HW 2 PGM

Arthur Lavergne and Tamim El Ahmad

31 décembre 2019

Classification : K-Means and the EM algorithm

question 1

On a X_i pour i dans $\{1, ..., n\}$ avec la probabilité p_k d'être dans la composante k. De plus, on sait que l'on a K composantes, et que la loi conditionnelle de $X_i|Z_i = k$ est telle que : $X_i|Z_i = k \hookrightarrow \mathcal{N}(\mu_k, D_k)$ On pose donc les paramètre à trouver $\theta = (p, \mu, D)$ On a donc :

$$p(Z) = \sum_{k=1}^{K} p_k^{z_k}$$
$$p(X|Z; (_m u_k, D_k)) = \sum_{k=1}^{K} z_k \mathcal{N}(X; \mu_k, D_k)$$
$$p(X) = \sum_{k=1}^{K} p_k \mathcal{N}(X; \mu_k, D_k))$$

On cherche donc à estimer $argmax_{mu_k, D_k} log(p(X))$ A partir des information précédentes, on peut appliquer la méthode du maximum de vraisemblance au problème de la mixture gaussienne :

$$\mathcal{Z} = \{z \in \{0,1\}^K | \sum_{k=1}^{K} p_k = 1\}$$

$$\text{donc} \quad p(X) = \sum_{z \in \mathcal{Z}} p(X, Z) = \sum_{z \in \mathcal{Z}} \prod_{k=1}^{K} [p_k \mathcal{N}(X, \mu_k, D_k)]^{z_k}$$
$$= \sum_{i=1}^{K} p_k \mathcal{N}(x|\mu_k, D_k)$$

En appliquant le principe de l'étape E vu en cours, cela revient à calculer à l'itération t+1 :

$$F_{t+1}(\Pi,\mu,\Sigma) = \sum_{i=1}^{n}\sum_{j=1}^{k}\tau_{i,t}^{j}log(\Pi_j) + \sum_{i=1}^{n}\sum_{j=1}^{k}\tau_{i,t}^{j}[log(\frac{1}{(2\pi)^{k/2}} + log(\frac{1}{|\Sigma_j|)^{1/2}}]$$
$$-\frac{1}{2}(x_i-\mu_j)^T\Sigma_j^{-1}(x_i-\mu_j)$$

<u>Estimation de Π :</u> $\text{argmax}F_{t+1}(\Pi,\mu,\Sigma) = \text{argmax}_{\Pi}\sum_{i=1}^{n}\sum_{j=1}^{k}\tau_{i,t}^{j}log(\Pi_j)$ C'est à dire qu'on cherche à maximiser la valeur d'une distribution multinomiale, on a don directement :

$$\Pi_{j,t+1} = \frac{1}{n}\sum_{i=1}^{n}\tau_{i,t}^{j} \quad \forall j \tag{1}$$

<u>Estimation de μ</u> : $-\sum_{i=1}^{n}\sum_{j=1}^{k}\tau_{i,t}^{j}(x_i-\mu_j)^T(x_i-\mu_j)$ Cette fonction est concave en μ, on prend donc le gradient égale à zero ce qui nous donne l'équation suivante :

$$\sum_{i=1}^{n}\tau_{i,t}^{j}(x_i-\mu_{j,t+1}) = 0 \quad \forall j \tag{2}$$

On a donc à t+1 :

$$\mu_{j,t+1} = \frac{\sum_{i=1}^{n}\tau_{i,t}^{j}x_i}{\sum_{i=1}^{n}\tau_{i,t}^{j}} \quad \forall j \tag{3}$$

Comme dans le cours, pour trouver l'expression de Σ on prend le gradient de l'expression par rapport à Σ égale à 0 on en déduit que :

$$\Sigma_{j,t+1} = \frac{\sum_{i=1}^{n}(x_i-\mu_{j,t+1})(x_i-\mu_{j,t+1})^T\tau_{i,t+1}^{j}}{\sum_{i=1}^{n}\tau_{i,t+1}^{j}} \tag{4}$$

