Université de Montréal

IFT6269 - Probabilistic Graphical Models

Homework 3

by :

Patrice Béchard

20019173

3 novembre 2017

1 DGM

The implied factorization for the directed graphical model G for any joint distribution $p(x_V) \in \mathcal{L}(G)$ is given by :

$$p(x_V) = \prod_{i=1}^{N} p_i(x_i \mid x_{\pi_i}) = p(x)p(y)p(z|x,y)p(t|z) \tag{1}$$

It is **not** true that $X \perp Y \mid T$ for any $p \in \mathcal{L}(G)$. We have seen in class that

$$\boxed{\text{if } p \in \mathcal{L}(G) \Leftrightarrow X_A \perp X_B \mid X_C \; \forall \; A, B, C \text{ s. t. } A \;\&\; B \text{ are } \textbf{d-separated} \text{ by } C}$$

We have also seen the definition of **d-separation** :

> Sets A & B are said to be **d-separated** by C iff all chains from $a \in A$ to $b \in B$ are blocked given C, where a chain from a to b is blocked at node d if :
>
> 1. either $d \in C$ and (v_{i-1}, d, v_{i+1}) is not a v-structure.
> 2. $d \notin C$, (v_{i-1}, d, v_{i+1}) is a v-structure and no descendents of d is in C.

In our case, nodes X and Y are not d-separated by node T since (X, Z, Y) is a v-structure and T is a descendent of Z, which does not respect the second condition for v-separation.

2 d-separation in DGM

(a) No

(b) Yes

(c) No

(d) Yes

(e) No

(f) No

(g) Yes

(h) No

(i) Yes

(j) No

3 Positive interactions in-V-structures

(a)

(i) For the condition $a > c$, let's use the following conditional probability tables :

X	p(x)
1	0.5
0	0.5

Y	p(y)
1	0.5
0	0.5

X	Y	p(Z=1 | x,y)
1	1	0.1
1	0	1
0	1	0.2
0	0	1

We can now easily compute the different wanted values :

$$\begin{aligned} a &= P(X = 1) \\ &= 0.5 \end{aligned} \tag{2}$$

$$\begin{aligned} b = P(X = 1 \mid Z = 1) &= \frac{P(X = 1, Z = 1)}{P(Z = 1)} \\ &= \frac{\sum_{i \in \{0,1\}} P(X = 1)P(Y = i)P(Z = 1 \mid X = 1, Y = i)}{\sum_{i,j \in \{0,1\}} P(X = i)P(Y = j)P(Z = 1 \mid X = i, Y = j)} \\ &\approx 0.478 \end{aligned} \tag{3}$$

$$\begin{aligned} c = P(X = 1 \mid Z = 1, Y = 1) &= \frac{P(X = 1, Z = 1, Y = 1)}{P(Z = 1, Y = 1)} \\ &= \frac{P(X = 1)P(Y = 1)P(Z = 1 \mid X = 1, Y = 1)}{\sum_{i \in \{0,1\}} P(X = i)P(Y = 1)P(Z = 1 \mid X = i, Y = 1)} \\ &\approx 0.333 \end{aligned} \tag{4}$$

(ii) Now, for the condition $a < c < b$, let's use the following CPTs :

X	p(x)
1	0.5
0	0.5

Y	p(y)
1	0.5
0	0.5

X	Y	p(Z=1 | x,y)
1	1	1
1	0	0.5
0	1	0.5
0	0	0

We can now easily compute a, b and c using equations 2, 3 and 4.

$$a = 0.5 \qquad b = 0.75 \qquad c \approx 0.667$$

(iii) $b < a < c$

X	p(x)
1	0.5
0	0.5

Y	p(y)
1	0.5
0	0.5

X	Y	p(Z=1 | x,y)
1	1	1
1	0	0
0	1	0.1
0	0	1

a, b and c can now be easily computed :

$$a = 0.5 \qquad b = 0.4 \qquad c \approx 0.667$$

(b)

(i) If the probability of $Z = 1$ is slim if $Y = 1$, then the probability of $X = 1 \mid Z = 1, Y = 1$ will be smaller than the probability of $X = 1$ alone.

(ii) If it is highly probable that we obtain $Z = 1$ if $X = 1$, then $b > a$. However, if the probability of having $Z = 1$ given $Y = 1$ is high as well, then the probability of having $X = 1 \mid Y = 1, Z = 1$ will be lower than if we don't know the value of Y yet.

(iii) If it is certain for event Z to happen if both events X and Y happen at the same time or do not happen, but it is also possible for Z to happen if $Y = 1$ and $X = 0$ in some rare cases (the opposite case where $Z = 1$ when $X = 1, Y = 0$ is impossible). If we know that Z happened, chances are X and Y happened (or did not happen) at the same time since the other option is a lot less probable. If we know that $Y = 1$, it is almost certain that $X = 1$ as well, but if we don't know the result of Y, the probability of $X = 1 \mid Z = 1$ is just a bit less than the probability that $X = 1$.

