\dots,K\} \quad \nabla_{c_k}\mathcal{L}(c) = \frac{-2}{n}\sum_{i=1}^{n} z_i^k (x_i - c)$$

Therefore,

$$\begin{aligned}
\nabla_c \mathcal{L}(c) = 0 &\iff \forall k \in \{1, \dots, K\} \quad \nabla_{c_k} \mathcal{L}(c) = 0 \\
&\iff \forall k \in \{1, \dots, K\} \quad c_k = \frac{\sum_{i=1}^{n} z_i^k x_i}{\sum_{i=1}^{n} z_i^k} \\
&\iff \forall k \in \{1, \dots, K\} \quad c_k = c_k^{(t+1)} \\
&\iff c = c^{(t+1)}
\end{aligned}$$

We conclude that

$$\forall c = (c_1, \dots, c_K) \in \mathbb{R}^{K \times p} \quad \mathcal{L}(c) \geq \mathcal{L}(c^{(t+1)})$$

And therefore:

$$\mathcal{L}(c^{(t)}) \geq \mathcal{L}(c^{(t+1)}) \tag{6}$$

From equations 1, 2, 3 and 6, we conclude that:

$$J(\Psi^{(t)}, c^{(t)}) \geq \frac{1}{n} \sum_{i=1}^{n} \left\| x_i - c_{\Psi^{(t+1)}(x_i)}^{(t+1)} \right\|^2 = J(\Psi^{(t+1)}, c^{(t+1)})$$

Corollary 1.2.2. *The K-means algorithm 1 stops after a finite number of steps.*

Proof. The number of possible assignations is finite.

Thus, there exists t such that

$$J(\Psi^{(t)}, c^{(t)}) = J(\Psi^{(t+1)}, c^{(t+1)})$$

□

2 Gaussian Mixture Model

2.1 Introduction

Gaussian Mixture Models (GMMs) are a probabilistic model for representing normally distributed subpopulations within an overall population. Unlike single Gaussian models, which assume that all observations are drawn from a single distribution, GMMs consider a mixture of several Gaussian distributions, each with its own mean and variance, thus providing a more flexible approach to modeling data distributions. This flexibility makes GMMs particularly useful for modeling complex data sets with hidden or latent variables—where observations may originate from one of several unknown subpopulations.

Let's present a simple example to illustrate what we just said. The probability density represented on Figure 2 is akin to an average of two Gaussians. Thus, it is natural to use a mixture model and to introduce an hidden variable z, following a Bernoulli distribution defining which Gaussian the point is sampled from.

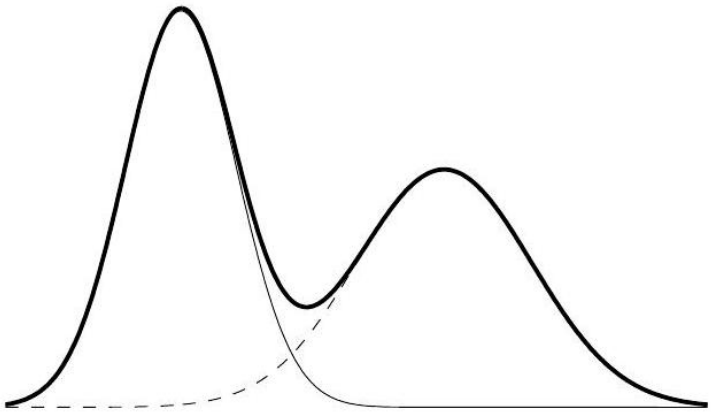

Figure 2: Average of two probability distributions of two Gaussian for which it is natural to introduce a mixture model

Thus we have : $z \in \{1, 2\}$ and $x \mid z = i \sim \mathcal{N}(\mu_i, \Sigma_i)$. The density $p(x)$ is a convex combination of normal density:

$$p(x) = p(x, z = 1) + p(x, z = 2) = p(x \mid z = 1)p(z = 1) + p(x \mid z = 2)p(z = 2)$$

It is a mixture model. It represents a simple way to model complicated phenomena.

2.2 Expectation Maximization Algorithm

2.2.1 Introducing the context

The Expectation-Maximization (EM) algorithm is an iterative method used for obtaining maximum likelihood estimates of parameters within statistical models. These models are characterized by their reliance on unobserved latent variables or hidden variables. Latent variables, denoted as z, are not directly observed but are inferred through the variables that are observed, denoted as x.