Ici, on suppose les matrices de covariance $\Sigma_{j,t}$ étant des matrices diagonales $D_{j,t}$ dans le cadre de l'exercice. On note chaque élément de la diagonale $d_{j,t}^{l}$ pour $l = 1, ..., d$. Calculons maintenant le gradient de la log likelihood le long d'un $d_{j,t}^{l}$:

$$\begin{aligned}\log(\frac{1}{|D_{j,t}|^{\frac{1}{2}}}) &= -\frac{1}{2}\log(\prod_{l=1}^{d}d_{j,t}^{l}) \\ &= -\frac{1}{2}\sum_{l=1}^{d}\log(d_{j,t}^{l})\end{aligned}$$

De plus, en notant $x_i - \mu_{j,t} = (a_1^i, ..., a_p^i)^T$:

$$(x_i-\mu_{j,t})^T D_{j,t}^{-1}(x_i-\mu_{j,t}) = \sum_{l=1}^{d}\frac{{a_l^i}^2}{d_{j,t}^{l}}$$

Donc pour tout j et pour tout l, on obtient $d^l_{j,t+1}$ en résolvant l'équation suivante :

$$\frac{\partial F_{t+1}}{\partial d^l_{j,t+1}} = \sum_{i=1}^{n} \tau_i^j (-\frac{1}{2}\frac{1}{d^l_{j,t+1}} + \frac{1}{2}\frac{a_l^{i2}}{d^l_{j,t+1}}) = 0$$

Et donc :

$$d^l_{j,t+1} = \frac{\sum_{i=1}^{n} \tau_i^j a_l^{i2}}{\sum_{i=1}^{n} \tau_i^j} \tag{5}$$

$$d^l_{j,t+1} = \frac{\sum_{i=1}^{n} \tau_i^j (x_i^l - \mu^l_{j,t+1})^2}{\sum_{i=1}^{n} \tau_i^j} \tag{6}$$

Ce qui nous donne la même formule que précédemment en ne gardant que la diagonale. Choisir une matrice diagonale permet de diminuer la complexité de l'algorithme car au lieu d'avoir $\frac{d(d+1)}{2}$ paramètres à calculer, on a plus que d. Par contre cette suppositions peut faire intervenir plus d'erreurs de classification car on l'hypothèse de matrice diagonale est bien plus forte et donc peut vraisemblable.

question 3

Afin de restreindre la longueur du documents nous nous contenteront de présenter les résultats sur les graphes représentants la "sepal width" en fonction de la "petal lenght".

Pour K = 2,3 et 4, nous avons les résultats suivants :