4 Flipping a covered edge in a DGM

We have that $\mathcal{L}(G) = \{p : p(x_V) = p(i|\pi_i)p(j|\pi_i, i)p_*\}$ and $\mathcal{L}(G') = \{p : p'(x_V) = p(i|\pi_i, j)p(j|\pi_i)p_*$, where p_* represents the rest of the distribution, which is identical for both graphs. We can prove that the family of distributions represented by both of these graphs are identical by showing that the factorization of joint distribution is identical :

$$
\begin{aligned}
p(x_{_V}) &= p_* p(i|\pi_i) p(j|i,\pi_i) \\
&= p_* \frac{\cancel{p(i,\pi_i)}}{p(\pi_i)} \frac{p(i,j,\pi_i)}{\cancel{p(i,\pi_i)}} && \text{(By definition)} \\
&= p_* \frac{p(i,j,\pi_i)}{p(\pi_i)} \frac{p(j,\pi_i)}{p(j,\pi_i)} && \text{(Multiplying by 1)} \\
&= p_* \frac{p(i,j,\pi_i)}{p(j,\pi_i)} \frac{p(j,\pi_i)}{p(\pi_i)} && \text{(Rearranging)} \\
&= p_* p(i|j,\pi_i) p(j|\pi_i) && \text{(By definition)} \\
&= p'(x_{_V}) \quad \blacksquare
\end{aligned}
$$

5 Equivalence of directed tree DGM with undirected tree UGM

For $G(V,E)$ a directed acyclic graph (DAG), we call $\bar{G}(V,\bar{E})$ the *moralized graph* of G where $\bar{G}$ is an undirected graph with the same nodes V as G and edges $\bar{E}$ given by :

$$\bar{E} = \{\{i,j\} : (i,j) \in E\} \cup \{\{k,j\} : k \neq l \in \pi_i \text{ for some } i\}$$

In our case, since G is a directed tree (does not contain v-structures), no additional edges appear, meaning that $\bar{E} = E$ (except for the fact that the edges are now undirected). For the moralized graph $\bar{G}$ obtained, the family of distributions represented by it are :

$$\mathcal{L}(\bar{G}) = \left\{ p : \bar{p}(x_{_V}) = \frac{1}{Z} \prod_{c \in \mathcal{C}} \psi_c(x_c) \right\}$$

where $\psi_c(x_c) \geq 0 \; \forall \; x_c$ are the *potentials* and $Z = \sum_x \prod_{c \in \mathcal{C}} \psi_c(x_c)$ is the *partition function*. Since the moralization of the directed tree G did not yield any new edge, nodes are only fully connected 2x2, which means there is no new cliques formed. With $Z = 1$, each potential representing the edge between two nodes in the undirected graph still represents the same probability as it was representing in its directed form :

$$\psi_{\{a,b\}}(x_a, x_b) = p(b \mid a)$$

This yields $\bar{p}(x_{_V}) = p(x_{_V})$, and finally we get that $\mathcal{L}(G) = \mathcal{L}(\bar{G})$.

6 Hammersley-Clifford Counter example

For this problem, we consider a simple undirected 4-cycle graph ($X_1 - X_2 - X_3 - X_4 - X_1$), where each node is a binary random variable X_i for $i \in \{1, 2, 3, 4\}$. The probability distribution is 1/8 for each of these configurations :

$$(0,0,0,0) \quad (0,0,0,1) \quad (0,0,1,1) \quad (0,1,1,1)$$
$$(1,1,1,1) \quad (1,1,1,0) \quad (1,1,0,0) \quad (1,0,0,0)$$

and is 0 for all other configurations. We can first show that the graph respects the global Markov property :

> We say that p satisfies *global Markov property* with respect to undirected graph G iff
>
> $\forall\, A, B, S \subseteq V$ such that S separates A from B in G, then $X_A \perp X_B \mid X_S$

It is easy to see that for p to satisfy global Markov property, we must have that $X_1 \perp X_3 \mid X_2, X_4$ and that $X_2 \perp X_4 \mid X_1, X_3$. The graph tells us that we should be able to interchange X_1 with X_3 as well as X_2 with X_4 and still obtain the same distribution, i.e. $p(x_1, x_2, x_3, x_4) = p(x_4, x_3, x_2, x_1)$. Developing for all possible configurations :

$$\begin{aligned}
p(0,0,0,0) = p(0,0,0,0) &= 1/8 & p(1,0,0,0) = p(0,0,0,1) &= 1/8 \\
p(0,1,0,0) = p(0,0,1,0) &= 0 & p(1,1,0,0) = p(0,0,1,1) &= 1/8 \\
p(1,0,1,0) = p(0,1,0,1) &= 0 & p(1,0,0,1) = p(1,0,0,1) &= 0 \\
p(0,1,1,0) = p(0,1,1,0) &= 0 & p(1,1,1,0) = p(0,1,1,1) &= 1/8 \\
p(1,1,0,1) = p(1,0,1,1) &= 0 & p(1,1,1,1) = p(1,1,1,1) &= 1/8
\end{aligned}$$

We notice that for whatever pair of values (X_2, X_4) we choose, we know either X_1 or X_3 with certainty. If $(X_2, X_4) = (0, 0)$, then $X_3 = 0$. If $(X_2, X_4) = (0, 1)$, then $X_1 = 0$. If $(X_2, X_4) = (1, 0)$, then $X_1 = 1$. Finally, if $(X_2, X - 4) = (1, 1)$, then $X_3 = 1$. This proves conditional independence between the X_1 and X_3 given X_2, X_4 for all possible cases.

