Fundamentals of Probabilistic Data Mining Lab 3

Zeinab Abdallah, Lucas Batier, Ionut-Vlad Modoranu, Predrag Pilipovic

February 16, 2020

3 Variational EM algorithm:

Note. Code for this Lab is available in the notebook there:

`https://colab.research.google.com/drive/1ut20f54gULCQ2Sq4E-7O-zKOr-OgrkRX`

In this lab work we will extend the model discussed in class referred to as “GMM with prior means.”

3.1 The model

The model consists of an observation variable $\mathbf{x} \in \mathcal{X} = \mathbb{R}^{d_x}$, that, as in the case of GMM, can be generated from K different Gaussian distributions. The variable denoting from which of these K distributions is $\mathbf{x}$ generated, is denoted by $\mathbf{z} \in \mathcal{Z} = \{1, ..., K\}$. We will assume the existence of N i.i.d. observations $\mathbf{x}_{1:N} = \{\mathbf{x}_n\}_{n=1}^N$ and their corresponding hidden assignment random variables $\mathbf{z}_{1:N} = \{\mathbf{z}_n\}_{n=1}^N$.

In the case of GMM, the mean vectors and covariance matrices are parameters to be estimated. In the model that we consider for this lab work, both the mean vectors and the covariance matrices are going to be hidden random variables. Indeed, we will assume some prior information on the mean vectors and covariance matrices. In addition, to simplify the computations, the covariance matrices of the Gaussian components of this new GMM will be diagonal.

The joint probability factorises as:

$$p(\mathbf{x}_{1:N}, \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta) = \underbrace{p(\mathbf{x}_{1:N} \mid \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta)}_{\textcircled{\mathbf{x}}} \underbrace{p(\mathbf{z}_{1:N}; \theta)}_{\textcircled{\mathbf{z}}} \underbrace{p(\boldsymbol{\mu}_{1:K}; \theta)}_{\textcircled{\boldsymbol{\mu}}} \underbrace{p(\nu_{1:K}; \theta)}_{\textcircled{\nu}} \tag{1}$$

The first and second terms correspond to a GMM:

$$\textcircled{\mathbf{x}} \quad p(\mathbf{x}_{1:N} \mid \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta) = \prod_{n=1}^{N} p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta), \tag{2}$$

$$\text{with} \quad p(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta) = \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_k, \nu_k \mathbf{I}_{dx}). \tag{3}$$

$$\textcircled{\mathbf{z}} \quad p(\mathbf{z}_{1:N}; \theta) = \prod_{n=1}^{N} p(\mathbf{z}_n; \theta), \qquad \text{with} \;\; p(\mathbf{z}_n; \theta) = \pi_n. \tag{4}$$

The third term is the same as in the model discussed in class:

$$\textcircled{\boldsymbol{\mu}} \quad p(\boldsymbol{\mu}_{1:K}; \theta) = \prod_{k=1}^{K} p(\boldsymbol{\mu}_k; \theta), \qquad \text{with} \;\; p(\boldsymbol{\mu}_k; \theta) = \mathcal{N}(\boldsymbol{\mu}_k; \mathbf{m}, \boldsymbol{\Omega}). \tag{5}$$

The last term uses the *inverse gamma* distribution with parameters α (shape) and β (rate).

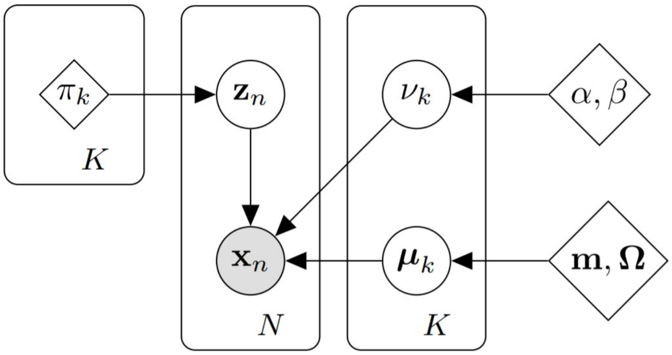
$$\textcircled{\nu} \quad p(\nu_{1:K};\theta) = \prod_{k=1}^{K} p(\nu_k;\theta), \qquad \text{with} \;\; p(\nu_k;\theta) = \mathcal{IG}(\nu_k;\alpha,\beta), \tag{6}$$

where

$$\mathcal{IG}(\nu_k;\alpha,\beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} \nu_k^{-\alpha-1} \exp\left(-\frac{\beta}{\nu_k}\right), \tag{7}$$

and $\Gamma(\alpha) = \int\limits_0^\infty t^{\alpha-1}\exp(-t)dt$ is the Gamma function evaluated at α.

The graphical model is shown in the figure below:



3.2 Preliminary work

The objective of this section is to understand the model, and to verify the equations for the variational E and M steps.

1. **What are the model's parameters, Θ? How many free parameters does the model have?**

   We know that in the standard GMM, the parameters are the mixture weights $\pi_1, ..., \pi_K$, the mean vectors $\boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_K$ and the covariance matrices $\boldsymbol{\Sigma}_1, ..., \boldsymbol{\Sigma}_K$. In this lab-work, we are assuming that the mean vectors follow a normal distribution with mean vector $\mathbf{m}$ and covariance matrix $\boldsymbol{\Omega}$, and the covariance matrices follow the inverse gamma distribution with parameters α, and β as it is mentioned above, and hence we get a set of parameters:

   - $\boldsymbol{\pi} = \{\pi_k\}_{k=1}^K$: mixture weights. Because we know that

   $$\sum_{k=1}^{K} \pi_k = 1,$$

   we only need to know $K-1$ parameters, because the remaining one can be written as a linear combination of the others. Therefore here we have $K-1$ free parameters.

- $\mathbf{m}$: the mean vector of the mean vectors of GMM. The dimension of $\mathbf{m}$ must be the same as the dimension of $\boldsymbol{\mu}_k$, which must be the same as $\mathbf{x}_n$, which is d_x. Thus, $\mathbf{m}$ is of dimension d_x, and the number of free parameters is d_x, because we need to estimate all of them.
- $\boldsymbol{\Omega}$: The covariance matrix of the mean vectors of GMM. It is well known that the covariance matrix is positive semidefinite symmetric matrix, so its enough to know, either the upper side with the diagonal, or the lower side with the diagonal, and in this case the number of free parameters is: $\frac{d_x(d_x+1)}{2}$
- α: the shape of the inverse gamma distribution, it is a scalar, so it is of dimension one and hence one free parameter
- β: the rate of the inverse gamma distribution it is a scalar, so it is of dimension one and hence one free parameter

So Finally the set of parameters are given by:

$$\Theta = (\boldsymbol{\pi}, \mathbf{m}, \boldsymbol{\Omega}, \alpha, \beta),$$

and the number of free parameters are $K - 1 + d_x + \frac{d_x(d_x+1)}{2} + 1 + 1 = K + \frac{d_x(d_x+3)}{2} + 1$.

Given an estimate of the parameters θ^{old} , we will assume that that a posteriori distribution is too complicated and approximate it with the following factorisation.

$$p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}}) \approx q_{\mathbf{z}}(\mathbf{z}_{1:N}) q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K}) q_{\nu}(\nu_{1:K}) \tag{8}$$

2. **E-Z step**: We will now find the expression for $q_{\mathbf{z}}$

   (a) **Prove the separability in n of $q_{\mathbf{z}}$ (no need to use the shape of the distributions)**:

$$q_{\mathbf{z}}\left(\mathbf{z}_{1:N}\right) \overset{(\mathbf{z}_{1:N})}{\propto} \prod_{n=1}^{N} p\left(\mathbf{z}_n; \theta^{\text{old}}\right) \exp\left(\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K}) q_{\nu}(\nu_{1:K})} \left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\}\right) \tag{9}$$

We have the generic expression

$$q_{\mathbf{z}}\left(\mathbf{z}_{1:N}\right) \overset{(\mathbf{z}_{1:N})}{\propto} \exp\left(\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K}) q_{\nu}(\nu_{1:K})} \left\{\log p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}})\right\}\right).$$

We will start with the probability inside the expectation

$$\begin{aligned}
&p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}}) \\
&= p(\mathbf{x}_{1:N} \mid \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}) p(\mathbf{z}_{1:N}; \theta^{\text{old}}) p(\boldsymbol{\mu}_{1:K}; \theta^{\text{old}}) p(\nu_{1:K}; \theta^{\text{old}}) \\
&\overset{(\mathbf{z}_{1:N})}{\propto} p(\mathbf{x}_{1:N} \mid \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}) p(\mathbf{z}_{1:N}; \theta^{\text{old}}) \\
&\overset{(\mathbf{z}_{1:N})}{\propto} \prod_{n=1}^{N} p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}) p(\mathbf{z}_n; \theta^{\text{old}}).
\end{aligned}$$

By taking the logarithm of the above expression, we have:

$$\begin{aligned}\log\{p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}})\} &\overset{(\mathbf{z}_{1:N})}{=} \log\left\{\prod_{n=1}^{N} p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}})p(\mathbf{z}_n; \theta^{\text{old}})\right\}\\ &\overset{(\mathbf{z}_{1:N})}{=} \sum_{n=1}^{N}\{\log p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}) + \log p(\mathbf{z}_n; \theta^{\text{old}})\}\end{aligned}$$

Now, by taking the expected value of the above expression with respect to q_μ and q_ν and using the fact that switching finite sum and the expected value is possible, and the fact that $\mathbf{z}_n$ is independent of $q_{\boldsymbol{\mu}}$ and q_ν, we get

$$\begin{aligned}&\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})q_\nu(\nu_{1:K})}\left\{\log p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}})\right\}\\ &\overset{(\mathbf{z}_{1:N})}{=} \sum_{n=1}^{N}\left(\log p(\mathbf{z}_n; \theta^{\text{old}}) + \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})q_\nu(\nu_{1:K})}\{\log p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}})\}\right).\end{aligned}$$

And finally, take the exponential of the latest equation:

$$\begin{aligned}&\exp\left(\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})q_\nu(\nu_{1:K})}\left\{\log p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}})\right\}\right)\\ &\overset{(\mathbf{z}_{1:N})}{\propto} \prod_{n=1}^{N} p\left(\mathbf{z}_n; \theta^{\text{old}}\right)\exp\left(\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})q_\nu(\nu_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\}\right).\end{aligned}$$

(b) **Given the following definitions:**

$$\mathbf{m}_k := \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\{\boldsymbol{\mu}_k\} \quad \boldsymbol{\Omega}_k := \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\boldsymbol{\mu}_k\boldsymbol{\mu}_k^\top\right\} - \mathbf{m}_k\mathbf{m}_k^\top \tag{10}$$

$$\eta_k := \mathbb{E}_{q_\nu(\nu_{1:K})}\{\log\nu_k\} \qquad \rho_k := \mathbb{E}_{q_\nu(\nu_{1:K})}\left\{\nu_k^{-1}\right\} \tag{11}$$

Prove that:

$$\lambda_{n,k} := q_z\left(\mathbf{z}_n = k\right) = \frac{\pi_k^{\text{old}}\exp\left(-\frac{1}{2}\left[d_x\eta_k + \rho_k\left(\|\mathbf{x}_n - \mathbf{m}_k\|^2 + \operatorname{Tr}\left(\boldsymbol{\Omega}_k\right)\right)\right]\right)}{\sum_{\ell=1}^{K}\pi_\ell^{\text{old}}\exp\left(-\frac{1}{2}\left[d_x\eta_\ell + \rho_\ell\left(\|\mathbf{x}_n - \mathbf{m}_\ell\|^2 + \operatorname{Tr}\left(\boldsymbol{\Omega}_\ell\right)\right)\right]\right)} \tag{12}$$

We will start by the fact that $q_{\mathbf{z}}(\mathbf{z}_n = k)$ is proportional up to additive $\mathbf{z}_{1:N}$ to

$$\underbrace{p\left(\mathbf{z}_n = k; \theta^{\text{old}}\right)}_{\pi_k^{\text{old}}}\exp\left(\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})q_\nu(\nu_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\}\right).$$

We already have π_k^{old} in the last expression, so let us focus on the other part. Firstly, we have

$$\begin{aligned}\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right) &= \log\mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_k, \nu_k\mathbf{I}_{dx})\\ &= \log\left(\frac{\exp\left(-\frac{1}{2}(\mathbf{x}_n - \boldsymbol{\mu}_k)^T\nu_k^{-1}(\mathbf{x}_n - \boldsymbol{\mu}_k)\right)}{(2\pi)^{dx/2}\det(\nu_k I_{d_x})^{1/2}}\right)\\ &= -\frac{d_x}{2}\log(2\pi) - \frac{d_x}{2}\log\nu_k - \frac{\nu_k^{-1}}{2}\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2.\end{aligned}$$

The first term in the previous line is the constant, so taking the expectation will not change it. After taking the exponential of that, we will get multiplicative constant, which will not affect the proportionality. In other words, we can skip this part from now on.