Within the context of the EM algorithm, we operate under the following framework:

- **Assumption:** We consider (x, z) to be random variables, where x represents the observed data, and z represents the hidden or latent variables (for example, unknown cluster centers in a clustering problem). The joint density function of x and z, $p_\theta(x, z)$, is parameterized by θ, indicating the model's parameters.
- **Objective:** The primary goal is to maximize the marginal likelihood of the observed data x with respect to the parameters θ, expressed as:

$$\max_\theta p_\theta(x) = \sum_z p_\theta(x, z)$$

This objective highlights the challenge posed by the presence of latent variables: maximizing the marginal likelihood is not straightforward due to the summation over the latent variable z. The summation introduces complexities, making the problem more challenging than optimizing a likelihood function without latent variables.

Specifically, taking the logarithm of the marginal likelihood does not lead to a simple convex optimization problem. The EM algorithm provides a robust method for addressing this challenge, facilitating the estimation of model parameters in the presence of latent variables.

2.2.2 The EM algorithm

The Dataset is composed of the pairs $(x_i, z_i)_{1 \leq i \leq n}$ where x_i is the observed data and z_i is the hidden data.

We make the assumption that the $(x_i, z_i)_{1 \leq i \leq n}$ are i.i.d.

The aim is to maximize the log likelihood:

$$\log p_\theta(x) = \sum_{i=1}^{n} \log \sum_{z_i} p_\theta(x_i, z_i)$$

We will use the following properties :

Proposition 2.2.1. *Jensen Inequality:*

1. *if $f : \mathbb{R} \to \mathbb{R}$ is convex and if X is an integrable random variable :*

$$\mathbb{E}_X(f(X)) \geq f(\mathbb{E}_X(X))$$

2. *if $f : \mathbb{R} \to \mathbb{R}$ is strictly convex, we have equality in the previous inequality if and only if $X =$ constant a.s.*

The EM algorithm is an iterative method for finding maximum likelihood estimates of parameters in statistical models, where the models depend on unobserved latent variables.

Consider for instance n observations $x_1, \ldots, x_n$ and the latent variables associateed with them $z_1, \ldots, z_n$.

We assume the pairs (x_i, z_i) to be independent and identically distributed.

For $(x, z) = (x_1, z_1, \ldots, x_n, z_n)$, the objective is to maximize:

$$\log(p(x; \theta)) = \sum_{i=1}^{n} \log \left(\sum_{z_i} p(x_i, z_i; \theta) \right)$$

For each $i \in \{1, \ldots, n\}$, we introduce a function $z_i \mapsto q(z_i)$ such that $q(z_i) \geq 0$ and $\sum_{z_i} q(z_i) = 1$ in the expression of the likelihood.

By conditioning on a latent variable z_i and using the Jensen inequality, we get a lower bound $\mathcal{L}(q, \theta)$ that depends on both q and θ.

$$\begin{aligned}
\log(p(x;\theta)) &= \sum_{i=1}^{n} \log\left(\sum_{z_i} p(x_i, z_i; \theta)\right) \\
&= \sum_{i=1}^{n} \log\left(\sum_{z_i} q(z_i) \frac{p(x_i, z_i; \theta)}{q(z_i)}\right) \\
&\geq \sum_{i=1}^{n} \sum_{z_i} q(z_i) \log\left(\frac{p(x_i, z_i; \theta)}{q(z_i)}\right) \\
&= \sum_{i=1}^{n} \underbrace{\mathbb{E}_{q(z_i)}\left[\log\left(\frac{p_\theta(x_i, z_i)}{q(z_i)}\right)\right]}_{\mathcal{L}(q(z_i),\theta)}
\end{aligned}$$

The EM algorithm can then be summarized as depicted in 2.

Algorithm 2 EM Algorithm

Require: Data set $X = \{x_1, \ldots, x_n\}$
Ensure: Optimal θ

1: **Initialization:** Choose initial parameters $\theta^{(0)}$.
2: Set iteration counter $i = 0$.
3: **while** not converged **do**
4: **E-step:** Update q to maximize the lower bound with respect to q.

$$q_{t+1} \in \arg\max_q \left(\mathcal{L}(q, \theta_t)\right)$$

5: **M-step:** Update θ to maximize the lower bound with respect to θ.

$$\theta_{t+1} \in \arg\max_\theta \left(\mathcal{L}(q_{t+1}, \theta)\right)$$

6: Check for convergence criterion (e.g., change in θ below a threshold).
7: $i \leftarrow i + 1$
8: **end while**
9: **return** Optimized parameters θ^*.