We can finally prove that the distribution cannot be factorized in the way stated in the Hammersley-Clifford Theorem. Assuming that the distribution *can* be factorized is such way, noting that the cliques in the graph are only the edges and absorbing the partition function Z, we have the following factorization :

$$p(x_1, x_2, x_3, x_4) = \psi_{\{1,2\}}(x_1, x_2)\psi_{\{2,3\}}(x_2, x_3)\psi_{\{3,4\}}(x_3, x_4)\psi_{\{4,1\}}(x_4, x_1)$$

Knowing that the following equations are true :

$$\begin{aligned}
p(0,0,1,0) &= \psi_{\{1,2\}}(0,0)\psi_{\{2,3\}}(0,1)\psi_{\{3,4\}}(1,0)\psi_{\{4,1\}}(0,0) = 0 \\
p(0,0,0,0) &= \psi_{\{1,2\}}(0,0)\psi_{\{2,3\}}(0,0)\psi_{\{3,4\}}(0,0)\psi_{\{4,1\}}(0,0) = 1/8 \\
p(0,0,1,1) &= \psi_{\{1,2\}}(0,0)\psi_{\{2,3\}}(0,1)\psi_{\{3,4\}}(1,1)\psi_{\{4,1\}}(1,0) = 1/8
\end{aligned}$$

This means that $\psi_{\{3,4\}}(1,0)$ must be 0. However, we also know that the following statement is true :

$$p(1,1,1,0) = \psi_{\{1,2\}}(1,1)\psi_{\{2,3\}}(1,1)\psi_{\{3,4\}}(1,0)\psi_{\{4,1\}}(0,1) = 1/8$$

This yields a contradiction, so the distribution cannot be factorized according to G. ■

7 [Bonus] : bizarre conditional independence properties

$$X \perp Y \mid Z \Leftrightarrow p(x, y \mid z) = p(x \mid z)p(y \mid z) \qquad X \perp Y \Leftrightarrow p(x, y) = p(x)p(y)$$

$$\begin{aligned}
p(x, y \mid z) &= \frac{p(z \mid x, y)p(x, y)}{p(z)} && \text{by Bayes rule} \\
&= \frac{p(z \mid x, y)p(x)p(y)}{p(z)} && \text{by hypothesis}
\end{aligned}$$

We also have that :

$$\begin{aligned}
p(x, y \mid z) &= p(x \mid z)p(y \mid z) && \text{by hypothesis} \\
&= \frac{p(z \mid x)p(x)}{p(z)} \frac{p(z \mid y)p(y)}{p(z)} && \text{by Bayes rule}
\end{aligned}$$

Putting both expressions together, we get :

$$\begin{aligned}
\frac{p(z \mid x)\cancel{p(x)}}{\cancel{p(z)}} \frac{p(z \mid y)\cancel{p(y)}}{p(z)} &= \frac{p(z \mid x, y)\cancel{p(x)}\cancel{p(y)}}{\cancel{p(z)}} \\
\frac{p(z \mid x)p(z \mid y)}{p(z)} &= p(z \mid x, y)
\end{aligned}$$

8 Implementation : EM and Gaussian mixtures

(a) The implementation of the K-means algorithm is found in the class `K_Means` in the file `hwk3.ipynb`. We present various figures obtained with different random initializations for the cluster means as well as the final objective function values obtained after convergence for each case in figures 1, 2 and 3.

(b) For a Gaussian mixture model, we have to optimize parameters $\theta = (\pi, \{\mu_j\}_{j=1}^k, \{\Sigma_j\}_{j=1}^k)$. Knowing that the random variables $Z_{1:n}$ and $X_{1:n}$ are distributed as such :

$$Z_i \sim \text{Mult}(\pi) \qquad X_i \mid Z_{i=j} \sim \mathcal{N}(\mu_j, \Sigma_j)$$

we have the complete log-likelihood given by :

$$\begin{aligned}
\log p(x, z; \theta) &= \sum_{i=1}^{n} \left[\log p(x_i \mid z_i; \theta) + \log p(z_i; \theta)\right] \\
&= \sum_{i=1}^{n} \left[\sum_{j=1}^{k} z_{i,j} \log \mathcal{N}(x_i \mid \mu_j, \Sigma_j) + \sum_{j=1}^{k} z_{i,j} \log \pi_j\right] \\
\mathbb{E}_q\left[\log p(x, z; \theta)\right] &= \sum_{i=1}^{n} \sum_{j=1}^{k} \mathbb{E}_q[z_{i,j}] \left[\log \mathcal{N}(x_i \mid \mu_j, \Sigma_j) + \log \pi_j\right]
\end{aligned}$$

The E step of the EM algorithm consists of computing the weights $\tau_{i,j}^t$ for all i, j, which are given by :

$$\begin{aligned}
\tau_{i,j}^t \triangleq p(z_{i,j} = 1 \mid x_i, \theta_t) &= q_{t+1}(z_{i,j} = 1) \\
&= \frac{\pi_j^{(t)} \mathcal{N}(x_i \mid \mu_j^{(t)}, \Sigma_j^{(t)})}{\sum_{l=1}^{k} \pi_l^{(t)} \mathcal{N}(x_i \mid \mu_l^{(t)}, \Sigma_j^{(t))}}
\end{aligned}$$