Now we have

$$\begin{aligned}
&\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})q_{\nu}(\nu_{1:K})}\left[-\frac{d_x}{2}\log\nu_k - \frac{\nu_k^{-1}}{2}\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2\right] \\
&= -\frac{d_x}{2}\mathbb{E}_{q_{\nu}(\nu_{1:K})}[\log\nu_k] - \frac{1}{2}\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})q_{\nu}(\nu_{1:K})}\left[\nu_k^{-1}\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2\right] \\
&= -\frac{d_x}{2}\eta_k - \frac{1}{2}\mathbb{E}_{q_{\nu}(\nu_{1:K})}\left[\nu_k^{-1}\right]\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left[\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2\right] \\
&= -\frac{d_x\eta_k}{2} - \frac{\rho_k}{2}\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left[\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2\right].
\end{aligned}$$

Finally, we need to deal with the last term. We will use the properties of the Euclidean norm and scalar product. So we have

$$\begin{aligned}
\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left[\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2\right] &= \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left[\|\mathbf{x}_n\|^2\right] - 2\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left[\mathbf{x}_n^T\boldsymbol{\mu}_k\right] + \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left[\|\boldsymbol{\mu}_k\|^2\right] \\
&= \|\mathbf{x}_n\|^2 - 2\mathbf{x}_n^T\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left[\boldsymbol{\mu}_k\right] + \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left[\mathrm{Tr}(\boldsymbol{\mu}_k\boldsymbol{\mu}_k^T)\right] \\
&= \|\mathbf{x}_n\|^2 - 2\mathbf{x}_n^T\mathbf{m}_k + \|\mathbf{m_k}\|^2 - \|\mathbf{m_k}\|^2 + \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left[\mathrm{Tr}(\boldsymbol{\mu}_k\boldsymbol{\mu}_k^T)\right] \\
&= \|\mathbf{x}_n - \mathbf{m}_k\|^2 + \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left[\mathrm{Tr}(\boldsymbol{\mu}_k\boldsymbol{\mu}_k^T)\right] - \mathrm{Tr}(\mathbf{m}_k\mathbf{m}_k^T) \\
&= \|\mathbf{x}_n - \mathbf{m}_k\|^2 + \mathrm{Tr}\left(\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left[\boldsymbol{\mu}_k\boldsymbol{\mu}_k^T\right] - \mathbf{m}_k\mathbf{m}_k^T\right) \\
&= \|\mathbf{x}_n - \mathbf{m}_k\|^2 + \mathrm{Tr}(\boldsymbol{\Omega}_k).
\end{aligned}$$

We used that for a vector $\mathbf{a}$ we have

$$\|\mathbf{a}\|^2 = \sum_{i=1}^{d_a} a_i^2 = \mathrm{Tr}(\mathbf{a}\mathbf{a}^T),$$

because the matrix $\mathbf{a}\mathbf{a}^T$ has a_i^2 on diagonal. We also used the property of switching a trace and integral, which is the consequence of representing a trace as a sum, and the ability to switch it with an integral. At the end, we got

$$q_{\mathbf{z}}(\mathbf{z}_n = k) \overset{(\mathbf{z}_{1:N})}{\propto} \pi_k^{\text{old}}\,\exp\left(-\frac{1}{2}\left[d_x\eta_k + \rho_k\left(\|\mathbf{x}_n - \mathbf{m}_k\|^2 + \mathrm{Tr}\left(\boldsymbol{\Omega}_k\right)\right)\right]\right).$$

As $q_{\mathbf{z}}(\mathbf{z}_n = k)$ must sum up to one to be a probability mass function, so we get that

$$q_z\left(\mathbf{z}_n = k\right) = \frac{\pi_k^{\text{old}}\,\exp\left(-\frac{1}{2}\left[d_x\eta_k + \rho_k\left(\|\mathbf{x}_n - \mathbf{m}_k\|^2 + \mathrm{Tr}\left(\boldsymbol{\Omega}_k\right)\right)\right]\right)}{\sum_{\ell=1}^{K}\pi_\ell^{\text{old}}\,\exp\left(-\frac{1}{2}\left[d_x\eta_\ell + \rho_\ell\left(\|\mathbf{x}_n - \mathbf{m}_\ell\|^2 + \mathrm{Tr}\left(\boldsymbol{\Omega}_\ell\right)\right)\right]\right)}.$$

E-μ step: We will now find q_μ

(a) **Prove that the expression of q_μ simplifies down to**:

$$q_{\boldsymbol{\mu}}\left(\boldsymbol{\mu}_{1:K}\right) \overset{(\boldsymbol{\mu}_{1:K})}{\propto} \exp\left(\sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n) q_\nu(\nu_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\}\right) p\left(\boldsymbol{\mu}_{1:K}; \theta^{\text{old}}\right) \tag{13}$$

Like in the previous question, we will start with the generic expression

$$q_{\boldsymbol{\mu}}\left(\boldsymbol{\mu}_{1:K}\right) \overset{(\boldsymbol{\mu}_{1:K})}{\propto} \exp\left(\mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N}) q_\nu(\nu_{1:K})}\left\{\log p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}})\right\}\right).$$

Like the last time, we know that the probability inside the expectation is

$$p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}}) \overset{(\boldsymbol{\mu}_{1:K})}{\propto} p(\boldsymbol{\mu}_{1:K}; \theta^{\text{old}}) \prod_{n=1}^{N} p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}).$$

By taking the logarithm of the above expression, we have

$$\begin{aligned}\log\{p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}})\} &\overset{(\boldsymbol{\mu}_{1:K})}{=} \log\left\{p(\boldsymbol{\mu}_{1:K}; \theta^{\text{old}}) \prod_{n=1}^{N} p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}})\right\} \\ &\overset{(\boldsymbol{\mu}_{1:K})}{=} \log p(\boldsymbol{\mu}_{1:K}; \theta^{\text{old}}) + \sum_{n=1}^{N} \log p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}).\end{aligned}$$

Now, by taking the expected value of the above expression and using the fact that switching finite sum and the expected value is possible, and the fact that $\boldsymbol{\mu}_{1:k}$ is independent of $q_{\mathbf{z}}$ and q_ν, we get

$$\begin{aligned}&\mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N}) q_\nu(\nu_{1:K})}\left\{\log p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}})\right\} \\ &\overset{(\boldsymbol{\mu}_{1:K})}{=} \log p(\boldsymbol{\mu}_{1:K}; \theta^{\text{old}}) + \sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n) q_\nu(\nu_{1:K})}\{\log p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}})\}.\end{aligned}$$

And finally, by applying the exponential function on the latest equation, we get

$$\begin{aligned}&\exp\left(\mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N}) q_\nu(\nu_{1:K})}\left\{\log p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}})\right\}\right) \\ &\overset{(\boldsymbol{\mu}_{1:K})}{\propto} \exp\left(\sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n) q_\nu(\nu_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\}\right) p\left(\boldsymbol{\mu}_{1:K}; \theta^{\text{old}}\right)\end{aligned}$$

(b) **Use the fact that $q_z\left(\mathbf{z}_n = k\right) = \lambda_{n,k}$ to prove that $q_\mu\left(\boldsymbol{\mu}_{1:K}\right) = \prod_{k=1}^{K} q_\mu\left(\boldsymbol{\mu}_k\right)$ with:**

$$q_{\boldsymbol{\mu}}\left(\boldsymbol{\mu}_k\right) \overset{(\boldsymbol{\mu}_k)}{\propto} \exp\left(\sum_{n=1}^{N} \lambda_{n,k} \mathbb{E}_{q_\nu(\nu_k)}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\}\right) p\left(\boldsymbol{\mu}_k; \theta^{\text{old}}\right). \tag{14}$$

Starting with the previous we have

$$q_{\boldsymbol{\mu}}\left(\boldsymbol{\mu}_{1:K}\right) \overset{(\boldsymbol{\mu}_{1:K})}{\propto} \exp\left(\sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n) q_{\nu}(\nu_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\}\right) \prod_{k=1}^{K} p\left(\boldsymbol{\mu}_k; \theta^{\text{old}}\right).$$

We will start with the expectation inside the exponential

$$\begin{aligned}
&\mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n) q_{\nu}(\nu_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\} \\
&= \mathbb{E}_{q_{\nu}(\nu_{1:K})}\left\{\mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n)} \log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\} \\
&= \mathbb{E}_{q_{\nu}(\nu_{1:K})}\left\{\sum_{k=1}^{K} q_{\mathbf{z}}(\mathbf{z}_n = k) \log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\} \\
&= \sum_{k=1}^{K} \lambda_{n,k} \mathbb{E}_{q_{\nu}(\nu_k)}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\}.
\end{aligned}$$

Coming back to the first equation we have

$$\begin{aligned}
q_{\boldsymbol{\mu}}\left(\boldsymbol{\mu}_{1:K}\right) &\overset{(\boldsymbol{\mu}_{1:K})}{\propto} \exp\left(\sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n) q_{\nu}(\nu_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\}\right) \prod_{k=1}^{K} p\left(\boldsymbol{\mu}_k; \theta^{\text{old}}\right) \\
&\overset{(\boldsymbol{\mu}_{1:K})}{\propto} \exp\left(\sum_{n=1}^{N} \sum_{k=1}^{K} \lambda_{n,k} \mathbb{E}_{q_{\nu}(\nu_k)}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\}\right) \prod_{k=1}^{K} p\left(\boldsymbol{\mu}_k; \theta^{\text{old}}\right) \\
&\overset{(\boldsymbol{\mu}_{1:K})}{\propto} \prod_{k=1}^{K} \exp\left(\sum_{n=1}^{N} \lambda_{n,k} \mathbb{E}_{q_{\nu}(\nu_k)}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\}\right) \prod_{k=1}^{K} p\left(\boldsymbol{\mu}_k; \theta^{\text{old}}\right) \\
&\overset{(\boldsymbol{\mu}_{1:K})}{\propto} \prod_{k=1}^{K} \left(\exp\left(\sum_{n=1}^{N} \lambda_{n,k} \mathbb{E}_{q_{\nu}(\nu_k)}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\}\right) p\left(\boldsymbol{\mu}_k; \theta^{\text{old}}\right)\right).
\end{aligned}$$

This means that

$$q_{\boldsymbol{\mu}}\left(\boldsymbol{\mu}_k\right) \overset{(\boldsymbol{\mu}_k)}{\propto} \exp\left(\sum_{n=1}^{N} \lambda_{n,k} \mathbb{E}_{q_{\nu}(\nu_k)}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\}\right) p\left(\boldsymbol{\mu}_k; \theta^{\text{old}}\right)$$

(c) **Use the definition of the probabilistic model and of ρ_k, to prove that:**

$$q_{\boldsymbol{\mu}}\left(\boldsymbol{\mu}_k\right) = \mathcal{N}\left(\boldsymbol{\mu}_k; \mathbf{m}_k, \boldsymbol{\Omega}_k\right) \tag{15}$$

with the following definitions:

$$\boldsymbol{\Omega}_k^{-1} = \left(\boldsymbol{\Omega}^{\text{old}}\right)^{-1} + \mathbf{I}_{d_x} \rho_k \sum_{n=1}^{N} \lambda_{n,k} \quad \mathbf{m}_k = \boldsymbol{\Omega}_k \left(\left(\boldsymbol{\Omega}^{\text{old}}\right)^{-1} \mathbf{m}^{\text{old}} + \rho_k \sum_{n=1}^{N} \lambda_{n,k} \mathbf{x}_n\right) \tag{16}$$

When we want to prove that something has Normal distribution, we want to find $\log p$ and to prove that it is equal (up to constant) to quadratic and linear term, so let us start from there

$$\log q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k) \overset{(\boldsymbol{\mu}_k)}{=} \sum_{n=1}^{N} \lambda_{n,k}\mathbb{E}_{q_\nu(\nu_k)}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\} + \log p\left(\boldsymbol{\mu}_k; \theta^{\text{old}}\right).$$