Exercise:

Show that the gap beween the marginal log-likelihood and the **lower bound** $\sum_{i=1}^{n} \mathcal{L}(q(z_i), \theta)$ is reduced to 0 when $q(z_i) = p_\theta(z_i \mid x_i)\ \forall i \in \{1, \dots, n\}$.

$p_\theta(z_i \mid x_i)$ is called the **posterior distribution**

Solution: Let $d = \log(p_\theta(x)) - \sum_{i=1}^{n} \mathcal{L}(q(z_i), \theta)$.

We have:

$$
\begin{aligned}
\mathrm{d} &= \log(p_\theta(x)) - \sum_{i=1}^{n} \mathcal{L}(q(z_i), \theta) \\
&= \sum_{i=1}^{n} \left(\log(p_\theta(x_i)) - \mathcal{L}(q(z_i), \theta)\right) \\
&= \sum_{i=1}^{n} \left(\sum_{z_i} q(z_i) \log(p_\theta(x_i)) - \sum_{z_i} q(z_i) \log\left(\frac{p_\theta(x_i, z_i)}{q(z_i)}\right)\right) \\
&= \sum_{i=1}^{n} \sum_{z_i} q(z_i) \left(\log(p_\theta(x_i)) - \log\left(\frac{p_\theta(x_i, z_i)}{q(z_i)}\right)\right) \\
&= \sum_{i=1}^{n} \sum_{z_i} q(z_i) \log\left(\frac{q(z_i)}{p_\theta(z_i|x_i)}\right) \\
&= \sum_{i=1}^{n} D_{\mathrm{KL}}\left(q(z_i) \,\|\, p_\theta(z_i|x_i)\right)
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
d = 0 &\iff \sum_{i=1}^{n} \underbrace{D_{\mathrm{KL}}\left(q(z_i) \,\|\, p_\theta(z_i|x_i)\right)}_{\geq 0} \\
&\iff \forall i \in \{1, \dots, n\} \quad D_{\mathrm{KL}}\left(q(z_i) \,\|\, p_\theta(z_i|x_i)\right) \\
&\iff \forall i \in \{1, \dots, n\} \quad q(z_i) = p_\theta(z_i|x_i)
\end{aligned}
$$

Therefore, maximizing the lower bound $\log(p_\theta(x))$ with respect to q consists in taking the posterior distributions $\forall i \in \{1, \ldots, n\} \quad q(z_i) = p_\theta(z_i|x_i)$.

Let's recall the expression of the lower bound:

$$\mathcal{L}(q, \theta) = \sum_{i=1}^{n} \left(\sum_{z_i} q(z_i) \log p_\theta(x_i, z_i) - \sum_{z_i} q(z_i) \log q(z_i) \right)$$

Since $\sum_{z_i} q(z_i) \log q(z_i)$ doesn't depend on θ, maximizing the lower bound with respect to θ is equivalent to maximizing w.r.t θ the expected value of the complete log likelihood function $\log(p_{\theta_t}(x, z))$.

The final recipe is given in algorithm 3. It consists in the following steps:

1. Compute the probability of Z given X : $p_{\theta_t}(z \mid x)$ (Corresponding to $q_{t+1} = \arg\max_q \mathcal{L}(q, \theta_t)$)
2. Write the complete loglikelihood $l_c = \log(p_{\theta_t}(x, z))$.
3. **E-Step**: Calculate the expected value of the complete log likelihood function, with respect to the conditional distribution of Z given X under the current estimate of the parameter $\theta_t : \mathbb{E}_{Z|X}(l_c)$.
4. **E-Step**: Find θ_{t+1} by maximizing $\mathcal{L}(q_{t+1}, \theta)$ with respect to θ.