Following the E step, the M step consists of finding the maximum likelihood estimator for each parameter using the weights $\tau_{i,j}^t$ instead of $\mathbb{E}_q[z_{i,j}]$ in the complete log-likelihood as such :

$$\max_{\{\mu_j, \Sigma_j, \pi_j\}} \sum_{i=1}^{n} \sum_{j=1}^{k} \tau_{i,j}^{(t)} \left[\log \mathcal{N}(x_i \mid \mu_j, \Sigma_j) + \log \pi_j\right] \tag{5}$$

As usual, to maximize, we have to derivate the expression with respect to the wanted parameter and then set to 0. We first assume that the gaussians are spherical (i.e the covariance matrix is proportional to the identity matrix : $\Sigma_j = \sigma_j^2 I$). We also assume that the terms $\tau_{i,j}$ are constant

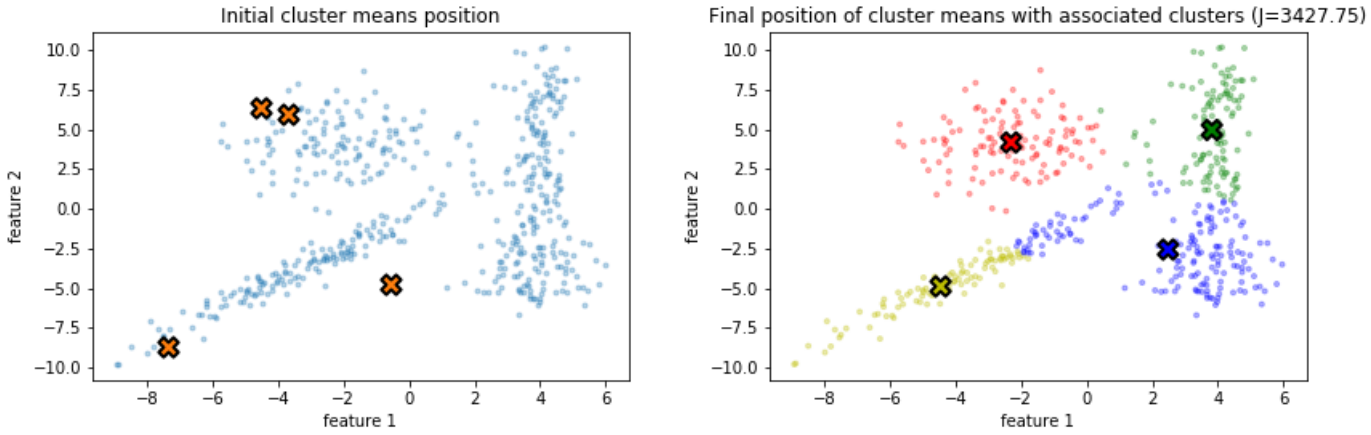

FIGURE 1 – Random initialization of cluster means (left) and final cluster means with associated data point (right) using the K-means algorithm implementation. The final objective function value obtained is $J = 3427.75$.

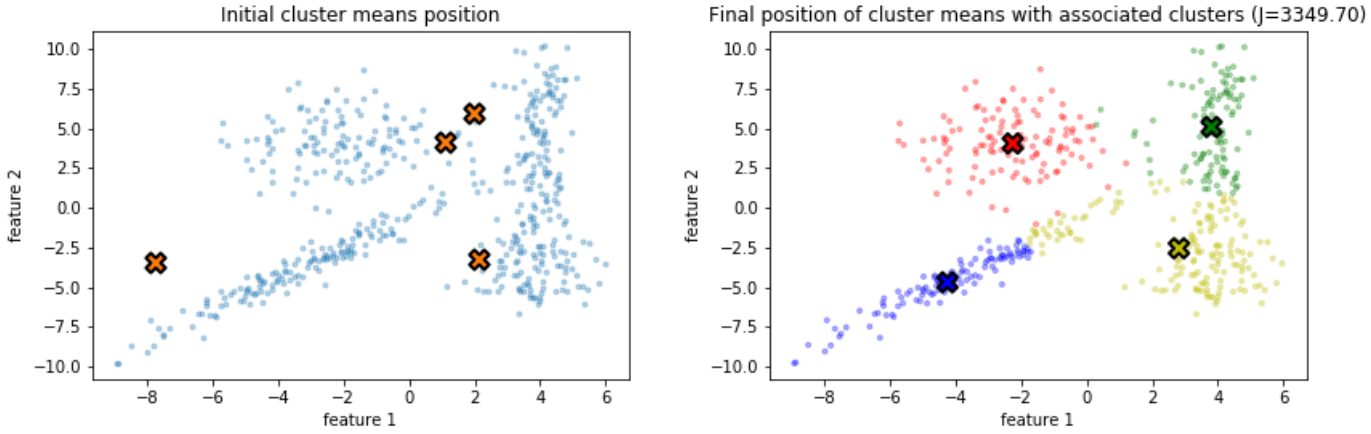

FIGURE 2 – Random initialization of cluster means (left) and final cluster means with associated data point (right) using the K-means algorithm implementation. The final objective function value obtained is $J = 3349.70$.