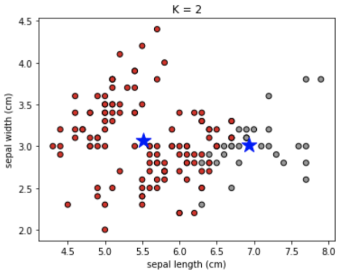

FIGURE 1 – k means avec K = 2

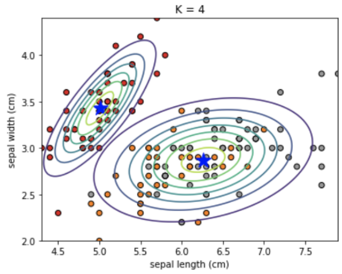

FIGURE 2 – EM isentropique avec K = 2

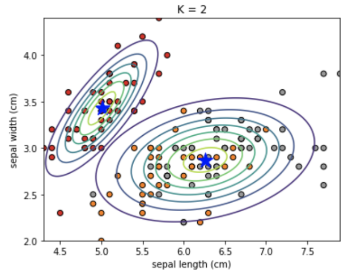

FIGURE 3 – EM avec matrice pleine et K = 2

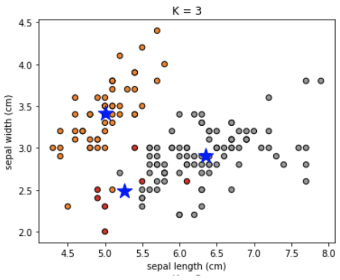

FIGURE 4 – k means avec K = 3

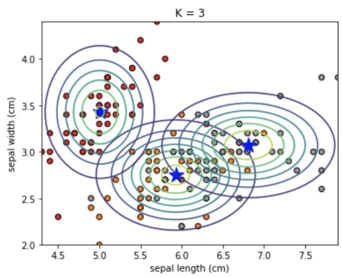

FIGURE 5 – EM isentropique avec K = 3

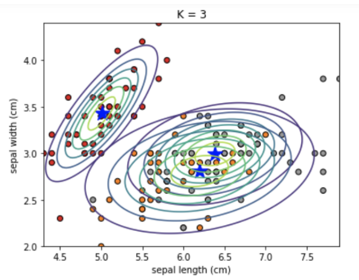

FIGURE 6 – EM avec matrice pleine et K = 3

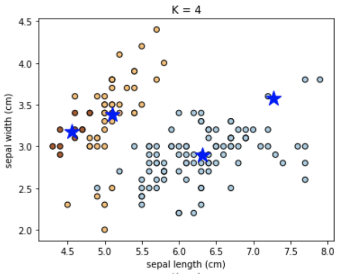

FIGURE 7 – k means avec K = 4

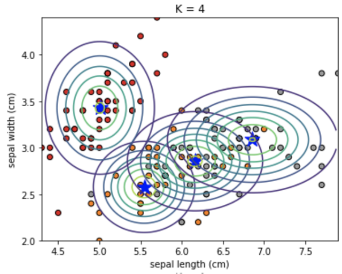

FIGURE 8 – EM isentropique avec K = 4

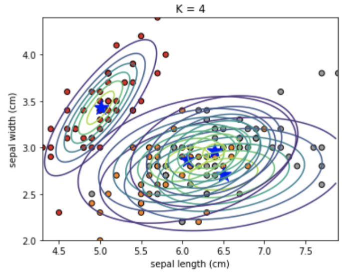

FIGURE 9 – EM avec matrice pleine et K = 4

4°/

Afin de comparer les performances des algorithmes, nous avons générer des données de telle manière ce que le clusters soient disposés de manière circulaire et concentrique. C'est à dire qu'un cluster de données est présents sur un cercle centrale, et le deuxième cluster de données est un autre cercle entourant le premier cluster. Pour plus de clarté les clusters ont été repésentés ci-dessous :

Ensuite, on applique les algorithmes de k-Means et expectation maximization. On remarque que k-means va simplement produire un hyperplan séparateur des données, et par conséquent l'erreur de classification va être très importante. L'EM va quand a lui produire une classification bien plus satisfaisante. Cet exemple pathologique permet donc de montrer que la dépendance de de l'algorithme k-means à la norme L2 le met en défaut dans le cas de clusters sphériques.

