Support de Cours sur l'algorithme **EM**

École Mathématique Africaine : Bases mathématiques de l'Intelligence Artificielle

Patrice TAKAM SOH, Université de Yaoundé I

1 Motivation

L'algorithme EM est un algorithme proposé par Dempster et al. (1977). Il s'agit d'un algorithme itératif qui permet de trouver les paramètres du maximum de vraisemblance d'un modèle probabiliste lorsque ce dernier dépend de variables latentes non observables. L'algorithme EM est devenu très célèbre et très utilisé dans plusieurs domaines (économie, finance, épidémiologie,...). Cet algorithme est généralement sollicité lorsque la maximisation de la vraisemblance est très complexe mais que, sous réserve de connaître certaines données judicieusement choisies, on peut très simplement déterminer les paramètres en maximisant la vraisemblance.

2 Principe de l'algorithme EM

1. Se servir du fait qu'on sait maximiser la vraisemblance sous les observations pour maximiser la vraisemblance en intégrant les données non observées.
2. La phase **E** (Expectation): estimation des données inconnues, sachant les données observées et la valeur des paramètres determinés à l'itération précédente;
3. Phase **M** (Maximisation): maximisation de la vraisemblance, rendue désormais possible en utilisant l'estimation des données inconnues effectuées à l'étape précédente, et la mise à jour de la valeur du ou des paramètre(s) pour la prochaine itération.

3 Cadre théorique de l'algorithme EM

3.1 Notations

Nous donnons dans cette section, les différentes notations que nous allons utiliser dans la suite.

- On dispose d'observations: $Y = (Y_1, ..., Y_n)$
- On note $L(\theta|Y) := p(Y|\theta)$ la vraisemblance de Y;
- On veut maximiser $L(\theta|Y)$ ou encore $log[L(\theta|Y)]$
- On dispose de données cachées $Z = (Z_1, ..., Z_n)$
- On sait maximiser la vraisemblance complète

$$L^c(\theta|Y, Z) = log[p(Y, Z|\theta)] \tag{1}$$

Dans la suite on notera $X = (Y, Z)$ données complètes et

$$\ell(\theta|Y) := log[p(Y|\theta)] \tag{2}$$

3.2 Comment choisir le paramètre θ_{m+1} pour augmenter la vraisemblance à l'itération m?

l'idée ici est de construire une application $\theta \mapsto \delta(\theta|\theta_m)$ qu'on sait maximiser qui vérifie

$$\begin{cases} \Delta(\theta, \theta_m) & \geq \delta(\theta|\theta_m) \\ \\ \delta(\theta_m|\theta_m) & = 0 \end{cases}$$

Nous allons montrer dans la suite pourquoi une telle fonction marche!

Si on note

$$\theta' = \arg\max_{\theta} \delta(\theta, \theta_m) \tag{3}$$

alors

$$\Delta(\theta', \theta_m) \geq \delta(\theta', \theta_m) = \max_{\theta} \delta(\theta, \theta_m) \geq \delta(\theta_m, \theta_m) = 0.$$

On conclut alors que

- θ' est plus vraisemblable que θ
- il reste donc à construire la fonction δ

3.3 Construction de la fonction δ

La vraisemblance selon les "données cachées"

$$p(y|\theta) = \sum_z p(y, z|\theta) p(z|\theta) \tag{4}$$

On montre ensuite que la diffrence Δ est minore de la façon suivante

$$\begin{aligned} \Delta(\theta, \theta_m) &= log[p(y|\theta)] - log[p(y|\theta_m)] \\ \\ &= log\left[\sum_z p(y|z, \theta) p(z|\theta)\right] - log[p(y|\theta_m)] \\ \\ &\geq \sum_z p(z|y, \theta_m).log\left[\frac{p(y,z|\theta)}{p(y,z|\theta_m)}\right] \end{aligned}$$

3.4 Expression de la fonction δ

En posant

$$\delta(\theta, \theta_m) = \sum_z p(z|y, \theta_m).log\left[\frac{p(y, z|\theta)}{p(y, z|\theta_m)}\right], \text{ on a}$$

- $\Delta(\theta, \theta_m) \geq \delta(\theta, \theta_m)$ par construction
- $\delta(\theta_m, \theta_m) = 0$ d'après son expression.