We know that distribution under logarithm under expectation is Gaussian, also the distribution of μ_k is Gaussian. So, we can rewrite the previous and obtain

$$\begin{aligned}
\log q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k) &\overset{(\boldsymbol{\mu}_k)}{=} \sum_{n=1}^{N} \lambda_{n,k}\mathbb{E}_{q_\nu(\nu_k)}\left\{-\frac{1}{2}(\mathbf{x}_n - \boldsymbol{\mu}_k)^T \nu_k^{-1}(\mathbf{x}_n - \boldsymbol{\mu}_k)\right\} \\
&\quad - \frac{1}{2}\left(\boldsymbol{\mu}_k - \mathbf{m}^{\text{old}}\right)^T \boldsymbol{\Omega}^{\text{old}^{-1}}\left(\boldsymbol{\mu}_k - \mathbf{m}^{\text{old}}\right) \\
&= -\frac{1}{2}\rho_k \sum_{n=1}^{N} \lambda_{n,k}\left(\mathbf{x}_n^T\mathbf{x}_n - 2\boldsymbol{\mu}_k^T\mathbf{x}_n + \boldsymbol{\mu}_k^T\boldsymbol{\mu}_k\right) \\
&\quad - \frac{1}{2}\left(\boldsymbol{\mu}_k^T\boldsymbol{\Omega}^{\text{old}^{-1}}\boldsymbol{\mu_k} - 2\boldsymbol{\mu}_k^T\boldsymbol{\Omega}^{\text{old}^{-1}}\mathbf{m}^{\text{old}} + \mathbf{m}^{\text{old}^T}\boldsymbol{\Omega}^{\text{old}^{-1}}\mathbf{m}^{\text{old}}\right) \\
&\overset{(\boldsymbol{\mu}_k)}{=} -\frac{1}{2}\left[\boldsymbol{\mu}_k^T \underbrace{\left(\mathbf{I}_{d_x}\rho_k \sum_{n=1}^{N}\lambda_{n,k} + \boldsymbol{\Omega}^{\text{old}^{-1}}\right)}_{\boldsymbol{\Omega}_k^{-1}} \boldsymbol{\mu}_k - 2\boldsymbol{\mu}_k^T \underbrace{\left(\rho_k \sum_{n=1}^{N}\lambda_{n,k}\mathbf{x}_n + \boldsymbol{\Omega}^{\text{old}^{-1}}\mathbf{m}^{\text{old}}\right)}_{\boldsymbol{\Omega}_k^{-1}\mathbf{m}_k}\right].
\end{aligned}$$

(d) **What happens to the posterior of $\boldsymbol{\mu}_k$ if no observation is assigned to the k-th cluster?**

If no observation is assigned to the k-th cluster, then $\pi_k = 0$, and every new π_k will remain zero. This means that $\lambda_{n,k} = 0$, for all $n \in \{1, 2, ..., N\}$, and thus parameters of the distribution of $\boldsymbol{\mu}_k$ will remain $\mathbf{m}^{\text{old}}$ and $\boldsymbol{\Omega}^{\text{old}}$.

The **E-ν step** follows the same reasoning, and the derivations will be asked in the report (see the Mandatory additional questions), but for now, you can assume that the expression of q_ν simplifies down to:

$$q_\nu(\nu_{1:K}) = \prod_{k=1}^{K} q_\nu(\nu_k), \qquad q_\nu(\nu_k) = \mathcal{IG}(\nu_k; \alpha_k, \beta_k), \tag{17}$$

with the following definitions:

$$\alpha_k = \alpha^{\text{old}} + \frac{d_x}{2}\sum_{n=1}^{N}\lambda_{n,k}, \qquad \beta_k = \beta^{\text{old}} + \frac{1}{2}\sum_{n=1}^{N}\lambda_{n,k}\left(\|\mathbf{x}_n - \mathbf{m}_k\|^2 + \text{Tr}(\boldsymbol{\Omega}_k)\right). \tag{18}$$

Because $q_\nu(\nu_k)$ follows an inverse-gamma distribution, we can now compute η_k and ρ_k, defined in (11), with the following formulae:

$$\eta_k = \mathbb{E}_{q_\nu(\nu_k)}\{\log \nu_k\} = \log \beta_k - \psi(\alpha_k), \qquad \rho_k = \mathbb{E}_{q_\nu(\nu_k)}\{\nu_k^{-1}\} = \frac{\alpha_k}{\beta_k}, \tag{19}$$

where ψ stands for the digamma function (python will compute it for you).

4. **Variational M step:**

(a) **Prove that the expected complete-data log-likelihood simplifies down to:**

$$Q\left(\theta, \theta^{\text{old}}\right) \overset{(\theta)}{=} \underbrace{\sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n)} \left\{\log p(\mathbf{z}_n; \theta)\right\}}_{Q_{\mathbf{z}}} + \underbrace{\sum_{k=1}^{K} \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)} \left\{\log p(\boldsymbol{\mu}_k; \theta)\right\}}_{Q_{\boldsymbol{\mu}}} + \underbrace{\sum_{k=1}^{K} \mathbb{E}_{q_{\nu}(\nu_k)} \left\{\log p(\nu_k; \theta)\right\}}_{Q_{\nu}} . \tag{20}$$

Why did $p(\mathbf{x}_{1:N} \mid \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta)$ disappear?

Let us start with the $Q\left(\theta, \theta^{\text{old}}\right)$, we know that

$$Q\left(\theta, \theta^{\text{old}}\right) = \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N}) q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K}) q_{\nu}(\nu_{1:K})} \left\{\log p\left(\mathbf{x}_{1:N}, \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \theta\right)\right\} .$$

We can decompose the probability inside the expectation and obtain

$$p\left(\mathbf{x}_{1:N}, \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \theta\right) = p\left(\mathbf{x}_{1:N} \mid \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K}, \theta\right) \cdot p\left(\mathbf{z}_{1:N} \mid \theta\right) \cdot p\left(\boldsymbol{\mu}_{1:K} \mid \theta\right) \cdot p\left(\nu_{1:K} \mid \theta\right) .$$

Now, if we apply logarithm on the previous we have

$$\begin{aligned} &\log p\left(\mathbf{x}_{1:N}, \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \theta\right) \\ &= \log p\left(\mathbf{x}_{1:N} \mid \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K}, \theta\right) + \log p\left(\mathbf{z}_{1:N} \mid \theta\right) + \log p\left(\boldsymbol{\mu}_{1:K} \mid \theta\right) + \log p\left(\nu_{1:K} \mid \theta\right) . \end{aligned}$$

Next, we will compute the expectation of the previous result with respect to the distributions $q_{\mathbf{z}}(\mathbf{z}_{1:N})$, $q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})$ and $q_{\boldsymbol{\nu}}(\boldsymbol{\nu}_{1:K})$ and we obtain

$$\begin{aligned} &\mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N}) q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K}) q_{\nu}(\nu_{1:K})} \left\{\log p\left(\mathbf{x}_{1:N}, \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \theta\right)\right\} \\ &\overset{(\theta)}{=} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N})} \left\{\log p\left(\mathbf{z}_{1:N} \mid \theta\right)\right\} + \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})} \left\{\log p\left(\boldsymbol{\mu}_{1:K} \mid \theta\right)\right\} + \mathbb{E}_{q_{\nu}(\nu_{1:K})} \left\{\log p\left(\nu_{1:K} \mid \theta\right)\right\} \\ &= \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N})} \left\{\log \prod_{n=1}^{N} p\left(\mathbf{z}_n \mid \theta\right)\right\} + \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})} \left\{\log \prod_{k=1}^{K} p\left(\boldsymbol{\mu}_n \mid \theta\right)\right\} + \mathbb{E}_{q_{\nu}(\nu_{1:K})} \left\{\log \prod_{k=1}^{k} p\left(\nu_k \mid \theta\right)\right\} \\ &= \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N})} \left\{\sum_{n=1}^{N} \log p\left(\mathbf{z}_n \mid \theta\right)\right\} + \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})} \left\{\sum_{k=1}^{K} \log p\left(\boldsymbol{\mu}_n \mid \theta\right)\right\} + \mathbb{E}_{q_{\nu}(\nu_{1:K})} \left\{\sum_{k=1}^{k} \log p\left(\nu_k \mid \theta\right)\right\} \\ &= \underbrace{\sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n)} \left\{\log p\left(\mathbf{z}_n \mid \theta\right)\right\}}_{Q_{\mathbf{z}}} + \underbrace{\sum_{k=1}^{K} \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)} \left\{\log p\left(\boldsymbol{\mu}_n \mid \theta\right)\right\}}_{Q_{\mu}} + \underbrace{\sum_{k=1}^{k} \mathbb{E}_{q_{\nu}(\nu_k)} \left\{\log p\left(\nu_k \mid \theta\right)\right\}}_{Q_{\nu}} . \end{aligned}$$

Let us now discuss where $Q_{\mathbf{x}} = \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N}) q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K}) q_{\nu}(\nu_{1:K})} \left[\log p\left(\mathbf{x}_{1:N} \mid \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K}, \theta\right)\right]$ disappear. Intuitively, we know that parameters will be inside $p\left(\mathbf{x}_{1:N} \mid \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K}, \theta\right)$, which is product of normal distributions with parameters μ_k and ν_k. By taking expectations with respect to $q_{\boldsymbol{\mu}}$ and q_{ν} we will compute $p\left(\mathbf{x}_{1:N} \mid \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K}, \theta\right)$ over all values of $\boldsymbol{\mu}_k$ and ν_k, which means we will loose the parameters. Let us show this more formally

$$\begin{aligned}
Q_{\mathbf{x}} &= \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N})q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})q_{\nu}(\nu_{1:K})}\left[\log p\left(\mathbf{x}_{1:N} \mid \mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K}, \theta\right)\right] \\
&= \sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n)q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})q_{\nu}(\nu_{1:K})}\left[\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta\right)\right] \\
&= \sum_{n=1}^{N}\sum_{k=1}^{K} \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})q_{\nu}(\nu_{1:K})}\left[q_{\mathbf{z}}(\mathbf{z}_n = k)\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta\right)\right] \\
&= \sum_{n=1}^{N}\sum_{k=1}^{K} \lambda_{n,k} \int_{\boldsymbol{\mu}_k}\int_{\nu_k} q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)q_{\nu}(\nu_k)\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta\right) d\boldsymbol{\mu}_k\, d\nu_k \\
&= \sum_{n=1}^{N}\sum_{k=1}^{K} \lambda_{n,k} \int_{\boldsymbol{\mu}_k}\int_{\nu_k} q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)q_{\nu}(\nu_k)\ \left(-\frac{d_x}{2}\log\nu_k - \frac{1}{2\nu_k}\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2\right) d\boldsymbol{\mu}_k\, d\nu_k.
\end{aligned}$$

We can see that the previous integrals are constant in terms of $\boldsymbol{\mu}_k$ and ν_k where all the parameters are. This means, that $Q_{\mathbf{x}}$ does not depend on parameters θ.

(b) **Prove that $Q_{\mathbf{z}}$ boils down to:**

$$Q_{\mathbf{z}} = \sum_{n=1}^{N}\sum_{k=1}^{K} \lambda_{n,k}\log\pi_k, \tag{21}$$

and that the optimal solution for π_n are:

$$\pi_k^* = \frac{1}{N}\sum_{n=1}^{N}\lambda_{n,k}. \tag{22}$$

We start from

$$\begin{aligned}
Q_{\mathbf{z}} &= \sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n)}\left\{\log p(\mathbf{z}_n; \theta)\right\} \\
&= \sum_{n=1}^{N}\sum_{k=1}^{K} q_{\mathbf{z}}(\mathbf{z}_n = k)\left\{\log p(\mathbf{z}_n = k; \theta)\right\} \\
&= \sum_{n=1}^{N}\sum_{k=1}^{K} \lambda_{n,k}\log\pi_k.
\end{aligned}$$

Now, we want to find π_k which maximize $Q_{\mathbf{z}}$. We will need to use Lagrange multiplier, because of the constraint that $\sum_{k=1}^{K}\pi_k = 1$. So, we have

$$Q'_{\mathbf{z}} = \sum_{n=1}^{N}\sum_{k=1}^{K}\lambda_{n,k}\pi_k + \lambda\left(1 - \sum_{k=1}^{K}\pi_k\right).$$

Then, we have

$$\frac{\partial Q'_{\mathbf{z}}}{\partial \pi_k} = \sum_{n=1}^{N} \frac{\lambda_{n,k}}{\pi_k} - \lambda = 0.$$

This means that $\pi_k^* = \frac{1}{\lambda} \sum\limits_{n=1}^{N} \lambda_{n,k}$. Summing the previous from $k = 1$ until K we have

$$1 = \sum_{k=1}^{K} \pi_k^* = \frac{1}{\lambda} \sum_{n=1}^{N} \underbrace{\sum_{k=1}^{K} q_{\mathbf{z}}(\mathbf{z}_n = k)}_{1} = \frac{N}{\lambda},$$

which means that $\lambda = N$, or $\pi_k^* = \frac{1}{N} \sum\limits_{n=1}^{N} \lambda_{n,k}$.