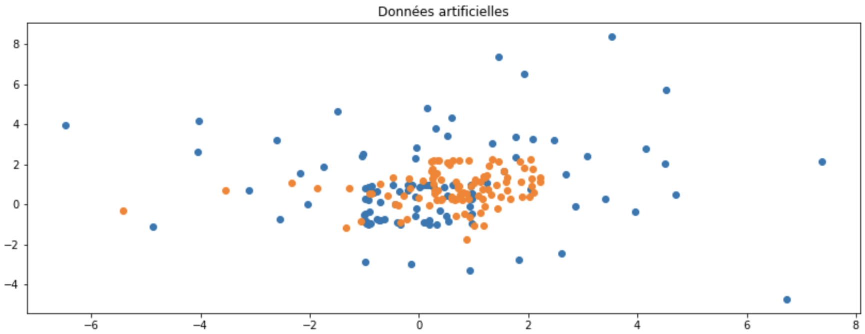

FIGURE 10 – Représentation de deux clusters concentriques

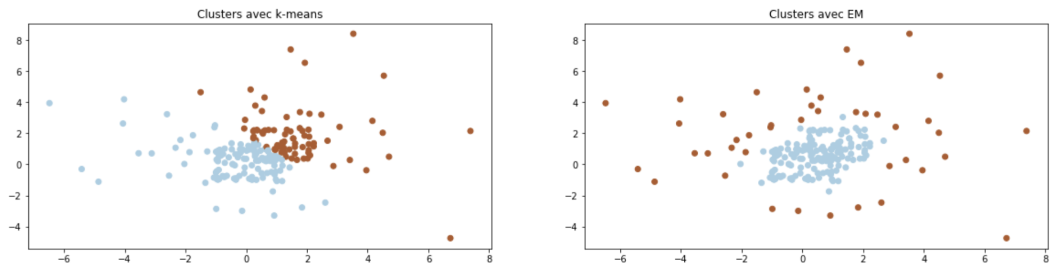

FIGURE 11 – comparaison de la performance de k-means et de EM sur les données artificielles sphériques

Graphs, algorithms and Ising

1) On implémente le "sum-product algorithm" pour une chaîne non dirigée de h noeuds dont les éléments à chaque noeud sont des vecteurs x_i, pour $i \in \{1, ..., h\}$, de taille w dont chaque élément $x_{i,j} \in \{0,1\}$, pour $j \in \{1, ..., w\}$. On représente donc l'input de la fonction qui sont les fonctions de potentiel aux noeuds ψ_i et aux bords $\psi_{i,i+1}$ de la façon suivante :

1. Pour ψ_i : une matrice N de taille $(2^w, h)$ dont chaque colonne N_i représente ψ_i. Chaque élément $N_{i,j} = \psi_i(x)$ pour $x \in \{0,1\}^w$, donc la colonne N_i contient toutes les valeurs de la fonction ψ_i appliqué à chaque élément de l'univers $\{0,1\}^w$, donc 2^w valeurs différentes.
2. Pour $\psi_{i,i+1}$: de même que pour ψ_i, un bloc de $h-1$ matrices E_k de taille $(2^w, 2^w)$ (un tenseur de taille $(2^w, 2^w, h-1)$), dont chaque matrice E_k représente $\psi_{k,k+1}$. Chaque élément $E_{i,j,k} = \psi_{k,k+1}(x, y)$ pour $x \in \{0,1\}^w$ et $y \in \{0,1\}^w$, donc la matrice E_k contient toutes les valeurs de la fonction $\psi_{k,k+1}$ appliqué à chaque élément de l'univers $\{0,1\}^w \times \{0,1\}^w$, donc $2^w \times 2^w$ valeurs différentes.

L'output de cette fonction est l'ensemble des forward $\mu_{i\to i+1}$ et backward $\mu_{i\to i-1}$ messages, représentée de même :

1. Pour les $\mu_{i\to i+1}$: une matrice F de taille $(2^w, h-1)$ dont chaque colonne F_i représente $\mu_{i\to i+1}$. Chaque élément $F_{i,j} = \mu_{i\to i+1}(x)$ pour $x \in \{0,1\}^w$, donc la colonne F_i contient toutes les valeurs de la fonction $\mu_{i\to i+1}$ appliqué à chaque élément de l'univers $\{0,1\}^w$, donc 2^w valeurs différentes.
2. Pour les $\mu_{i\to i-1}$: une matrice B de taille $(2^w, h-1)$ dont chaque colonne B_i représente $\mu_{i\to i-1}$. Chaque élément $B_{i,j} = \mu_{i\to i-1}(x)$ pour $x \in \{0,1\}^w$, donc la colonne B_i contient toutes les valeurs de la fonction $\mu_{i\to i-1}$ appliqué à chaque élément de l'univers $\{0,1\}^w$, donc 2^w valeurs différentes.