Avec cette expression de δ, on en déduit la paramètre à l'itration $m+1$ par

$$\begin{aligned}
\theta_{m+1} &= \arg\max_{\theta}(\delta(\theta, \theta_m)) \\
&= \arg\max_{\theta}\left(\sum_z p(z|y,\theta_m).\log\left[\tfrac{p(y,z|\theta)}{p(y,z|\theta_m)}\right]\right) \\
&= \arg\max_{\theta}\left(\sum_z p(z|y,\theta_m).\log p(y,z|\theta)]\right) \\
&= \arg\max_{\theta}(\mathbb{E}_{z|y,\theta_m}[\log p(y,z|\theta)])
\end{aligned}$$

3.5 Résumé sur les deux étapes de l'EM

- On démarre l'algorithme avec une ignorance absolue des données cachées z et en initialisant θ à une valeur θ_0
- L'algorithme se sert de θ_0 pour estimer z, puis se sert de $\hat{z}$ pour réestimer les paramètres en une valeurs θ_1 plus pertinente
- A l'itération suivante, on améliore donc l'estimation des données cachées z puisque cette nouvelle estimation se base cette fois sur $\hat{z}$ conduit à son tour à une meilleure précision sur θ_2, etc.

4 Exemples d'application

4.1 Modèle de Mélange gaussiens

On dit que f est une densité de mélange s'il existe $k \in \mathbb{N}$, des densités $f_1, ..., f_k$ et des réels $p_1, ..., p_k$ sommant à 1 tels que

$$f(x) = \sum_{i=1}^{k} 1_{\{Z=i\}}\mathbb{P}(Z=i)f_i(x) = \sum_{i=1}^{k} 1_{\{Z=i\}}p_i f_i(x) \tag{5}$$

avec

$$Z(\Omega) = \{1, ..., k\} \text{ et } \mathbb{P}(Z=i) = p_i \tag{6}$$

Dans le cas bidimensionel, la vraisemblance complète est donné par

$$L^c(\theta|\underline{x}, z) = \prod_{i=1}^{n}\left[\prod_{j=1}^{2}(\lambda_j f_j(x_i))^{1_{\{Z_i=j\}}}\right] \tag{7}$$

et la **Log-Vraisemblance complète** est donnée par

$$\log L^c(\theta|\underline{x}, z) = \sum_{i=1}^{n}\left[\sum_{j=1}^{2} 1_{\{Z_i=j\}}\log(\lambda_j) - \log(2\pi) - \frac{1}{2}\log(\det(\Sigma_j))\right.$$