(c) **Prove that $Q_{\boldsymbol{\mu}}$ boils down to:**

$$Q_{\boldsymbol{\mu}} \overset{(\theta)}{=} -\frac{1}{2} \left(K \log |\boldsymbol{\Omega}| + \sum_{k=1}^{K} (\mathbf{m}_k - \mathbf{m})^T \Omega^{-1} (\mathbf{m}_k - \mathbf{m}) + \mathrm{Tr}(\boldsymbol{\Omega}^{-1} \boldsymbol{\Omega}_k) \right), \tag{23}$$

and that the optimal value for $\mathbf{m}$ and $\boldsymbol{\Omega}$ are:

$$\mathbf{m}^* = \frac{1}{K} \sum_{k=1}^{K} \mathbf{m}_k, \qquad \boldsymbol{\Omega}^* = \frac{1}{K} \sum_{k=1}^{K} (\mathbf{m}_k - \mathbf{m}^*)(\mathbf{m}_k - \mathbf{m}^*)^T + \boldsymbol{\Omega}_k. \tag{24}$$

Again, we will start from the given

$$\begin{aligned}
Q_{\boldsymbol{\mu}} &= \sum_{k=1}^{K} \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)} \left\{ \log p(\boldsymbol{\mu}_k; \theta) \right\} \overset{(\theta)}{=} -\frac{1}{2} \sum_{k=1}^{K} \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)} \left[\log |\boldsymbol{\Omega}| + (\boldsymbol{\mu}_k - \mathbf{m})^T \boldsymbol{\Omega}^{-1} (\boldsymbol{\mu}_k - \mathbf{m}) \right] \\
&= -\frac{1}{2} \sum_{k=1}^{K} \left(\log |\boldsymbol{\Omega}| + \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)} \left[\boldsymbol{\mu}_k^T \boldsymbol{\Omega}^{-1} \boldsymbol{\mu}_k - 2 \boldsymbol{\mu}_k^T \boldsymbol{\Omega}^{-1} \mathbf{m} + \mathbf{m}^T \boldsymbol{\Omega}^{-1} \mathbf{m} \right] \right) \\
&= -\frac{1}{2} \sum_{k=1}^{K} \left(\log |\boldsymbol{\Omega}| + \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)} \underbrace{\left[\boldsymbol{\mu}_k^T \boldsymbol{\Omega}^{-1} \boldsymbol{\mu}_k \right]}_{\in \mathbb{R}} - 2 \mathbf{m}_k^T \boldsymbol{\Omega}^{-1} \mathbf{m} + \mathbf{m}^T \boldsymbol{\Omega}^{-1} \mathbf{m} \right) \\
&= -\frac{1}{2} \sum_{k=1}^{K} \left(\log |\boldsymbol{\Omega}| + \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)} \left[\mathrm{Tr}(\boldsymbol{\mu}_k^T \boldsymbol{\Omega}^{-1} \boldsymbol{\mu}_k) \right] - 2 \mathbf{m}_k^T \boldsymbol{\Omega}^{-1} \mathbf{m} + \mathbf{m}^T \boldsymbol{\Omega}^{-1} \mathbf{m} \right) \\
&= -\frac{1}{2} \sum_{k=1}^{K} \left(\log |\boldsymbol{\Omega}| + \mathrm{Tr} \left(\boldsymbol{\Omega}^{-1} \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)} \left[\boldsymbol{\mu}_k \boldsymbol{\mu}_k^T \right] \right) - 2 \mathbf{m}_k^T \boldsymbol{\Omega}^{-1} \mathbf{m} + \mathbf{m}^T \boldsymbol{\Omega}^{-1} \mathbf{m} \right).
\end{aligned}$$

We used the property of trace (it can switch with an integral), and the matrices inside can rotate. Now, we will use the fact that $E[XX^T] - E[X]E[X]^T = \mathrm{cov}(X)$, so we have

$$\begin{aligned}
Q_{\boldsymbol{\mu}} &= -\frac{1}{2}\sum_{k=1}^{K}\Bigg(\log|\boldsymbol{\Omega}| + \mathrm{Tr}\left(\boldsymbol{\Omega}^{-1}\left(\boldsymbol{\Omega}_k + \mathbf{m}_k\mathbf{m}_k^T\right)\right) - 2\mathbf{m}_k^T\boldsymbol{\Omega}^{-1}\mathbf{m} + \mathbf{m}^T\boldsymbol{\Omega}^{-1}\mathbf{m}\Bigg) \\
&= -\frac{1}{2}\sum_{k=1}^{K}\Bigg(\log|\boldsymbol{\Omega}| + \mathrm{Tr}\left(\boldsymbol{\Omega}^{-1}\boldsymbol{\Omega}_k\right) + \mathrm{Tr}\left(\boldsymbol{\Omega}^{-1}\mathbf{m}_k\mathbf{m}_k^T\right) - 2\mathbf{m}_k^T\boldsymbol{\Omega}^{-1}\mathbf{m} + \mathbf{m}^T\boldsymbol{\Omega}^{-1}\mathbf{m}\Bigg) \\
&= -\frac{1}{2}\sum_{k=1}^{K}\Bigg(\log|\boldsymbol{\Omega}| + \mathrm{Tr}\,\underbrace{\left(\mathbf{m}_k^T\boldsymbol{\Omega}^{-1}\mathbf{m}_k\right)}_{\in\mathbb{R}} -2\mathbf{m}_k^T\boldsymbol{\Omega}^{-1}\mathbf{m} + \mathbf{m}^T\boldsymbol{\Omega}^{-1}\mathbf{m} + \mathrm{Tr}\left(\boldsymbol{\Omega}^{-1}\boldsymbol{\Omega}_k\right)\Bigg) \\
&= -\frac{1}{2}\sum_{k=1}^{K}\Bigg(\log|\boldsymbol{\Omega}| + \mathbf{m}_k^T\boldsymbol{\Omega}^{-1}\mathbf{m}_k - 2\mathbf{m}_k^T\boldsymbol{\Omega}^{-1}\mathbf{m} + \mathbf{m}^T\boldsymbol{\Omega}^{-1}\mathbf{m} + \mathrm{Tr}\left(\boldsymbol{\Omega}^{-1}\boldsymbol{\Omega}_k\right)\Bigg) \\
&= -\frac{1}{2}\Bigg(K\log|\boldsymbol{\Omega}| + \sum_{k=1}^{K}\left((\mathbf{m}_k-\mathbf{m})^T\boldsymbol{\Omega}^{-1}(\mathbf{m}_k-\mathbf{m}) + \mathrm{Tr}\left(\boldsymbol{\Omega}^{-1}\boldsymbol{\Omega}_k\right)\right)\Bigg) \\
&= -\frac{1}{2}\Bigg(- K\log\left|\boldsymbol{\Omega}^{-1}\right| + \sum_{k=1}^{K}\left((\mathbf{m}_k-\mathbf{m})^T\boldsymbol{\Omega}^{-1}(\mathbf{m}_k-\mathbf{m}) + \mathrm{Tr}\left(\boldsymbol{\Omega}^{-1}\boldsymbol{\Omega}_k\right)\right)\Bigg).
\end{aligned}$$

In order to compute the partial gradients of $Q_{\boldsymbol{\mu}}$ with respect to $\mathbf{m}$ and $\boldsymbol{\Omega}$, we will use that

$$\begin{aligned}
\frac{\partial}{\partial A}\,\mathrm{Tr}(ABA^T) &= A(B+B^T) \\
\frac{\partial}{\partial A}\,\mathrm{Tr}(A^TB) &= B \\
\frac{\partial}{\partial A}\log|A| &= \left(A^{-1}\right)^T,
\end{aligned}$$

and the chain rule. Having in mind that $\boldsymbol{\Omega}$ is symmetric, we have

$$\frac{\partial Q_{\boldsymbol{\mu}}}{\partial \mathbf{m}} = \sum_{k=1}^{K}(\mathbf{m}_k - \mathbf{m})\boldsymbol{\Omega}^{-1} = 0,$$

which means that $\mathbf{m}^* = \frac{1}{K}\sum\limits_{k=1}^{K}\mathbf{m}_k$. On the other hand, we have

$$\frac{\partial Q_{\boldsymbol{\mu}}}{\partial \boldsymbol{\Omega}^{-1}} = -\frac{1}{2}\left(-K\boldsymbol{\Omega} + \sum_{k=1}^{K}(\mathbf{m}_k-\mathbf{m})(\mathbf{m}_k-\mathbf{m})^T + \boldsymbol{\Omega}_k\right) = 0,$$

and we have that

$$\boldsymbol{\Omega}^* = \frac{1}{K}\sum_{k=1}^{K}\left((\mathbf{m}_k-\mathbf{m}^*)(\mathbf{m}_k-\mathbf{m}^*)^T + \boldsymbol{\Omega}_k\right).$$

(d) **The last step is to derive Q_ν. In this lab work, we will assume that $\alpha^{\textbf{old}}$ is a fixed parameter that does NOT need to be estimated: $\alpha^{\textbf{old}}$ is a constant. Under this hypothesis, derive the expression for Q_ν down to:**

$$Q_\nu \overset{(\beta)}{=} K\alpha^{\text{old}} \log \beta - \beta \sum_{k=1}^{K} \rho_k, \tag{25}$$

and provide that the optimal value for β writes:

$$\beta^* = \frac{K\alpha^{\text{old}}}{\sum_{k=1}^{K} \rho_k}. \tag{26}$$

As always, we start with

$$\begin{aligned}
Q_\nu &= \sum_{k=1}^{K} \mathbb{E}_{q_\nu(\nu_k)}\{\log p(\nu_k;\theta)\} = \sum_{k=1}^{K} \mathbb{E}_{q_\nu(\nu_k)}\left\{\log \mathcal{IG}(\nu_k;\alpha^{\text{old}},\beta)\right\} \\
&= \sum_{k=1}^{K} \mathbb{E}_{q_\nu(\nu_k)}\left\{\log \frac{\beta^{\alpha^{\text{old}}}}{\Gamma(\alpha^{\text{old}})} \nu_k^{-\alpha^{\text{old}}-1} \exp\left(-\frac{\beta}{\nu_k}\right)\right\} \\
&= \sum_{k=1}^{K} \mathbb{E}_{q_\nu(\nu_k)}\left\{\alpha^{\text{old}} \log\beta - \log\Gamma(\alpha^{\text{old}}) - (\alpha^{\text{old}}+1)\log\nu_k - \frac{\beta}{\nu_k}\right\} \\
&\overset{(\beta)}{=} \sum_{k=1}^{K} \left(\alpha^{\text{old}} \log\beta - \beta \mathbb{E}_{q_\nu(\nu_k)}\left\{\nu_k^{-1}\right\}\right) \\
&= K\alpha^{\text{old}} \log\beta - \beta \sum_{k=1}^{K} \rho_k.
\end{aligned}$$

Now, we want to find beta that maximize the Q_ν, so we need to find the derivative in respect to β and put it to be zero, i.e.

$$\frac{\partial Q_\nu}{\partial \beta} = \frac{K\alpha^{\text{old}}}{\beta} - \sum_{k=1}^{K} \rho_k = 0.$$

The solution of the previous equation is obviously $\beta^* = \frac{K\alpha^{\text{old}}}{\sum_{k=1}^{K} \rho_k}$.

3.3 Practical work

Download `TP3-VEM-main.py` and `digamma.py` from `chamilo`, save them in the same folder.

1. **Code analysis:** read the code provided and identify the various sections: data generation, initialisation, VEM and evaluation.

(a) **Does the data generation process follow the probabilistic generative model?**
We will provide the data generation in short, here. Firstly, we will work with 2-dimensional observations and the number of components is $K = 3$. The distribution of means will be

$$p(\boldsymbol{\mu}_k; \theta) = \mathcal{N}\left(\boldsymbol{\mu}_k; \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}\right).$$

The distribution of variances are

$$p(\nu_k; \theta) = \mathcal{G}(\nu_k; k = 40, \theta = 0.1).$$

Then the data is mixed randomly into three classes and for each class we used corresponding mean and variance generated from the previous distributions. So, the process fallows the probabilistic generative model. We can see the the plot of the data in the Figure 1.

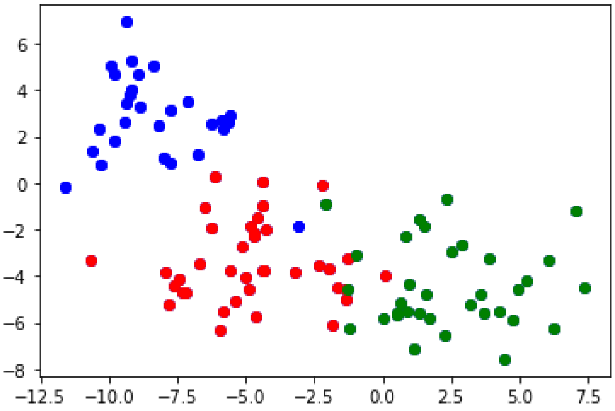


Figure 1: Data from the GMM with prior mean and covariance for $K = 3$.

(b) **What are the two proposed ways to initialise $\mathbf{m}_k$? How are the $\lambda_{n,k}$ initialised?**
The proposals to initialize the $\mathbf{m}_k$ are either randomly chosen observations or ground truth values (which we can choose because we generated the data, but in the real life this cannot be done). We initialised $\lambda_{n,k}$ to be all zeros.