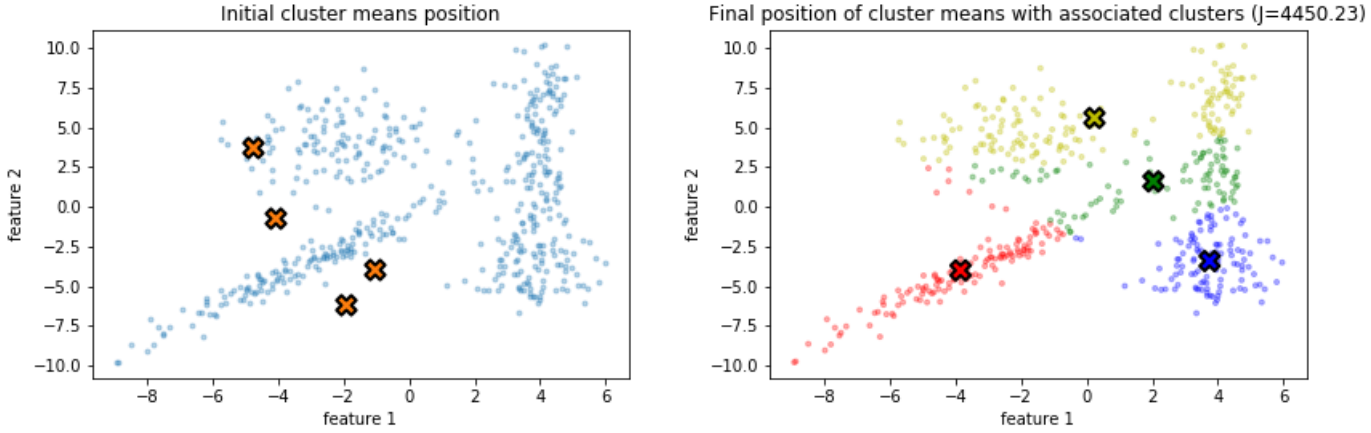

FIGURE 3 – Random initialization of cluster means (left) and final cluster means with associated data point (right) using the K-means algorithm implementation. The final objective function value obtained is $J = 4450.23$.

with respect to the parameter by which we derivate. To find the expression for π_j, we introduce the Lagrange multiplier λ, knowing that $\sum_{j=1}^{k} \pi_j = 1$, and we solve for π_j :

$$\frac{\partial}{\partial \pi_j} \left[\sum_{i=1}^{n} \sum_{j'=1}^{k} \tau_{i,j'} \left[\cancel{\log \mathcal{N}(x_i \mid \mu_{j'}, \sigma_{j'}^2)} + \log \pi_{j'} \right] + \lambda \left(\sum_{j'=1}^{k} \pi_{j'} - 1 \right) \right] = 0$$

$$\begin{aligned}
\sum_{i=1}^{n} \frac{\tau_{i,j}}{\pi_j} + \lambda &= 0 \\
\sum_{i=1}^{n} \tau_{i,j} &= -\lambda \pi_j \\
\sum_{j=1}^{k} \sum_{i=1}^{n} \tau_{i,j} &= \sum_{j=1}^{k} \lambda \pi_j \\
\sum_{i=1}^{n} \sum_{j=1}^{k} \tau_{i,j} &= \lambda \sum_{j=1}^{k} \pi_j
\end{aligned} \tag{6}$$

Noticing that both π_j and $\tau_{i,j}$ summed over j both equal 1, we find that $\lambda = -n$. We finally get the result we want, coming back to equation 6 :

$$\boxed{\hat{\pi}_j = \frac{1}{n} \sum_{i=1}^{n} \tau_{i,j}} \tag{7}$$

We can now do the same for the mean μ_j. Starting again from equation 5, we have :

$$\begin{aligned}
\frac{\partial}{\partial \mu_j} \left[\sum_{i=1}^{n} \sum_{j'=1}^{k} \tau_{i,j'} \left[\log \mathcal{N}(x_i \mid \mu_{j'}, \sigma_{j'}^2) + \cancel{\log \pi_{j'}} \right] \right] &= 0 \\
\sum_{i=1}^{n} \frac{\partial}{\partial \mu_j} \left[\sum_{j'=1}^{k} \tau_{i,j'} \left[\cancel{-\frac{d}{2} \log 2\pi\sigma_{j'}^2} - \frac{1}{2\sigma_{j'}^2} (x_i - \mu_{j'})^T (x_i - \mu_{j'}) \right] \right] &= 0 \\
\sum_{i=1}^{n} \frac{1}{\sigma_j^2} \tau_{i,j} (x_i - \mu_j) &= 0
\end{aligned}$$

Separating into two distinct summations and cancelling the σ_j^2's, we have :

$$\begin{aligned}
\sum_{i=1}^{n} \tau_{i,j} x_i &= \sum_{i=1}^{n} \tau_{i,j} \mu_j \\
\boxed{\hat{\mu}_j = \frac{\sum_{i=1}^{n} \tau_{i,j} x_i}{\sum_{i=1}^{n} \tau_{i,j}}}
\end{aligned} \tag{8}$$

Finally, let's do the same thing for σ_j^2, the last parameter to maximize.