$$\left. -\frac{1}{2}(x-\mu_j)^T\Sigma_j^{-1}(x-\mu_j)\right]$$

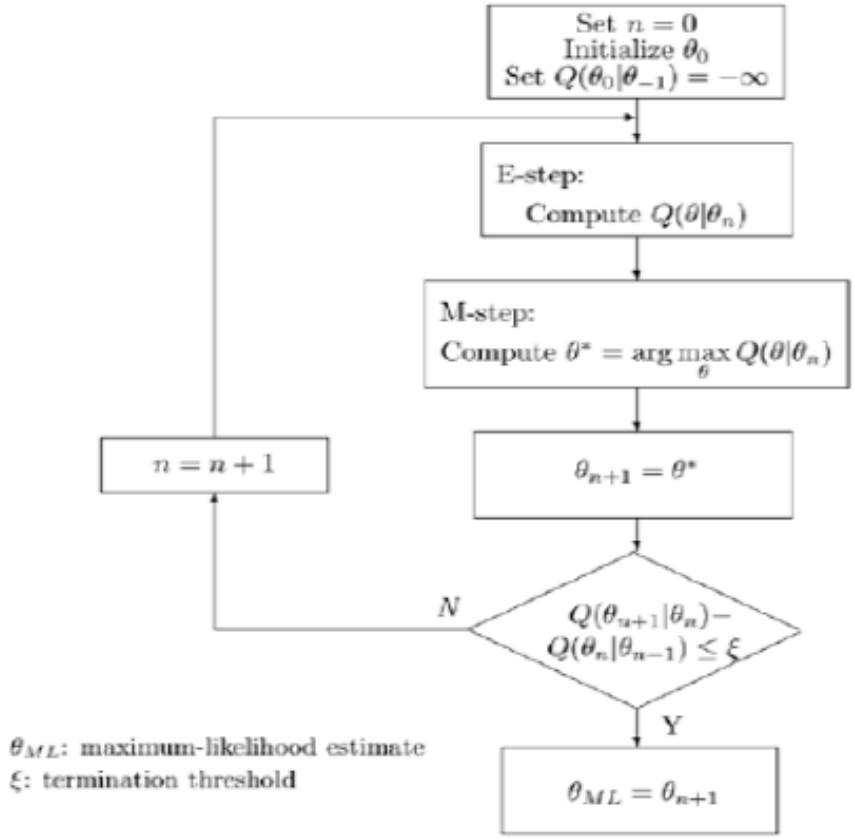

Figure 1: Schma de l'algorithme EM

A l'étape E, l'esprance conditionne donne

$$\begin{aligned} Q(\theta,\theta^s) &= \mathbb{E}_{z|y,\theta^{(s)}}[\log L^c(\theta|X,Z)] \\ &= \sum_{i=1}^{n}\left[\sum_{j=1}^{2}\tilde{p}_{i,j}\log(\lambda_j) - \log(2\pi) - \frac{1}{2}\log(\det(\Sigma_j))\right. \\ &\left. -\frac{1}{2}(x-\mu_j)^T\Sigma^{-1}(x-\mu_j)\right] \end{aligned}$$

avec

$$\begin{aligned} \tilde{p}_{i,j} &= \mathbb{E}\left(1_{\{Z_i=j\}} \mid y,\theta^{(s)}\right) = \mathbb{P}\left(Z_i=j|y,\theta^{(s)}\right) \\ &= \frac{\lambda_j f_j(x_i)}{\lambda_1 f_1(x_i)+\lambda_2 f_2(x_i)} \end{aligned}$$

4.2 Algorithme EM dans les données censurées à droite

Une étude consiste à étudier l'effet d'un médicament sur le temps de guérison. Pour cela, on a donné le médicament à 21 patients et on a noté le temps de guérison. On a obtenu les valeurs suivantes:

6	6	6	6+	7	9+	10
10+	11+	13	16+	17+	19+	20+
22	23	25+	32+	32+	34+	35+

On observe que parmi ces $n = 21$ observations, on a 13 qui sont censurées et $m = 9$ qui ne sont pas censurées. On suppose que la durée suit une loi exponentielle de paramètre σ (qu'on veut estimer à partir des observations).

La Vraisemblance complète est donnée par

$$L^c(\sigma|y,z) = \prod_{i=1}^{m} f(y_i) \prod_{i=m+1}^{n} f(z_i) \text{ où } f(x) = \sigma\exp(-\sigma x)$$

et la Log-vraisemblance complète:

$$\log L^c(\sigma|y,z) = n\log(\sigma) - \sigma\sum_{i=1}^{m} y_i - \sigma\sum_{i=m+1}^{n} z_i$$

A l'étape E, l'espérance conditionnelle est donnée par

$$Q(\sigma,\sigma_m) = \int \log L^c(\sigma|y,z)p(z|y,\sigma_m)dz, \text{ avec } \sigma^{(m)} = \sigma_m$$

Il est alors de calculer la loi des inconnues sachant les observations et le paramètre de l'étape courante

4.3 Loi de z sachant les donnés et le paramètre à l'étape courante

Il est question de calcul de $p(z|y,\sigma_m)$. Pour cela, on a

$$p(z|y,\theta_m) = \prod_{i=m+1}^{n} p_{z_i}(z_i|y,\theta_m)$$

On montre que

$$p(z|y,\sigma_m) = \prod_{i=m+1}^{n} \sigma_m \exp(-\sigma_m(z_i - R_i))$$

A l'étape E, on obtient alors Calcul de $Q(\sigma, \sigma_m)$

$$\begin{aligned} Q(\theta,\theta_m) &= n\log(\sigma) - \sigma\sum_{i=1}^{m} y_i \\ &\quad -\sigma\sum_{i=m+1}^{n}\int z_i \prod_{j=m+1}^{n} \sigma_m \exp\left(-\sigma_m(z_j - R_j)\right) dz, \end{aligned}$$