(c) **Regarding the evaluation: which two metrics are used? What is done before computing these metrics? Why?**
We used Mean vector distances and the Accuracy. Before computing these two met rices, we are solving the linear sum assignment optimization problem. First of all, we need to compute a matrix `cluster_assignment_cost` $\in \mathbb{R}^{K \times K}$. In our case, we have three cluster $0, 1$ and 2. The element of matrix at position $(1, 2)$ represents sum of the rates from the points that are placed by the hard clustering assignment in cluster 1 and were labeled by the program to belong to cluster 2. We would like to have non zero values just in the diagonal of the matrix. In other cases, we want to find most efficient movement to the nearer clusters. We are doing this by linear assignment sum algorithm. The output of this optimization algorithm is a matrix $X \in \mathbb{R}^{K \times K}$, which has ones and zeros for elements, with constrain that in each row/column there is exactly one 1. The reason for doing this is next. We know that EM is maximizing the Q function, which is concave and it does not guarantee that we get the

optimal solution. This means that it might converge to a local optimum. On the other hand, with solving linear sum assignment optimization problem, we can find the optimal solution. Because EM may find a sub-optimal solution, we need the optimal solution to compare with, and this optimal one is provided by linear sum assignment optimization problem.

2. **VEM Implementation:**

(a) **What would happen to $\lambda_{n,k}$ if, for a given n:**

$$\pi_k^{\text{old}} \exp\left(-\frac{1}{2}\left[d_x\eta_k + \rho_k\left(\|\mathbf{x}_n - \mathbf{m}_k\|^2 + \text{Tr}(\boldsymbol{\Omega}_k)\right)\right]\right) \to 0 \quad \forall k? \tag{27}$$

How could we overcome that?

We know that

$$\lambda_{n,k} = \frac{\pi_k^{\text{old}} \exp\left(-\frac{1}{2}\left[d_x\eta_k + \rho_k\left(\|\mathbf{x}_n - \mathbf{m}_k\|^2 + \text{Tr}\left(\boldsymbol{\Omega}_k\right)\right)\right]\right)}{\sum\limits_{\ell=1}^{K} \pi_\ell^{\text{old}} \exp\left(-\frac{1}{2}\left[d_x\eta_\ell + \rho_\ell\left(\|\mathbf{x}_n - \mathbf{m}_\ell\|^2 + \text{Tr}\left(\boldsymbol{\Omega}_\ell\right)\right)\right]\right)}.$$

If for all k and fixed n we have (27), then we will have zero in both denominator and numerator and the $\lambda_{n,k}$ will have an undefined value due to the non-determination case $\frac{0}{0}$. In order to avoid this problem, we can add a small value (say $\varepsilon = 10^{-10}$) both to the quantities at the numerator and denominator and have limit case

$$\lambda_{n,k} = \frac{\pi_k^{\text{old}} \exp\left(-\frac{1}{2}\left[d_x\eta_k + \rho_k\left(\|\mathbf{x}_n - \mathbf{m}_k\|^2 + \text{Tr}\left(\boldsymbol{\Omega}_k\right)\right)\right]\right) + \varepsilon}{\sum\limits_{\ell=1}^{K} \left[\pi_\ell^{\text{old}} \exp\left(-\frac{1}{2}\left[d_x\eta_\ell + \rho_\ell\left(\|\mathbf{x}_n - \mathbf{m}_\ell\|^2 + \text{Tr}\left(\boldsymbol{\Omega}_\ell\right)\right)\right]\right) + \varepsilon\right]} \to \frac{\varepsilon}{K\varepsilon} = \frac{1}{K}.$$

In this way, the values $\lambda_{n,k}$ will be uniformly distributed, $\forall k \in \{1, ..., K\}$ and this is how we solved the issue.

(b) **What would happen to $\boldsymbol{\Omega}_k$ if $(\boldsymbol{\Omega}^{\textbf{old}})^{-1} + \mathbf{I}_{d_x}\rho_k \sum\limits_{n=1}^{N} \lambda_{n,k}$ is not full rank? How could we overcome that?**

We have seen that

$$\boldsymbol{\Omega}_k = \left[(\boldsymbol{\Omega}^{\text{old}})^{-1} + \left(\rho_k \sum_{n=1}^{N} \lambda_{n,k}\right)\mathbf{I}_{d_x}\right]^{-1}.$$

This means that we need to take the inverse of $(\boldsymbol{\Omega}^{\text{old}})^{-1} + \mathbf{I}_{d_x}\rho_k \sum\limits_{n=1}^{N} \lambda_{n,k}$. If this matrix is not full rank, than we cannot take the inverse. To overcome this we can add $\varepsilon\mathbf{I}_{d_x}$ to the matrix and we can achieve full rank by that. The problem is that we changed our original matrix, so in order to minimize the change, we should consider small ε, like 10^{-10} and the formula becomes:

$$\boldsymbol{\Omega}_k = \left[(\boldsymbol{\Omega}^{\text{old}})^{-1} + \varepsilon\mathbf{I}_{d_x} + \left(\rho_k \sum_{n=1}^{N} \lambda_{n,k}\right)\mathbf{I}_{d_x}\right]^{-1} = \left[(\boldsymbol{\Omega}^{\text{old}})^{-1} + \left(\varepsilon + \rho_k \sum_{n=1}^{N} \lambda_{n,k}\right)\mathbf{I}_{d_x}\right]^{-1}.$$

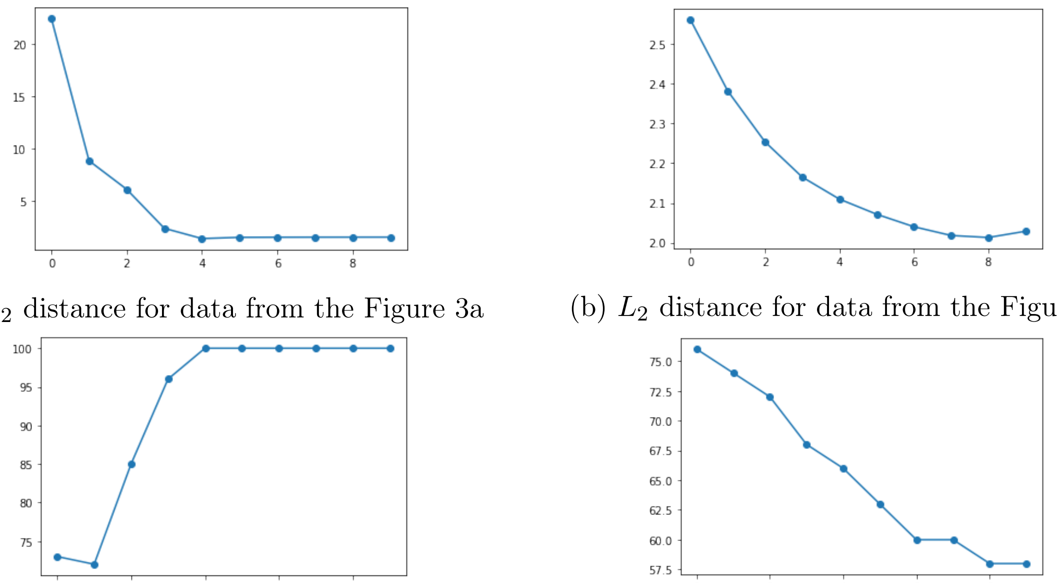
(c) **Implement all the steps of the VEM in the order given in the code template and taking into account the previous two tricks.**

Note: The digamma function, ψ, is computed in the initialisation. Re-use!

3. **VEM Evaluation**:

(a) **Plot the evolution of the two performance measures as a function of the iteration. Plot the input data, the initial, ground truth and final estimated assignment. Do this for an easy case (no overlap) and a difficult case (overlap). Discuss the differences. Save the two random datasets (easy and difficult) together with the associated ground truth data.**

In the first row of the Figure 3 we can see on the left data without overlapping and on the right data with overlapping. Below we can see the ground truth assignment, initial assignment, and the final assignment. The evaluation of the performance measures (L_2 distance and accuracy) for both cases are given in the Figure 2.



(a) L_2 distance for data from the Figure 3a

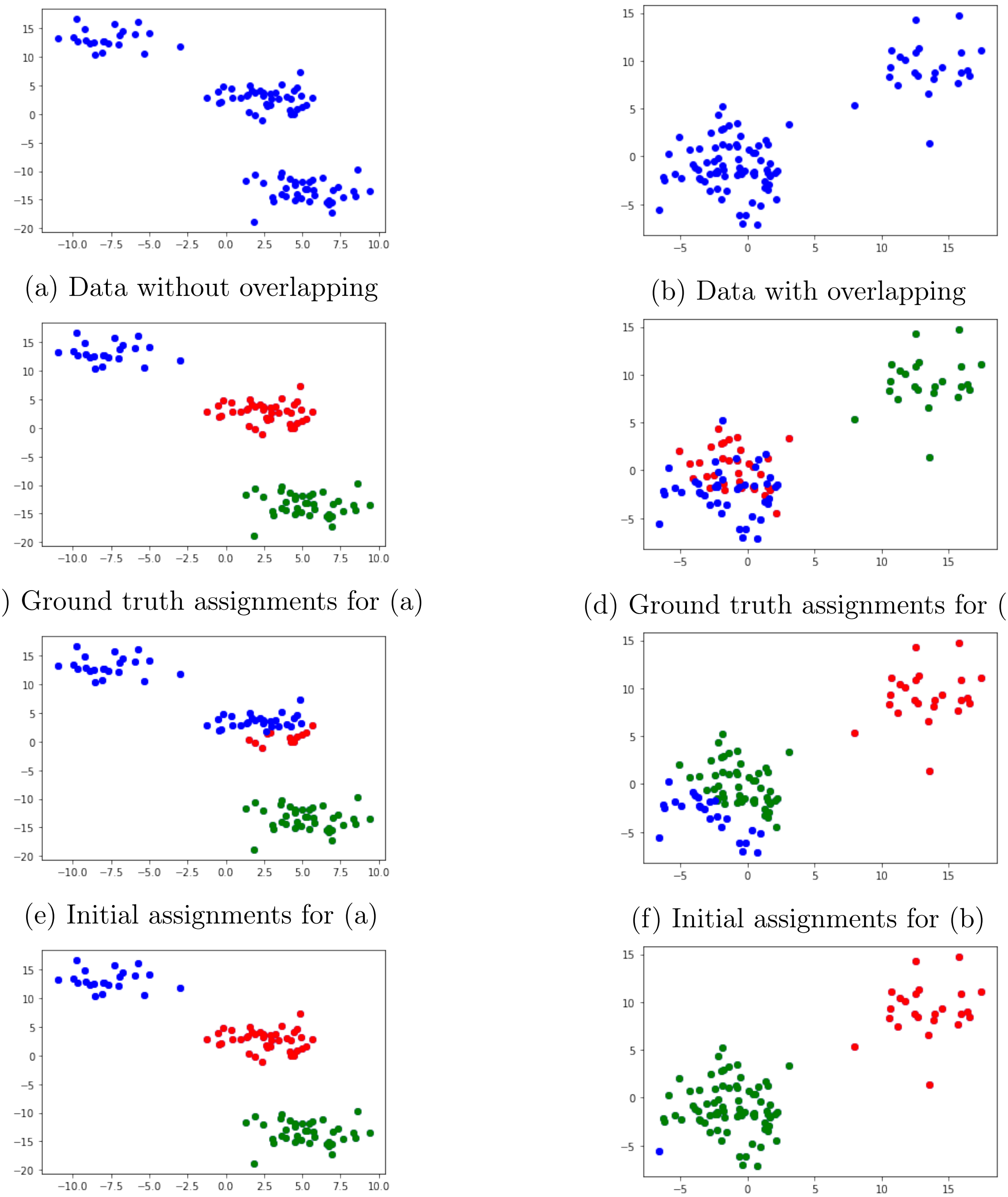
(b) L_2 distance for data from the Figure 3b

(c) Accuracy for data from the Figure 3a

(d) Accuracy for data from the Figure 3b

Figure 2: Performance measures through iterations for data with and without overlapping

From the Figure 3 we can compare the model for "easy" and "difficult" data. We can see that the model works really well for the "easy" data. The hard assignment is exactly the same as the ground truth. On the other hand, for the "difficult" data, the situation is different. We can see that ground truth is really complicated, and there is no way that unsupervised model can learn to get clusters correctly. This leads to not so perfect hard assignment, but for this data it is still good. If we go back to Figure 2 we can see that for the easy data, the L_2 distance between means is smaller with the number of iteration increasing. Similarly, the accuracy (the percent of correctly classified data) increases and in the end it reaches 100% after fifth iteration.