Algorithm 3 EM algorithm

Require: Observations $x_1, \ldots, x_n$
Ensure: Optimal θ

Initialize $\theta^{(0)}$
while not converged **do**
    E-step: $q(z) = p(z|x; \theta^{(i-1)})$
    M-step: $\theta^{(i)} = \arg\max_\theta \mathbb{E}_q[\log p(x, z; \theta)]$
    $i \leftarrow i + 1$
end while

Remarks:

- It is an ascent algorithm, indeed it goes up in term of likelihood (compare to before where we were descending along the distortion).
- The sequence of log-likelihoods converges.

- It does not converge to a global maximum but rather to a local maximum because we are dealing here with a non-convex problem. An illustration is given in Figure 3

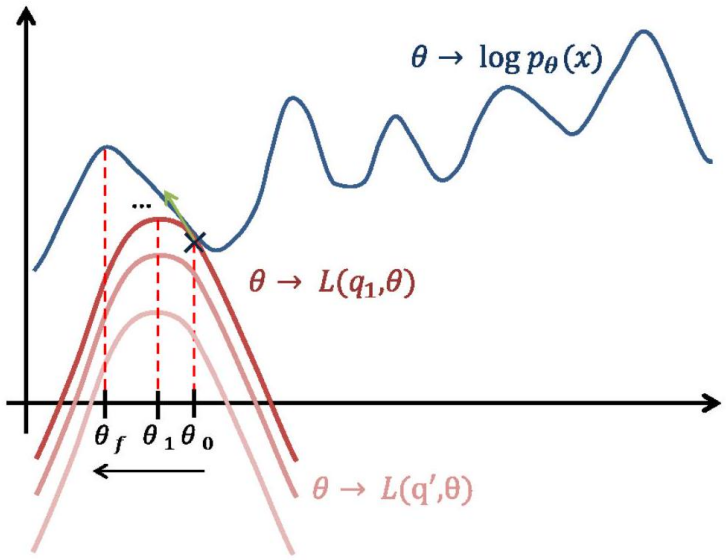

Figure 3: An illustration of the EM algorithm that converges to a local minimum.

- As it was already the case for K-means, we reiterate the result in order to be more confident. Then we keep the one with the highest likelihood.
- Because EM gives a local maximum, it is clever to choose a θ_0 relatively close to the final solution. For Gaussian mixtures, it is quite usual to initiate EM by a K-means.

Exercise:

Suppose we have observations n observations $x_1, \dots x_n$ in $\mathbb{R}^p$.

We make the assumption of the existence of latent variables $z_1, \dots, z_n$ from a multinomial distribution with K possible outcomes.

i.e:

$\forall i \in \{1, \dots, n\} \quad x_i \in \mathbb{R}^p, z_i \sim \mathcal{M}(1, \pi_1, \dots, \pi_K)$ and $(x_i \mid z_i = j) \sim \mathcal{N}(\mu_j, \Sigma_j)$.

Here we have $\theta = (\pi, \mu, \Sigma)$.

Use the EM algorithm to estimate θ.

Solution:

1. **Calculation of the posterior distributions** $p_\theta(z_i \mid x_i)$:
   We write $p_\theta(x_i)$:

$$\begin{aligned} p_\theta(x_i) &= \sum_{z_i} p_\theta(x_i, z_i) = \sum_{z_i} p_\theta(x_i \mid z_i)\, p_\theta(z_i) \\ &= \sum_{j=1}^{K} p_\theta(x_i \mid z_i = j)\, p_\theta(z_i = j) \end{aligned}$$

   Then we use the Bayes formula to estimate $p_\theta(z \mid x)$:

$$\begin{aligned} p_\theta(z_i = j \mid x_i) &= \frac{p_\theta(x_i \mid z_i = j)\, p_\theta(z_i = j)}{p_\theta(x_i)} \\ &= \frac{\pi_j \mathcal{N}(x_i \mid \mu_j, \Sigma_j)}{\sum_{j'} \pi_{j'} \mathcal{N}\left(x_i \mid \mu'_j, \Sigma'_j\right)} \\ &= \tau_i^j(\theta). \end{aligned}$$

   We recall that $\mathcal{N}(x_i \mid \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{p}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right)$.
   Suppose that we are at the t-th iteration of the algorithm.