2) On se place dans le cas du modèle Ising, pour n variables aléatoires binaires $X_1, ..., X_n$, on a la loi suivante :

$$p(x_1, ..., x_n) = \frac{1}{Z(\alpha, \beta)} \exp\{\alpha \sum_i x_i + \beta \sum_{i\sim j} 1_{x_i = x_j}\}$$

Chaque variable est associée à un noeud dans une grille de taille $h \times w$. On souhaite maintenant utiliser l'implémentation du "sum-product algorithm" pour une chaîne non dirigée de la question 1, on va donc fusionner tous les noeuds de chaque ligne de la grille en un noeud comme si on regardait la grille de très loin et que l'on voyait uniquement une chaîne de taille h. Donc maintenant, on considère une chaîne de taille h dont chaque noeud X_i est un vecteur de taille w dont les éléments sont binaires : $X_{i,j} \in \{0,1\}$ pour tout $j \in \{1, ..., w\}$.

Cherchons maintenant les fonctions potentiels aux noeuds ψ_i et aux bords $\psi_{i,i+1}$.
Revenons à la loi, prenons $\alpha = 0$, et écrivons la autrement :

$$\begin{aligned}
p(x) &= \frac{1}{Z(\beta)} \prod_{i=1}^{h} \prod_{j=1}^{w-1} \exp\left(\beta 1_{x_{i,j}=x_{i,j+1}}\right) \prod_{j=1}^{w} \prod_{i=1}^{h-1} \exp\left(\beta 1_{x_{i,j}=x_{i+1,j}}\right) \\
&= \frac{1}{Z(\beta)} \prod_{i=1}^{h} \exp\left(\beta \sum_{j=1}^{w-1} 1_{x_{i,j}=x_{i,j+1}}\right) \prod_{i=1}^{h-1} \exp\left(\beta \sum_{j=1}^{w} 1_{x_{i,j}=x_{i+1,j}}\right) \\
&= \frac{1}{Z(\beta)} \prod_{i=1}^{h} \psi_i(x_i) \prod_{i=1}^{h-1} \psi_{i,i+1}(x_i, x_{i+1})
\end{aligned}$$

On obtient donc :

$$\psi_i(x_i) = \exp\left(\beta \sum_{j=1}^{w-1} 1_{x_{i,j}=x_{i,j+1}}\right) \textbf{ et } \psi_{i,i+1}(x_i, x_{i+1}) = \exp\left(\beta \sum_{j=1}^{w} 1_{x_{i,j}=x_{i+1,j}}\right)$$

avec $x_i \in \{0,1\}^w$ pour tout $i = 1, ..., h$.
On peut donc calculer les forward $\mu_{i\to i+1}$ et backward $\mu_{i\to i-1}$ messages :

$$\begin{aligned}
\mu_{1\to 2}(x_2) &= \sum_{x_1} \psi_1(x_1)\psi_{1,2}(x_1, x_2) \\
\mu_{h\to h-1}(x_{h-1}) &= \sum_{x_h} \psi_h(x_h)\psi_{h-1,h}(x_{h-1}, x_h)
\end{aligned}$$

Et pour tout $i = 2, ..., h-1$:

$$\begin{aligned}
\mu_{i\to i+1}(x_{i+1}) &= \sum_{x_i} \psi_i(x_i)\psi_{i,i+1}(x_i, x_i + 1)\mu_{i-1,i}(x_i) \\
\mu_{i\to i-1}(x_{i-1}) &= \sum_{x_i} \psi_i(x_i)\psi_{i-1,i}(x_{i-1}, x_i)\mu_{i+1,i}(x_i)
\end{aligned}$$