(a) Data without overlapping

(b) Data with overlapping

(c) Ground truth assignments for (a)

(d) Ground truth assignments for (b)

(e) Initial assignments for (a)

(f) Initial assignments for (b)

(g) Final assignments for (a)

(h) Final assignments for (b)

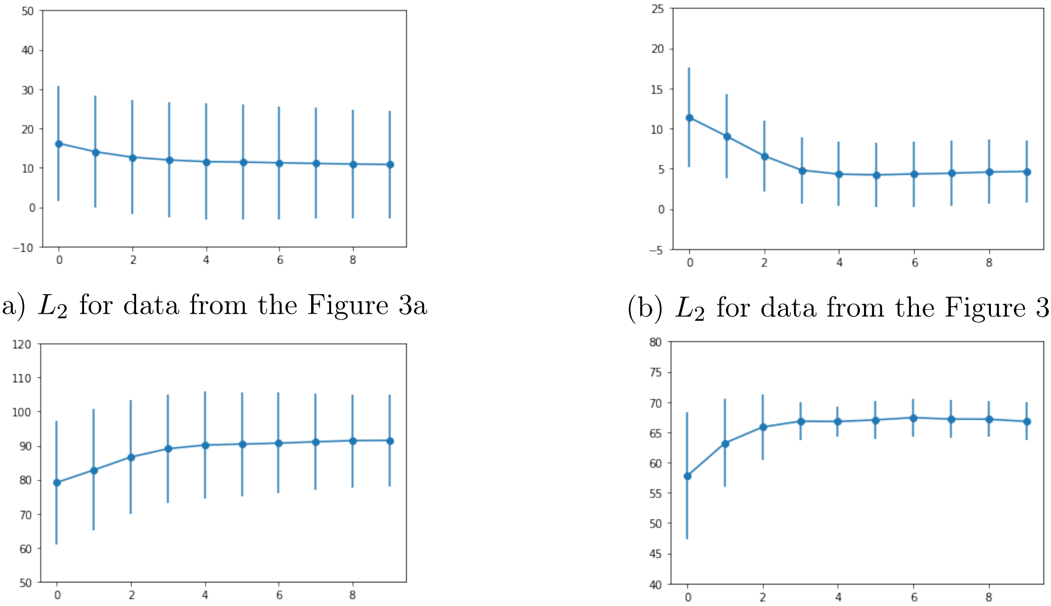
Figure 3: Testing the model on data with and without overlapping

When it comes to "difficult" data we can see completely different graph for the accuracy: the number of correctly classified data decreases with the number of iterations. This can also be noticed in the Figure 3 in parts (d), (f) and (h), where we can see that initial assignments are much closer to the ground truth than the final one. This is because we are using the measures for the hard assignment and although we have distribution for each cluster, we are choosing the one with highest probability. And it looks like that one is not the good cluster to be chosen.

(b) **For the two previous stored datasets, run $R = 50$ times the VEM with random initialisation. Report the mean and standard deviation of the evolution of the two performance measures as a function of the iteration.**

Note: You may use `matplotlib.pyplot.errorbar`.

For the "easy" and "difficult" data we saved ground truth assignments and ground truth means. Then, we run $R = 50$ times the VEM and for each iteration and each run we saved the obtained measures. After that, for fixed iteration and fixed measure we calculated the mean and standard deviation from the sample of 50 measures. And finally, we plot the results with `errorbar`, which can be seen in the Figure 4.



(a) L_2 for data from the Figure 3a

(b) L_2 for data from the Figure 3a

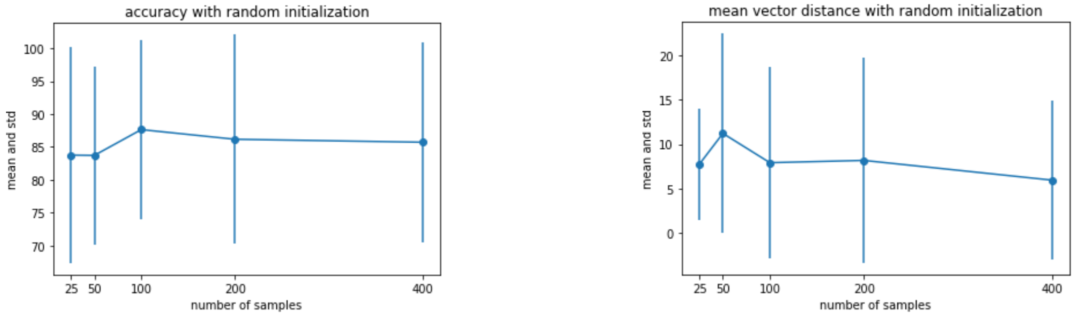
(c) ACC for data from the Figure 3b

(d) ACC for data from the Figure 3b

Figure 4: Means and standard deviation of measures through iterations computed on the sample of size $R = 50$ for data with and without overlapping

For the "easy" data we can see that the L_2 distance between means are almost constant through iterations. This is because of "easiness" of data, i.e. from the beginning the algorithm will be successful and it will not change a lot through time. Similar can be said for the accuracy. The algorithm needs few iteration to obtain the best possible accuracy and it will not change much later. Also, for both measures the standard deviation is not chaining through time. On the other hand, for the "difficult" data we can see that both measures needs more iteration to became constant and the standard deviation is increasing through time. This means that we are more certain that the measure is around the mean.

(c) **Run a sensitivity analysis on N (# of observations). To do so, take a list of at least five values centered at $N_g = 100$ (value given in the code) $[N_g/4, N_g/2, N_g, 2N_g, 4N_g]$. For each of these values, run the $R = 50$ initialisation, and report the mean and standard deviation in an error bar plot, as a function of the values of N.**

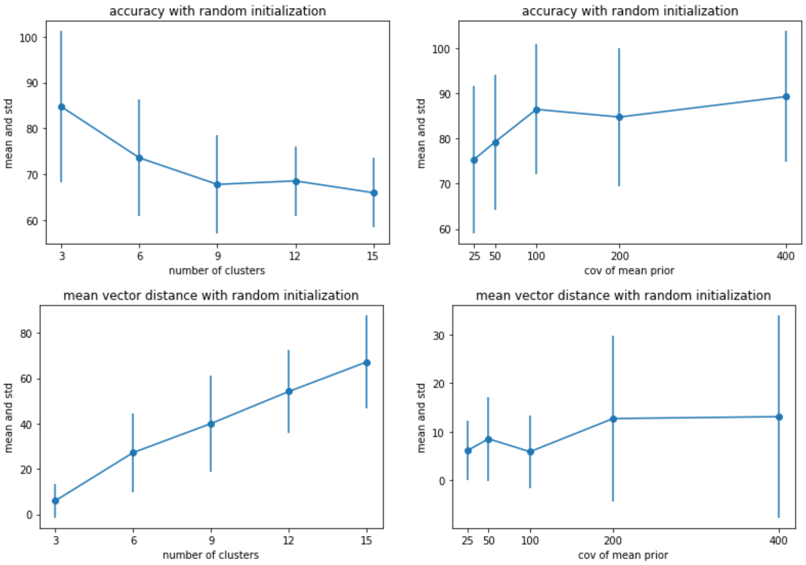


(a) Accuracy when N is variable, $K = 3$, $\omega = 100$ (b) MVD when N is variable, $K = 3$, $\omega = 100$

Figure 5: Sensitivity analysis plots for N for random initialization

In the Figure 5 we can see a small improvement in accuracy when we have at least 100 samples, but the accuracy does not reach 90%. Given the fact that initial means are chosen randomly, the variance is also high both for accuracy and mean vector distance and thus the mean of mean vector distance does not significantly change with number of samples. Random initialization introduces noise in metric estimations and this can be seen in the standard deviation as previously mentioned.

(d) **Repeat the operation with ω ($\Omega = \omega \mathbf{I}_{d_x}$ is the covariance of the mean prior) around $\omega_g = 100$: $[\omega_g/4, \omega_g/2, \omega_g, 2\omega_g, 4\omega_g]$, and with K, the number of clusters picked from the list $[K_g, 2K_g, 3K_g, 4K_g, 5K_g]$, being $K_g = 3$. Discuss these three plots.**



(a) K is variable (b) ω is variable

Figure 6: Sensitivity analysis plots for K and ω for random initialization

Below we will discuss each subplots (a) and (b) from the Figure 6, separately.

(a) Because the number of samples is fixed, the more clusters we want to find, the lower the accuracy will be because the number of samples associated to each mean/cluster becomes smaller. As a consequence, we will get a poor estimation for the mean which causes the mean distance vector metric to increase, creating a negative correlation with the accuracy because the accuracy is decreasing when the mean vector distance is increasing.

(b) An important observation is that ω is a measure of spread for data inside the clusters. A small value for ω results in a big spread of clusters. When ω is small, the clusters tend to have a big overlap. We can see that the variance for mean vector distance is increasing when ω increases. This means the clusters tend to have small spread and the random initialization might choose some initial means inside one cluster, causing the big variance as ω increases because one cluster can be very far from all other means. When ω is small, we have low variance because the clusters have a big spread and the random initialization has more chances to pick the means such that they have many neighbors around.

3.4 Mandatory additional questions

1. Derivation of the E-ν step. Let us compute q_ν:

(a) **Prove that the expression of q_ν simplifies down to:**

$$q_\nu(\nu_{1:K}) \overset{(\nu_{1:K})}{\propto} \exp\left(\sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n)q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\}\right) p\left(\nu_{1:K}; \theta^{\text{old}}\right). \quad (28)$$

As always, we start with the generic expression

$$q_\nu\left(\nu_{1:K}\right) \overset{(\nu_{1:K})}{\propto} \exp\left(\mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N})q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\log p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}})\right\}\right).$$

The probability inside the expectation writes as

$$p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}}) \overset{(\nu_{1:K})}{\propto} p(\nu_{1:K}; \theta^{\text{old}}) \prod_{n=1}^{N} p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}).$$

By taking the logarithm of the above expression, we have

$$\begin{aligned}\log\{p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}})\} &\overset{(\nu_{1:K})}{=} \log\left\{p(\nu_{1:K}; \theta^{\text{old}}) \prod_{n=1}^{N} p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}})\right\} \\ &\overset{(\nu_{1:K})}{=} \log p(\nu_{1:K}; \theta^{\text{old}}) + \sum_{n=1}^{N} \log p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}).\end{aligned}$$

Since $\nu_{1:k}$ is independent of $q_{\mathbf{z}}$ and $q_{\boldsymbol{\mu}}$, we get

$$\begin{aligned}&\mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N})q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\log p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}})\right\} \\ &\overset{(\nu_{1:K})}{=} \log p(\nu_{1:K}; \theta^{\text{old}}) + \sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n)q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\{\log p(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}})\}.\end{aligned}$$

And finally, applying the exponential function on the latest equation, we have

$$\begin{aligned}
&\exp\left(\mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_{1:N})q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\log p(\mathbf{z}_{1:N}, \boldsymbol{\mu}_{1:K}, \nu_{1:K} \mid \mathbf{x}_{1:N}; \theta^{\text{old}})\right\}\right) \\
&\overset{(\nu_{1:K})}{\propto} \exp\left(\sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n)q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\}\right) p\left(\nu_{1:K}; \theta^{\text{old}}\right)
\end{aligned}$$

(b) **Use the fact that $q_{\mathbf{z}}(\mathbf{z}_n = k) = \lambda_{n,k}$ to prove that $q_\nu(\nu_{1:K}) = \prod_{k=1}^{K} q_\nu(\nu_k)$ with:**

$$q_\nu(\nu_k) \overset{(\nu_k)}{\propto} \exp\left(\sum_{n=1}^{N} \lambda_{n,k}\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\}\right) p\left(\nu_k; \theta^{\text{old}}\right). \tag{29}$$

Starting with the previous we have

$$q_\nu(\nu_{1:K}) \overset{(\nu_{1:K})}{\propto} \exp\left(\sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n)q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\}\right) \prod_{k=1}^{K} p\left(\nu_k; \theta^{\text{old}}\right).$$

Let us consider the expectation term

$$\begin{aligned}
&\mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n)q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\} \\
&= \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n)} \log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\} \\
&= \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\sum_{k=1}^{K} q_{\mathbf{z}}(\mathbf{z}_n = k) \log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\} \\
&= \sum_{k=1}^{K} \lambda_{n,k}\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\}.
\end{aligned}$$

Coming back to the first equation we have

$$\begin{aligned}
q_\nu(\nu_{1:K}) &\overset{(\nu_{1:K})}{\propto} \exp\left(\sum_{n=1}^{N} \mathbb{E}_{q_{\mathbf{z}}(\mathbf{z}_n)q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\}\right) \prod_{k=1}^{K} p\left(\nu_k; \theta^{\text{old}}\right) \\
&\overset{(\nu_{1:K})}{\propto} \exp\left(\sum_{n=1}^{N}\sum_{k=1}^{K} \lambda_{n,k}\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\}\right) \prod_{k=1}^{K} p\left(\nu_k; \theta^{\text{old}}\right) \\
&\overset{(\nu_{1:K})}{\propto} \prod_{k=1}^{K} \exp\left(\sum_{n=1}^{N} \lambda_{n,k}\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\}\right) \prod_{k=1}^{K} p\left(\nu_k; \theta^{\text{old}}\right) \\
&\overset{(\nu_{1:K})}{\propto} \prod_{k=1}^{K} \left(\exp\left(\sum_{n=1}^{N} \lambda_{n,k}\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}}\right)\right\}\right) p\left(\nu_k; \theta^{\text{old}}\right)\right).
\end{aligned}$$