A l'étape M Etape on maximise $Q(\sigma, \sigma_m)$ en σ,

$$\hat{\sigma} = \frac{n}{\frac{(n-m)}{\sigma_m} + \sum_{i=1}^{m} y_i + \sum_{i=m+1}^{n} R_i}$$

Et La relation entre σ_m et σ_{m+1} est alors:

$$\sigma_{m+1} = \frac{n}{\frac{(n-m)}{\sigma_m} + \sum_{i=1}^{m} y_i + \sum_{i=m+1}^{n} R_i} \tag{8}$$

5 Algorithme MCEM

Il s'agit de la variante de l'EM la plus utilisé qui est utilisé lorsqu'on est incapable de calculer l'espérance conditionnelle à l'étape E. Ainsi, à l'étape **E**, on calcule

$$\hat{Q}(\theta,\theta') = \frac{1}{K}\sum_{k=1}^{K} \log L^c(\theta|y, z^{(k)})$$

où les $z^{(k)}$ sont obtenues selon la loi $p(z|y,\theta_m)$ et la principale difficulté reste de savoir **comment simuler les valeurs selon** $p(z|y,\theta_m)$.

C'est qui est à l'origine des approches de type **MCEM**

5.1 Pourquoi simuler les v.a

- générer une variable aléatoire X ayant pour distribution la fonction de masse π c'est à dire $p(X = x_i) = \pi_i$
- Pour approximer les quantités sous forme d'espérance $\mathbb{E}(h(X))$ où h est une fonction mesurable, X une v.a. En effet d'après la loi forte des grands nombres

$$\frac{1}{n}\sum_{i=1}^{n} h(X_i) \xrightarrow{p.s} \mathbb{E}(h(X))$$

- pour approximer les quantités sous forme de probabilité par la relation

$$\mathbb{P}(A) = \mathbb{E}(1_A)$$

- pour approximer les quantités sous forme d'intégrale (Monte-Carlo)

$$\int_a^b f(x)dx = (b-a)\mathbb{E}(f(U)) \text{ avec} U \sim \mathcal{U}([a,b])$$

5.2 Méthodes de simulation des variables aléatoires

Dans cette partie,

- On suppose qu'on dispose d'un générateur de variable aléatoire de loi uniforme sur $[0;1]$ indépendantes : $U \sim \mathcal{U}([0,1])$
- On note F_X la fonction de répartion de la v.a. X: $F_X(x) = \mathbb{P}(X \leq x)$
- On rappelle que la fonction de répartition de U est donnée par

$$F_U(u) = \begin{cases} 0 & \text{si } u < 0 \\ u & \text{si } 0 \leq u \leq 1 \\ 1 & \text{si } u > 1 \end{cases}$$

- La fonction densité de u est donnée par $f_U(u) = 1_{[0,1]}(u)$
- On rappelle que pour générer u selon une uniforme dans $[0,1]$ on utilise
  - **runif()** sous R

5.2.1 Loi de Bernoulli et loi binomiale

- **blue**Loi de bernouilli
  - On veut simuler $X \sim \mathcal{B}(p)$
  - Si on tire $u \sim \mathcal{U}([0,1])$
    - si $u \leq p$ alors $x = 1$
    - sinon $x = 0$
  - On peut donc écrire $X = 1_{\{U \leq p\}}$ et il est immédiat que $X \sim \mathcal{B}(p)$
- **blue**Loi binomiale
  - On sait que $\{Y_1, ..., Y_n\}$ i.i.d., $Y_i \sim \mathcal{B}(p) \Longrightarrow X = \sum_{i=1}^{n} Y_i \sim \mathcal{B}(n,p)$
  - Il suffit donc de simuler n variables aléatoires indépendantes de loi $\mathcal{B}(p)$ et d'en faire la somme, soit

$$X = \sum_{i=1}^{n} 1_{\{U_i \leq p\}}$$

5.3 Loi de probabilité discrète sur un ensemble fini ou infini

1. Cas où $card(X(\Omega)) < \infty$
   - Soit X une v.a. telle que $X(\Omega) = \{x_1, ..., x_K\}$ avec $p_k = \mathbb{P}(X = x_k)$
   - On pose $P_k = \sum_{i=1}^{K} p_i$ (avec $P_1 = 0$ et $P_K = 1$).