$$\frac{\partial}{\partial \sigma_j^2}\left[\sum_{i=1}^{n}\sum_{j'=1}^{k}\tau_{i,j'}\left[\log \mathcal{N}(x_i \mid \mu_{j'}, \sigma_{j'}^2) + \log \pi_{j'}\right]\right] = 0$$

$$\sum_{i=1}^{n}\frac{\partial}{\partial \sigma_j^2}\left[\sum_{j'=1}^{k}\tau_{i,j'}\left[-\frac{d}{2}\log 2\pi\sigma_{j'}^2 - \frac{1}{2}(x_i-\mu_{j'})^T(x_i-\mu_{j'})\left(\frac{1}{\sigma_j^2}\right)\right]\right] = 0$$

$$\sum_{i=1}^{n}\tau_{i,j}\left[-\frac{d}{2\sigma_j^2} + \frac{(x_i-\mu_j)^T(x_i-\mu_j)}{2(\sigma_j^2)^2}\right] = 0$$

Multiplying both sides by $2\sigma_j^4$, we obtain :

$$\sum_{i=1}^{n}\tau_{i,j}\left[-d\sigma_j^2 + (x_i-\mu_j)^T(x_i-\mu_j)\right] = 0$$

$$\sum_{i=1}^{n}\tau_{i,j}(x_i-\mu_j)^T(x_i-\mu_j) = \sum_{i=1}^{n}\tau_{i,j}d\sigma_j^2$$

$$\boxed{\hat{\sigma}_j^2 = \frac{\sum_{i=1}^{n}\tau_{i,j}(x_i-\mu_j)^T(x_i-\mu_j)}{d\sum_{i=1}^{n}\tau_{i,j}}} \tag{9}$$

We are now ready to implement the gaussian mixture model with the EM algorithm for spherical gaussians. The class `GMM` of the file `hwk3.ipynb` does precisely that for both the spherical gaussian (covariance matrix proportional to identity) and the full covariance matrix. Figures 4, 5 and 6 presents the data and the cluster means after initialization using k-means clustering on the right and the clustering obtained using the gaussian mixture model with the EM algorithm on the right. We present the ellipse containing 95.45% of the mass of the Gaussian distribution (2 standard deviations) as well as the most likely latent variables for all data points.

(c) Figures 7, 8 and 9 presents the data and the cluster means after initialization using k-means clustering on the right and the clustering obtained using the gaussian mixture model with the EM algorithm on the right. We present the ellipse containing 95.45% of the mass of the Gaussian distribution (2 standard deviations) as well as the most likely latent variables for all data points.

(d) By looking at the different plots generated, we can easily see that the best model for the data is the gaussian mixture model with full covariance matrix, which constantly returns really similar results, while the other models return results that differ depending on the initialization.

We can compare the normalized log-likelihood of the two mixture models on the training data found in the file `EMGaussian.train` as well as the test data found in `EMGaussian.test`. Knowing that the a cost function largely used in unsupervised learning is $-\log p(X;\Theta)$, we thus know that the smaller is this value on the test data, the better our model is. Table 1 presents the results obtained for each mixture model implemented.

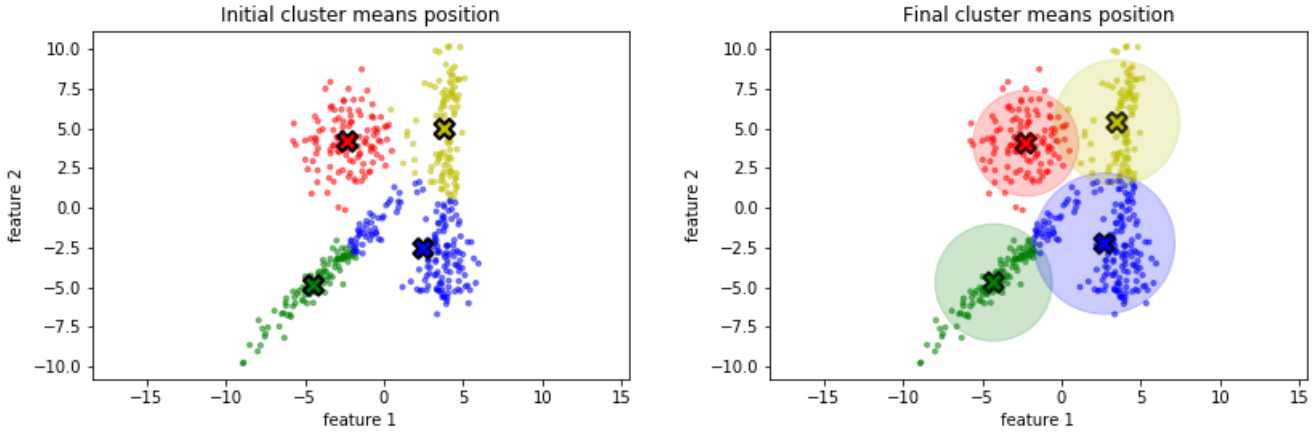

FIGURE 4 – Initialization of cluster means with K-means algorithm(left) and final cluster means with associated data point (right) using the gaussian mixture model and EM algorithm implementation using isotropic gaussians.