This means that

$$q_\nu(\nu_k) \overset{(\nu_k)}{\propto} \exp\left(\sum_{n=1}^{N} \lambda_{n,k}\mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_{1:K})}\left\{\log p\left(\mathbf{x}_n \mid \mathbf{z}_n, \boldsymbol{\mu}_{1:K}, \nu_{1:K}; \theta^{\text{old}}\right)\right\}\right) p\left(\nu_k; \theta^{\text{old}}\right).$$

(c) **Use the definition of the probabilistic model and of ρ_k, to prove that:**

$$q_\nu(\nu_k) = \mathcal{IG}(\nu_k; \alpha_k, \beta_k), \tag{30}$$

with the following definitions:

$$\alpha_k = \alpha^{\text{old}} + \frac{d_x}{2} \sum_{n=1}^{N} \lambda_{n,k}, \qquad \beta_k = \beta^{\text{old}} + \frac{1}{2} \sum_{n=1}^{N} \lambda_{n,k} \left(\|\mathbf{x}_n - \mathbf{m}_k\|^2 + \text{Tr}(\boldsymbol{\Omega}_k) \right). \tag{31}$$

When we want to prove that something has inverse gamma distribution, we want to find out what is the standard shape of logarithm of p.d.f from $\mathcal{IG}(x, \alpha, \beta)$. We have

$$\begin{aligned} \log \mathcal{IG}(x, \alpha, \beta) &= \alpha \log \beta - \log \Gamma(\alpha) - (\alpha + 1) \log x - \frac{\beta}{x} \\ &\overset{(x)}{=} -(\alpha + 1) \log x - \frac{\beta}{x}. \end{aligned}$$

So, if we get logarithm term and x^{-1} from log p.d.f, we will know that we have inverse gamma distribution. Let us see what will have from $\log q_\nu(\nu_k)$:

$$\log q_\nu(\nu_k) \overset{(\nu_k)}{=} \sum_{n=1}^{N} \lambda_{n,k} \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)} \left\{ \log p \left(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}} \right) \right\} + \log p \left(\nu_k; \theta^{\text{old}} \right).$$

We know that distribution under logarithm under expectation is Gaussian, and the distribution of μ_k is inverse gamma. So, we can rewrite the previous and obtain

$$\begin{aligned} \log q_\nu(\nu_k) &\overset{(\nu_k)}{=} \sum_{n=1}^{N} \lambda_{n,k} \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)} \left\{ -\frac{d_x}{2} \log \nu_k - \frac{1}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \nu_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right\} \\ &\quad - (\alpha^{\text{old}} + 1) \log \nu_k - \frac{\beta^{\text{old}}}{\nu_k} \\ &= -\frac{1}{2} \sum_{n=1}^{N} \lambda_{n,k} \left(d_x \log \nu_k + \frac{1}{\nu_k} \mathbb{E}_{q_{\boldsymbol{\mu}}(\boldsymbol{\mu}_k)} \left\{ \|\mathbf{x}_n - \boldsymbol{\mu}_k)\|^2 \right\} \right) - (\alpha^{\text{old}} + 1) \log \nu_k - \frac{\beta^{\text{old}}}{\nu_k} \\ &= - \left(\frac{d_x}{2} \sum_{n=1}^{N} \lambda_{n,k} + \alpha^{\text{old}} + 1 \right) \log \nu_k \\ &\quad - \left(\frac{1}{2} \sum_{n=1}^{N} \lambda_{n,k} \left(\|\mathbf{x}_n - \mathbf{m}_k)\|^2 + \text{Tr}(\boldsymbol{\Omega}_k) \right) + \beta^{\text{old}} \right) \frac{1}{\nu_k}. \end{aligned}$$

From the previous we can see that ν_k has inverse gamma distribution with parameters

$$\frac{d_x}{2} \sum_{n=1}^{N} \lambda_{n,k} + \alpha^{\text{old}} \text{ and } \frac{1}{2} \sum_{n=1}^{N} \lambda_{n,k} \left(\|\mathbf{x}_n - \mathbf{m}_k\|^2 + \text{Tr}(\boldsymbol{\Omega}_k) \right) + \beta^{\text{old}}.$$

Few remarks: In the first line we used the fact that $p(\mathbf{x}_n \mid \mathbf{z}_n = k, \boldsymbol{\mu}_k, \nu_k; \theta^{\text{old}})$ is p.d.f of normal distribution $\mathcal{N}(\boldsymbol{\mu}_k, \nu_k \mathbf{I}_{d_x})$. We needed just the terms depending of ν_k, so that is the reason why we have additional $-\frac{d_x}{2} \log \nu_k$ which came from $\frac{1}{|\nu_k \mathbf{I}_{d_x}|^{1/2}}$. Also, the expectation in the third line we have already seen in the Question 3.2.2 (b).

(d) **What happens to the posterior of ν_k if no observation is assigned to the k-th cluster?**

We have discussed already that if no observation is assigned to the k-th cluster, then $\lambda_{n,k} = 0$ for all n. This means that posterior for ν_k won't change, i.e. it will remain the same as before $\mathcal{IG}(\alpha^{\text{old}}, \beta^{\text{old}})$. This is of course expected, because if no observation is assigned to the k-th cluster, we cannot conclude anything new about the posterior of ν_k.

2. **Gamma *vs.* inverse gamma:** Consult a referred text that discusses the gamma and inverse gamma distributions (the Wikipedia pages should be enough):

(a) **What is the relation between the gamma and inverse gamma distributions?**

Let X be a random variable from the Gamma distribution $\mathcal{G}(\alpha, \beta)$. We know it has the following p.d.f.:

$$p_X(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} \exp(-\beta x),$$

for $x > 0$. We can define the transformation $Y = f(X) = \frac{1}{X}$, and by the rule of changing the variable we can compute p.d.f. for Y like

$$\begin{aligned} p_Y(y) &= p_X\left(f^{-1}(y)\right)\left(f^{-1}(y)\right)' \\ &= \frac{\beta^\alpha}{\Gamma(\alpha)}\left(\frac{1}{y}\right)^{\alpha-1} \exp\left(-\frac{\beta}{y}\right)\frac{1}{y^2} \\ &= \frac{\beta^\alpha}{\Gamma(\alpha)} y^{-\alpha-1} \exp\left(-\frac{\beta}{y}\right), \end{aligned}$$

for $y > 0$. This proved that if $X \sim \mathcal{G}(\alpha, \beta)$, then, $\frac{1}{X} \sim \mathcal{IG}(\alpha, \beta)$.

(b) **In our work, β is the rate parameter, but the gamma and inverse gamma distributions can be defined with a scale parameter: what is their relationship?**

Gamma distribution can be defined with the shape parameter α and rate parameter β, or with the shape parameter $k = \alpha$ and scale parameter $\theta = \frac{1}{\beta}$. Characterization by the shape and scale parameters is given with

$$\mathcal{G}(x; k, \theta) = \frac{\theta^{-k}}{\Gamma(k)} x^{k-1} \exp\left(-\frac{x}{\theta}\right).$$

The same applies for the Inverse Gamma distribution, and characterization is given by

$$\mathcal{IG}(x; k, \theta) = \frac{\theta^{-k}}{\Gamma(k)} x^{-k-1} \exp\left(-\frac{1}{x\theta}\right).$$

(c) **In the data generation procedure, are we using the "inverse-gamma" or the "gamma" distribution? Are we declaring a shape or rate parameter? And which parameter are we using? What do you conclude?**

In the data generation we used next line for generating the variances

```
np.random.gamma(shape = GroundTruth_alpha, scale = GroundTruth_beta, size = K)
```

If we look at the documentation of the function `numpy.random.gamma` we can see that it is using the next p.d.f. for the generating data

$$p(x) = x^{k-1}\frac{e^{-x/\theta}}{\theta^k\Gamma(k)},$$

where κ is the shape and θ the scale, and Γ is the Gamma function. This means that we used Gamma distribution described with shape k and scale θ. We choose $k = \alpha = 40$ and $k = \frac{1}{\beta} = 0.1$, i.e. $\beta = 10$. We can conclude that we generated data with the Gamma distribution, and we assumed that data is generated with the Inverse Gamma, that was our prior assumption. The reason for choosing the Inverse Gamma as a prior is because it is a *conjugate prior*. If the posterior distributions are in the same probability distribution family as the prior probability distribution, the prior and posterior are then called conjugate distributions, and the prior is called a conjugate prior for the likelihood function. We derived the posterior distribution and made sure it is also from the Inverse Gamma distribution family. With this wrong assumption our model can not estimate β parameter properly. If we want to change this, we should generate data with covariance $\frac{1}{\nu_k}\mathbf{I}_{d_x}$ instead of $\nu_k\mathbf{I}_{d_x}$, because we know that $\frac{1}{\nu_k}$ is from the Inverse Gamma distribution, when ν_k is from the Gamma distribution.

3. **Initialisation procedure: Design a smarter initialisation procedure for the $\mathbf{m}_k$'s. Reevaluate 3.3.3 (c) and 3.3.3 (d), and discuss the plots.**

We choose to initialize the means using the K-Means algorithm. We will fit a K-Means on initial data and the final centers will be the initial guesses for our EM algorithm. We will start by discussing the problems that random initialization has and how it impacts the convergence of EM algorithm. Finally, we will explain why K-Means is a better alternative.

When we randomly initialize the means, we have no guarantee that each mean is generated close to a cluster so that EM will pull them to the center of the cluster and finally have the solution that we identify visually when we look at the data. In the worst case, all means will be randomly generated close one to another around a cluster, as shown in the Figure 7, where the red symbols (square, circle and star) in cluster 0 represent the randomly generated means.

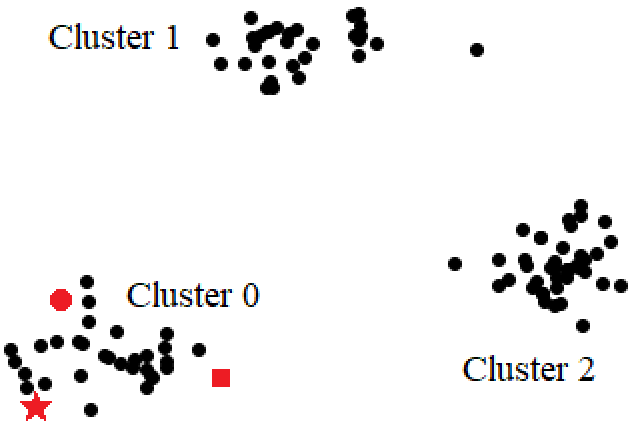


Figure 7: Worst case scenario for random initialization of means

Using the initialization as in the figure, the points in cluster 1 will have higher probability to belong to the cluster denoted by the red circle mean. Also, the points in cluster 2 will have a higher probability to belong to the cluster denoted by the red square mean. But at the end, these points (from cluster 1 and 2) will have low probability because the distance between them and the two means is big (see formula (32) where the distance between observations $\mathbf{x}_n$ and mean $\mathbf{m}_k$ is involved in the exponential).

$$\lambda_{n,k} = \frac{\pi_k^{\text{old}} \, \exp\left(-\frac{1}{2}\left[d_x \eta_k + \rho_k \left(\left\|\mathbf{x}_n - \mathbf{m}_k\right\|^2 + \operatorname{Tr}\left(\boldsymbol{\Omega}_k\right)\right)\right]\right)}{\sum\limits_{\ell=1}^{K} \pi_\ell^{\text{old}} \, \exp\left(-\frac{1}{2}\left[d_x \eta_\ell + \rho_\ell \left(\left\|\mathbf{x}_n - \mathbf{m}_\ell\right\|^2 + \operatorname{Tr}\left(\boldsymbol{\Omega}_\ell\right)\right)\right]\right)} \tag{32}$$

Of course, there would be some points in the cluster 0 that will be assigned to these two means and they will have a higher probability compared to the points in clusters 1 and 2 just because they are closer to the mean. This behavior will cause the red star mean to have a small number of points assigned to it (even zero!) when we do hard assignment. Moreover, an important aspect to note is that the points closer to the mean weigh more in computing the new value for the mean, proportional to the probability assigned. Therefore, the red circle mean will not move towards the points in the cluster 1 because those points are not powerful enough to pull the mean towards them (they have low probability) and the mean will not move outside the area of cluster 0. This is why random initialization is not good and in the end it might lead to an empty cluster, even for this clearly separable data.