- On tire ensuite $u \sim \mathcal{U}([0,1])$
- si $p_{k-1} \leq u \leq p_k$ alors $x = x_k$
- On peut alors écrire que $X = \sum_{k=1}^{K} x_k 1_{\{p_{k-1} \leq U \leq p_k\}}$

2. Cas où $card(X(\Omega)) = \infty$

   - On reprend la même idée
   - Pour les conventions sur P_k on va prendre comme convention $P_{-1} = 0$ et la somme précédente devient $X = \sum_{k \geq 0} x_k 1_{\{p_{k-1} \leq U \leq p_k\}}$

5.4 Variables aléatoires continues

Soit F la f.r. d'une v.a. X

Definition 1. *L'inverse généralisée de F est définie par $F^{-1} :]0,1] \longrightarrow \mathbb{R} \cup \{+\infty\}$ avec $F^{-1}(y) = \inf\{x : F(x) \geq y\}$*

On montre que $\forall x \in \mathbb{R}$ et $y \in]0,1[$ on a l'équivalence

$$y \leq F(x) \Longleftrightarrow F^{-1}(y) \leq x$$

Proposition 1. *Soit X une v.a. de fonction de répartition F*

- *Si F est strictement croissante, on a $F(X) \sim \mathcal{U}([0,1])$*
  $\Longrightarrow$ Si $U \sim \mathcal{U}([0,1])$ alors $F^{-1}(U)$ a même loi que X.
- *Si F n'est pas bijective on a le résultat analogue avec l'inverse généralisée de F.*

5.5 Acceptation-rejet

On veut simuler une variable aléatoire X de densité f et de fonction de répartition F.

- on ne sait pas simuler directement la loi de densité de probabilité f
- on sait simuler selon une autre loi de densité de probabilité g telle que

$$\frac{f}{g} \text{ est bornée ou encore } \exists c \text{ telle que } f \leq c.g$$

Proposition 2. *Considérons une variable aléatoire réelle X ayant g pour densité de probabilité et Y une autre variable aléatoire réelle indépendante de X et suivant la loi uniforme standard $\mathcal{U}([0,1])$. On établit alors que la loi conditionnelle de X sachant $Y < f(X)/c.g(X)$ a pour densité f.*

5.6 Algorithme de type MCMC

5.6.1 Définitions autour d'une chaîne de Markov

Definition 2.
- *Un processus stochastique $\{X_t\}$ est une **chaîne de Markov** homogène de loi initiale λ et de **matrice de transition** $P = (p_{i,j})$ ssi $\forall n \geq 0$ et $i_o, ..., i_{n+1} \in I$, on a*

  (i) $\mathbb{P}(X_0 = i_0) = \lambda_{i_0}$

  (ii) $\mathbb{P}\{X_{i_n+1} = i_{n+1} \mid X_{i_n} = i_n, ..., X_0 = i_0\} \equiv p_{i_n, i_{n+1}}$

- $\{X_t\}_{t\geq 0}$ *est* ***irréductible*** *si, pour toutes les paires* (i,j)*, la probabilité de passer de l'état* i *á l'état* j *est strictement positive*

- π_j^* *est dite* ***stationnaire*** *ssi*

$$\mathbb{P}(X_0 = j) = \pi_j^* \ \ \forall j = 1,2,...,m \Longrightarrow \mathbb{P}(X_t = j) = \pi_j^* \ \ \forall t = 1,2,...$$

Proposition 3. *Les probabilités* π_j^* *sont déterminées*

(i) soit par la relation $\pi_j^* = \sum_{i=1}^{m} p_{i,j}\pi_i^*$

(ii) si on suppose qu'il existe des nombres positifs $x_1,...,x_m$ *satisfaisant* $x_i q_{i,j} = x_j q_{j,i}$ *et* $\sum_{i=1}^{m} x_i = 1$ *alors* $\pi_j = x_j$

Proposition 4. *Pour toute CM* $(X_t)_{t\geq 0}$ *admettant des probabilités stationnaires. Pour toute fonction* $h : \{1,...,m\} \longrightarrow \mathbb{R}$*,*

$$\lim_{T\longrightarrow\infty} \frac{1}{T}\sum_{i=1}^{m} h(X_t) = \sum_{i=1}^{m} h(i)\pi_i \ p.s$$