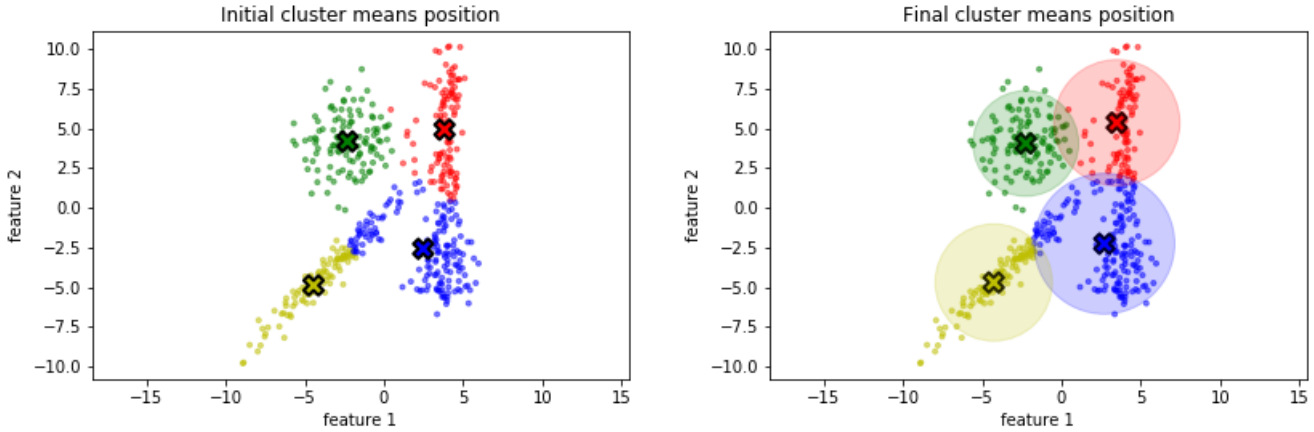

FIGURE 5 – Initialization of cluster means with K-means algorithm(left) and final cluster means with associated data point (right) using the gaussian mixture model and EM algorithm implementation using isotropic gaussians.

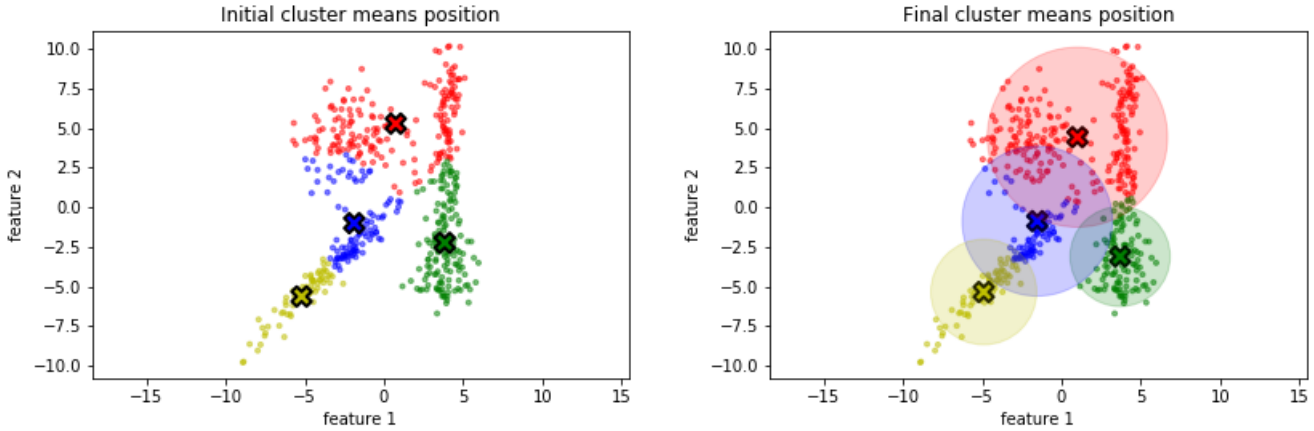

FIGURE 6 – Initialization of cluster means with K-means algorithm(left) and final cluster means with associated data point (right) using the gaussian mixture model and EM algorithm implementation using isotropic gaussians.

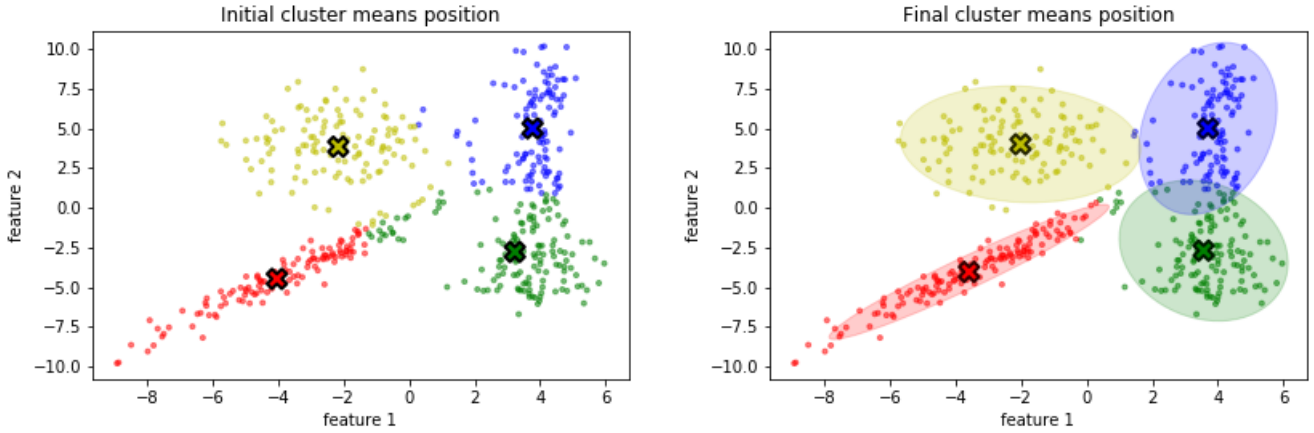

FIGURE 7 – Initialization of cluster means with K-means algorithm(left) and final cluster means with associated data point (right) using the gaussian mixture model and EM algorithm implementation using gaussians with full covariance matrices.