As we mentioned, we will use K-Means algorithm in order to initialize the mean vectors. The main difference is that K-means assigns hard labels to data points (zero or one) in contrast to EM in which one point can belong to all clusters with a given probability. This small observation is very important and in the case of K-Means, the points in cluster 1 will cause the red circle mean to move towards cluster 1 because there are a lot of points whose baricenter will be closer to them. Finally the red circle mean will move towards cluster 1 and the red square mean will move towards cluster 2. Of course, when this jump is performed, the red star mean will gain more points from cluster 0 and this way the means will be moved close to their real means.

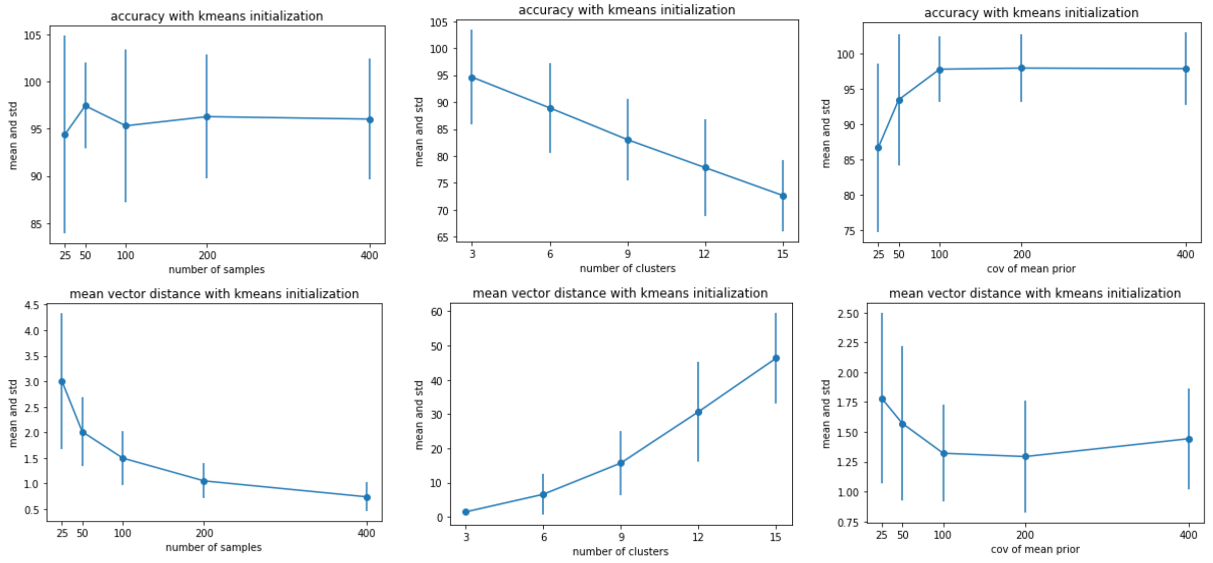
Algorithm 1: Algorithm to initialize means

```
Input: Dataset $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$
Output: Initial mean vectors $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, ..., \boldsymbol{\mu}_K$
Randomly initialize $\boldsymbol{\mu}_1$ $\left(\text{a random value in } \mathbf{X} \text{ or } \boldsymbol{\mu}_1 = \frac{1}{KN}\sum_{i=1}^{n}\mathbf{x}_n = \frac{\bar{\mathbf{x}}}{K}\right)$
for $i \in \{2, ..., K\}$ do
    $\boldsymbol{\mu}_i = \max_n \sum_{j=1}^{i-1} \|\mathbf{x}_n - \boldsymbol{\mu}_j\|^2$
end
return $\{\mu_1, ..., \mu_K\}$
```

Another idea that we have is to perform an iterative optimization with K steps that would choose the K initial mean vectors from the data points such that each data point that we choose as mean maximizes the distance between itself and the previous means. In the Algorithm 1 we provided a small pseudocode for this procedure.

By choosing the means to be as far as possible from each other we make sure that we do not encounter again the scenario corresponding to the random initialization with some points in the dataset. When dividing $\bar{\mathbf{x}}$ by K, the result will move towards a cluster or, if the data is completely mixed, we will avoid choosing the barycenter of the data. Note that this optimization problem can be difficult to solve when dataset $\mathbf{X}$ has many points or when the space of the points is big.

For the purpose of the lab, we will use K-Means to initialize the means because it is easy to use the implementation in Scikit-Learn.



(a) N is variable (b) K is variable (c) ω is variable

Figure 8: Sensitivity analysis plots for N, K, ω for K-Means initialization

Discussing the plots for K-Means initialization

Above we can see the subplots from the Figure 8. Let us discuss each of them.

(a) The accuracy is roughly between 90% and 100%, which means that varying the number of samples does not affect this metric and thus, it is more stable as we have at least 100 points (the standard deviations do not change significantly). The means will attract more samples, they will be computed more accurately and thus the mean vector distance metric is decreasing while the number of samples increases (the estimated means get closer to the ground truth means and the estimation is better because we have more and more points associated to each mean/cluster).

(b) As with the random initialization we can apply the same logic. The more clusters we want to find, the lower the accuracy will be because the number of samples associated to each cluster becomes smaller. Consequently, we will get a poor estimation for the mean which causes the mean distance vector metric to increase, creating a negative correlation with the accuracy because the accuracy is decreasing when the mean vector distance is increasing. This time, unlike the case with the random initialization, we can see clear dependence without noise coming from the random initialization.

(c) Again, we can recall that ω is a measure of spread for data inside the clusters. This means that a small value for ω results in a big spread of clusters. When ω is small, the clusters tend to have a big overlap. In this case, the accuracy is still high, but it can have a bigger value when ω increases because the clusters tend to reduce their spread radius. For our experiment, when $\omega \geq 100$ we have not big improvement for both accuracy and mean vector distance because the clusters are almost separable. We can see again that does not show a big improvement.

To conclude with, we can see that with K-means initialization, we do not have noise from the random initialization and we can better describe dependencies of performances measures from the N, K and ω. Thus, the sensitivity analysis makes more sense.

3.5 Optional additional questions

1. **Assume we do not model ν_k but, its inverse, $\xi_k = \frac{1}{\nu_k}$, the precision. Would the inverse gamma distribution work? What could you use instead?**

   If we want to obtain conjugate prior, in this case inverse gamma would not work. The netural choice would be to try with gamma distribution instead. We have seen that if we assume Inverse Gamma for ν_k we will have again Inverse Gamma for the posterior. This means that we should assume that $\frac{1}{\xi_k}$ has Inverse Gamma distribution, which, as we have seen, means that ξ_k has Gamma distribution.

2. **Assume now that we do not model a variance, ν_k, but a full covariance matrix Σ_k: what probability distribution should we use? Provide its definition and describe its parameters.**

   Note: take into account the fact that covariance matrices have constraints.

   Unlike the previous case with ν_k we can not use Gamma or Inverse Gamma distributions, because they are defined over $\mathbb{R}^+$. This time, we need a distribution defined over symmetric, positive definite matrix-valued random variables, in other words - random matrices. To model Σ we can use the inverse Wishart distribution. It is a probability distribution defined on real-valued positive-definite matrices used as the conjugate prior for the covariance matrix of a multivariate normal distribution. We say $\mathbf{X}$ follows an inverse Wishart distribution, denoted as $\mathbf{X} \sim \mathcal{W}^{-1}(\mathbf{\Psi}, \nu)$, if its inverse $\mathbf{X}^{-1}$ has a Wishart distribution $\mathcal{W}(\mathbf{\Psi}^{-1}, \nu)$. Important identities have been derived for the inverse Wishart distribution. Parameters of the Inverse Wishart distribution are positive definite $p \times p$ matrix $\mathbf{\Psi}$, and the real number $\nu \geq p$ which denots degrees of freedom. The Inverse Wishart distribution can be characterized by its probability density function as follows

$$p(\mathbf{x} \mid \mathbf{\Psi}, \nu) = \frac{|\mathbf{\Psi}|^{\nu/2}}{2^{\nu p/2}\Gamma_p\left(\frac{n}{2}\right)} |\mathbf{x}|^{-(\nu+p+1)/2} \exp\left\{-\frac{1}{2}\operatorname{Tr}(\mathbf{\Psi}\mathbf{x}^{-1})\right\},$$

   where $\mathbf{x}$ is $p \times p$ positive definite matrices, and Γ_p is the multivariate gamma function. Suppose a covariance matrix $\mathbf{\Sigma}$ whose prior $p(\mathbf{\Sigma})$ has a $\mathcal{W}^{-1}(\mathbf{\Psi}, \nu)$ distribution. If the observations $\mathbf{X} =$

$(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n)$ are independent p-variate Gaussian variables drawn from a $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ distribution, then the conditional distribution $p(\mathbf{\Sigma} \mid \mathbf{X})$ has a $\mathcal{W}^{-1}(\mathbf{A}+\mathbf{\Psi}, n+\nu)$ distribution, where $\mathbf{A} = \mathbf{X}\mathbf{X}^T$. Because the prior and posterior distributions are the same family, we say the inverse Wishart distribution is conjugate to the multivariate Gaussian. Due to its conjugacy to the multivariate Gaussian, it is possible to marginalize out (integrate out) the Gaussian's parameter $\mathbf{\Sigma}$. A univariate specialization of the inverse-Wishart distribution is the inverse-gamma distribution. With $p = 1$ (i.e. univariate) and $\alpha = \nu/2$, $\beta = \mathbf{\Psi}/2$ and $x = \mathbf{X}$ the probability density function of the inverse-Wishart distribution becomes

$$p(x \mid \alpha, \beta) = \frac{\beta^\alpha x^{-\alpha-1} \exp(-\beta/x)}{\Gamma_1(\alpha)}$$

i.e., the inverse-gamma distribution, where $\Gamma_1(\cdot)$ is the ordinary Gamma function.

3. **If we compare the model with the standard GMM, the only parameters that remain deterministic are the prior weights π_k? If we wanted them to be also hidden variables, is there a distribution we could use as prior? Provide its definition and describe its parameters.**

Note: take into account that the weights have constraints.

We know that mixture weights have next constraints

$$0 \leq \pi_k \leq 1, \; \forall k = 1, 2, ..., K \text{ and } \sum_{k=1}^{K} \pi_k = 1.$$

So, we need distribution a multivariate distribution over K random variables, such that those constrains are satisfied. For that reason, we can use the Dirichlet distribution, denoted as $\mathrm{Dir}(\mathbf{x}; \boldsymbol{\alpha})$, is a family of continuous multivariate probability distributions parameterized by a vector $\boldsymbol{\alpha}$ of positive reals. It is a multivariate generalization of the beta distribution. The Dirichlet distribution of order $K \geq 2$ with parameters $\alpha_1, \alpha_2, ..., \alpha_K > 0$ has a probability density function with respect to Lebesgue measure on the Euclidean space $\mathbb{R}^{K-1}$ given by

$$\mathrm{Dir}\left(x_1, x_2, ..., x_K; \alpha_1, \alpha_2, ..., \alpha_K\right) = \frac{1}{\mathrm{B}(\boldsymbol{\alpha})} \prod_{i=1}^{K} x_i^{\alpha_i - 1},$$

where the normalizing constant is the multivariate beta function, which can be expressed in terms of the gamma function

$$\mathrm{B}(\boldsymbol{\alpha}) = \frac{\prod\limits_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma\left(\sum\limits_{i=1}^{K} \alpha_i\right)}, \qquad \boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_K).$$

The parameters of the Dirichlet distribution are, therefore, K and $\boldsymbol{\alpha}$. The support of the Dirichlet distribution is the set of K-dimensional vectors $\mathbf{x}$ whose entries are real numbers in the interval $(0, 1)$; furthermore, $\|\mathbf{x}\|_1 = 1$, i.e. the sum of the coordinates is 1. These can be viewed as the probabilities of a K-way categorical event. Another way to express this is that the domain of the Dirichlet distribution is itself a set of probability distributions, specifically the set of K-dimensional discrete distributions. This is one reason why Dirichlet distribution is a good choice for the prior of the mixture weights π_k. Another reason, is of course the fact that Dirichlet distribution will be conjugate prior distribution for the mixture weights. We therefore choose a Dirichlet distribution over the mixing coefficients $\boldsymbol{\pi}$

$$p(\boldsymbol{\pi}) = \mathrm{Dir}(\boldsymbol{\pi} \mid \boldsymbol{\alpha}_0) = \frac{1}{B(\boldsymbol{\alpha}_0)} \prod_{k=1}^{K} \pi_k^{\alpha_0 - 1},$$

where by symmetry we have chosen the same parameter α_0 for each of the components. The parameter α_0 can be interpreted as the effective prior number of observations associated with each component of the mixture. If the value of α_0 is small, then the posterior distribution will be influenced primarily by the data rather than by the prior.