On calcule chaque message avec une complexité $O((h-1)2^w)$.
On obtient ainsi chaque loi marginale, pour $i = 2, ..., h-1$:

$$\begin{aligned}
p(x_1) &= \frac{1}{Z(\beta)}\psi_1(x_1)\mu_{2\to 1}(x_1) \\
p(x_i) &= \frac{1}{Z(\beta)}\mu_{i-1\to i}(x_i)\psi_i(x_i)\mu_{i+1\to i}(x_i) \\
p(x_h) &= \frac{1}{Z(\beta)}\mu_{h-1\to h}(x_h)\psi_h(x_h)
\end{aligned}$$

Et donc on peut obtenir $Z(\beta)$ en partant du noeud 1 par exemple :

$$Z(\beta) = \sum_{x_1} \psi_1(x_1)\mu_{2\to 1}(x_1)$$

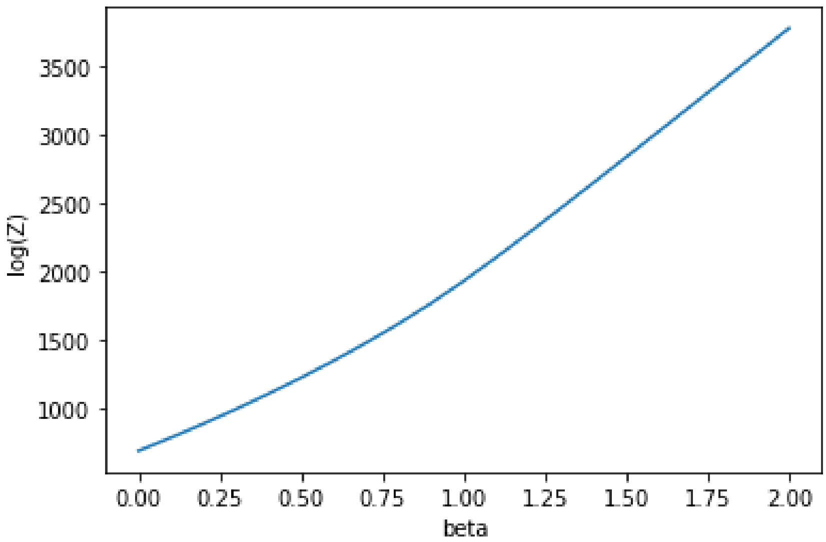

FIGURE 12 – $log(Z)$ en fonction de β pour β allant de 0 à 2

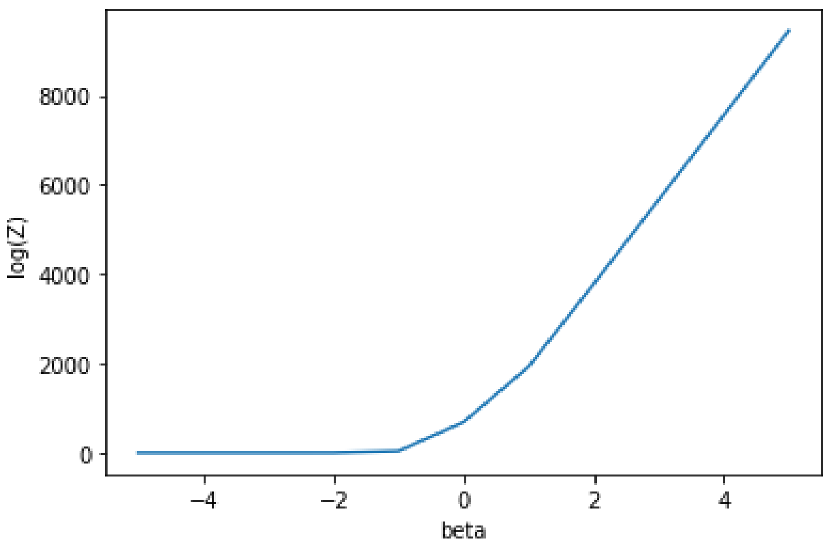

FIGURE 13 – $log(Z)$ en fonction de β pour β allant de -5 à 5