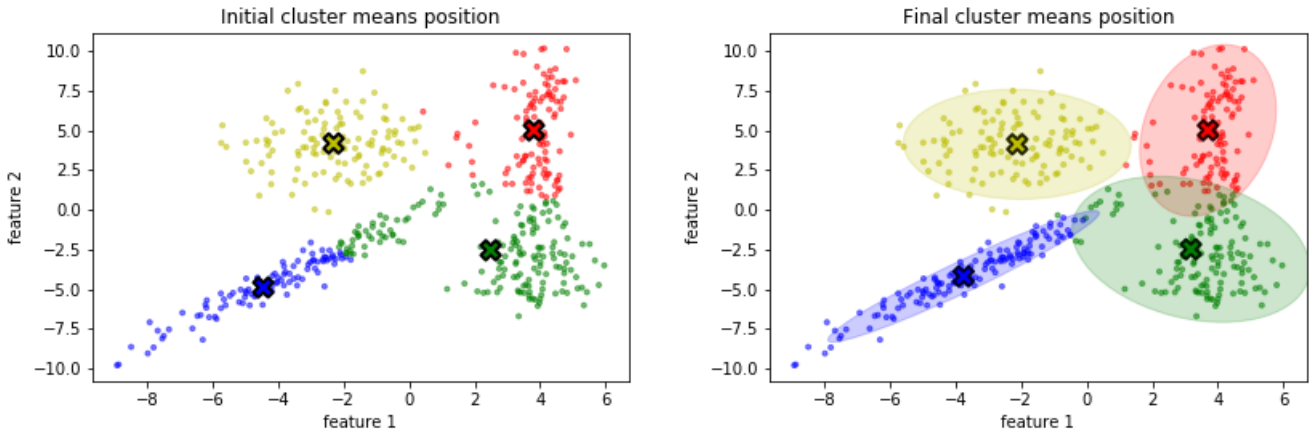

FIGURE 8 – Initialization of cluster means with K-means algorithm(left) and final cluster means with associated data point (right) using the gaussian mixture model and EM algorithm implementation using gaussians with full covariance matrices.

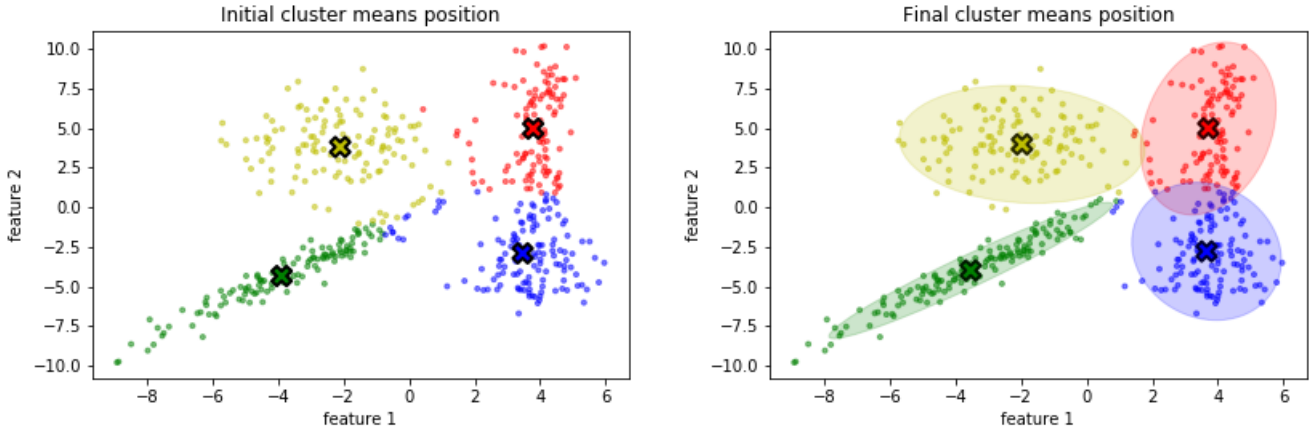

FIGURE 9 – Initialization of cluster means with K-means algorithm(left) and final cluster means with associated data point (right) using the gaussian mixture model and EM algorithm implementation using gaussians with full covariance matrices.

	Isotropic covariance matrix	Full covariance matrix
Training data	5.43	4.85
Test data	5.38	4.99

TABLE 1 – Negative normalized log-likelihood obtained with both mixture models that were implemented.

We see that the model using full covariance matrix is better than the other, obtaining a smaller negative normalized log-likelihood than the other model. We also see that both model do not tend to overfit the data, getting a similar negative normalized log-likelihood for both the training data and the test data in both cases.

It would be interesting in the future to implement the k-mean++ initialization of the cluster means to obtain more consistent and precise results.