2. **Complete likelihood**
   Let's write the complete likelihood of the problem.

$$
\begin{aligned}
l_{c,t} = \log p_{\theta_t}(x, z) &= \sum_{i=1}^{n} \log p_{\theta_t}\left(x_i, z_i\right) \\
&= \sum_{i=1}^{n} \log\left(p_{\theta_t}\left(z_i\right) p_{\theta_t}\left(x_i \mid z_i\right)\right) \\
&= \sum_{i=1}^{n} \log\left(p_{\theta_t}\left(z_i\right)\right) + \log\left(p_{\theta_t}\left(x_i \mid z_i\right)\right) \\
&= \sum_{i=1}^{n} \sum_{j=1}^{K} z_i^j \log\left(\pi_{j,t}\right) \\
&\quad + \sum_{i=1}^{n} \sum_{j=1}^{K} z_i^j \log\left(\mathcal{N}\left(x_i \mid \mu_{j,t}, \Sigma_{j,t}\right)\right)
\end{aligned}
$$

where $z_i^j \in \{0, 1\}$ with $z_i^j = 1$ if $z_i = j$ and 0 otherwise.

3. **E-Step** In the E-step, we compute the expectation of the complete log-likelihood with respect to the conditional distribution of the latent variables Z given the observed data X. This involves replacing the indicator variables z_i^j with their expected values:

$$
\mathbb{E}_{Z|X}\left(z_i^j\right) = p_{\theta_t}(z = j|x_i) = \tau_i^j(\theta_t),
$$

where τ_i^j represents the posterior probability that observation x_i belongs to component j, given the current parameter estimates. By substituting z_i^j with τ_i^j, we obtain the expected complete log-likelihood:

$$
\mathbb{E}_{Z|X}(l_{c,t}) = \sum_{i=1}^{n} \sum_{j=1}^{K} \tau_i^j \log(\pi_{j,t}) + \sum_{i=1}^{n} \sum_{j=1}^{K} \tau_i^j \log\left(\mathcal{N}(x_i|\mu_{j,t}, \Sigma_{j,t})\right).
$$

4. **M-Step**

   For the M-step, we this need to maximize:

$$
\begin{aligned}
&\sum_{i=1}^{n} \sum_{j=1}^{K} \tau_i^j \log\left(\pi_{j,t}\right) + \sum_{i=1}^{n} \sum_{j=1}^{K} \tau_i^j \left[\log\left(\frac{1}{(2\pi)^{\frac{p}{2}}}\right) + \log\left(\frac{1}{|\Sigma_{j,t}|^{\frac{1}{2}}}\right)\right. \\
&\left. -\frac{1}{2}\left(x_i - \mu_{j,t}\right)^T \Sigma_{j,t}^{-1}\left(x_i - \mu_{j,t}\right)\right]
\end{aligned}
$$

We want to maximize the previous equation with respect to $\theta_t = (\Pi_t, \mu_t, \Sigma_t)$

As the sum is separated into two terms independent along the variables we can first maximize with respect to π_t :

$$\max_{\Pi} \sum_{j=1}^{k} \sum_{i=1}^{n} \tau_i^j \log \pi_j \quad \Rightarrow \quad \pi_{j,t+1} = \frac{\sum_{i=1}^{n} \tau_i^j}{\sum_{i=1}^{n} \sum_{j'=1}^{k} \tau_i^{j'}} = \frac{1}{n} \sum_{i=1}^{n} \tau_i^j$$

We can now maximize with respect to μ_t and Σ_t. By computing the gradient along the $\mu_{j,t}$ and along the $\Sigma_{j,t}$, we obtain :

$$\mu_{j,t+1} = \frac{\sum_i \tau_i^j x_i}{\sum_i \tau_i^j}$$

$$\Sigma_{j,t+1} = \frac{\sum_i \tau_i^j \left(x_i - \mu_{j,t+1}\right) \left(x_i - \mu_{j,t+1}\right)^T}{\sum_i \tau_i^j}$$

The M-step in the EM algorithm corresponds to the estimation of means step in K-means. Note that the value of τ_i^j in the expressions above are taken for the parameter values of the previous iterate, i.e., $\tau_i^j = \tau_i^j(\theta_t)$.

Possible forms for Σ_j

- isotropic: $\Sigma_j = \sigma_j^2 \mathrm{Id}, 1$ parameter, the cluster is a sphere.
- diagonal: Σ_j is a diagonal matrix, d parameters, the cluster is an ellipse oriented along the axis.
- general: $\Sigma_j, \frac{d(d+1)}{2}$ parameters, the cluster is an ellipse.