5.6.2 Algorithme de Metropolis Hastings

1. Etant donné $\theta^{(t)}$
2. Générer $y_t \sim q(y \mid \theta^{(t)})$
3. Aceptation-rejet

$$\theta^{(t+1)} = \begin{cases} y_t & \text{avec probabilité } \rho(\theta^{(t)}, y_t) \\ \theta^{(t)} & \text{avec probabilité } 1 - \rho(\theta^{(t)}, y_t) \end{cases} \tag{9}$$

où

$$\rho(\theta, y) = \min\left(\frac{f(y)q(\theta \mid y)}{f(\theta)q(y \mid \theta)}, 1\right)$$

Nb: On notera $q(\theta, y) = q(y \mid \theta)$ et $(\rho(\theta, y))_{\theta,y}$ les termes génériques des matrice correspondantes.

La version standard qui se trouve dans tous les logiciels de simulations des chanes de markov est la suivante

- **Etape 0**: Choisir $X_0 \in \{1,2,...,m\}$

- **Etape t+1**

1. Générer une v.a. X de fonction de masse q

2. Générer une v.a U de loi uniforme sur $\mathcal{U}([0,1])$

3. Si $U < \alpha_{X_t,X}$ alors $X_{t+1} = X$

4. Si $U \geq \alpha_{X_t,X}$ alors $X_{t+1} = X_t$

On peut faire les remarques suivantes sur cet algorithme

- le choix de la loi de proposition est critique
- le choix du point initial est critique
- le temps de chauffage peut être très long
- l'échantillon obtenu est non indépendant

5.6.3 Algorithme de Gibbs sample

Notation

- Si on pose $Z = (Z_1, ..., Z_n)$ est un vecteur aléatoire dont la fonction de masse (densité dans le cas continu) conjointe est $p(z)$.
- On souhaite générer un vecteur aléatoire dont la distribution est conditionnelle à l'**appartenance à l'ensemble** A, c'est-à-dire que la loi conjointe est

$$f(z) = \frac{p(z)}{\mathbb{P}(Z \in A)}$$

- Il faut faire **l'hypothèse** que pour n'importe laquelle des n variables, nous pouvons générer des valeurs à partir de

$$f_i(z) = P(Z_i = z_i \mid Z_j = z_j, j \neq i).$$

6 Algorithme de l'échantillonneur de Gibbs

- $X_t = (z_1, ..., z_i, ..., z_n)$
- $U \sim \mathcal{U}([0,1])$, $C = [nU] + 1$
- Si $C = i$, alors $X = x$ est généré selon la loi f_i.
- Si $y = (z_1, ..., z_i, y, z_{i+1}, ..., z_n) \in A$, $X_{t+1} = x$, Sinon, $X_{t+1} = X_t$.

0. Étant donné $\theta^{(t)} = (\theta_1^{(t)}, ..., \theta_p^{(t)})$
1. Générer $\theta_1^{(1)} \sim f_1(\theta_1 \mid \theta_2^{(t)}, ..., \theta_p^{(t)})$

2. Générer $\theta_2^{(1)} \sim f_2(\theta_2 \mid \theta_1^{(t)}, \theta_3^{(t)}, ..., \theta_p^{(t)})$

   $\vdots$

p. Générer $\theta_p^{(1)} \sim f_2(\theta_2 \mid \theta_1^{(t)}, \theta_3^{(t)}, ..., \theta_p^{(t)})$

NB: Seules les lois conditionnelles $f_1, ..., f_p$ sont utilisées pour la simulation. Donc, même pour un problème de grande dimension, toutes les simulations sont univariées!.

Enfin on peut faire les remarques suivantes sur l'chantillon de Gibbs

- Taux d'acceptation uniformément égal á 1. Critères sur taux d'acceptation optimaux non valables. Convergence à établir suivant d'autres critères.
- Limitations fortes sur le choix des paramètres des lois instrumentales.
- Connaissance préalable de certaines propriétés (probabilistes ou analytiques) de la loi cible.
- Nécessairement multidimensionnel.
- Ne fonctionne pas lorsque le nombre de variables est variable.