Pierre Alquier

Analyse bayésienne de modèles de mélanges: Approche par MCMC à sauts réversibles

Mémoire de Dea Paris 6 / GT Ensae

Encadré par Mme Dominique Picard

Remis le 2 juin 2003

Table des matières

1 Introduction : l'inférence bayésienne dans les modèles de mélanges

1.1 Les modèles de mélanges

1.1.1 Définition

Les modèles de mélanges ont été initialement introduits afin de représenter l'existence de différents groupes relativement homogènes dans une population donnée.

Supposons que l'on étudie une quantité d'intérêt, x, dans une population composée en fait de k groupes relativement homogènes, en proportions $p_1, ..., p_k$. Dans chaque groupe, la quantité d'intérêt est supposée suivre une loi de densité $f_{\theta_i}(x)$. Ainsi, la quantité x sera répartie sur l'ensemble de la population suivant une loi de densité :

$$g(x) = \sum_{i=1}^{k} p_i f_{\theta_i}(x)$$

On appelle "mélange fini de distributions" une telle densité. Les différents $f_{\theta_i}(x)$ sont appelés les "composantes du mélange".

La figure 1 représente la densité d'un mélange de deux lois normales.

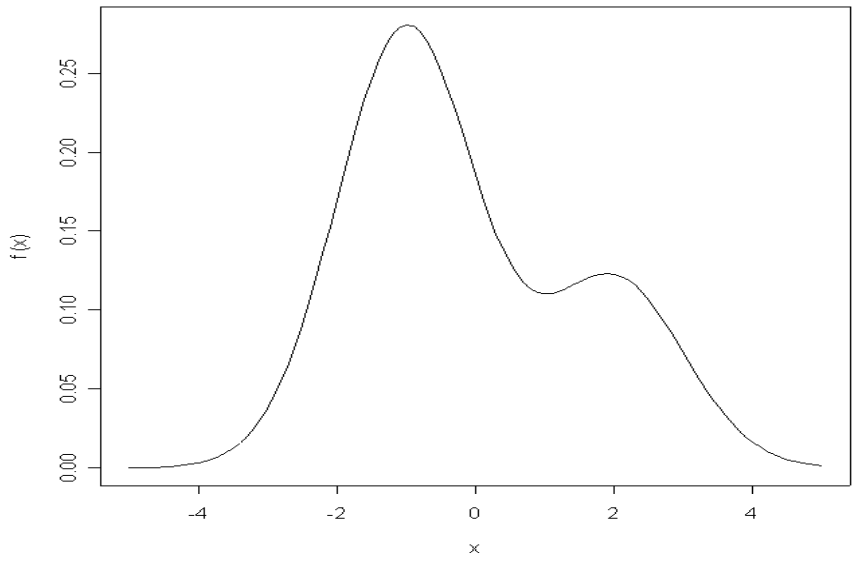

FIG. 1 – Mélange de deux lois normales de moyennes distinctes :"$0.7\mathcal{N}(-1,1) + 0.3\mathcal{N}(2,1)$"

1.1.2 Commentaires

D'un point de vue pratique, les paramètres θ_i et p_i sont presque toujours inconnus. Le but de la démarche statistique sera donc de proposer une méthode d'estimation de ces paramètres, ce que nous allons examiner d'un point de vue bayésien dans la fin de cette première partie. Il est un certain nombre de cas où la valeur de k elle-même sera inconnue. Le but de ce mémoire est de présenter un certain nombre de méthodes, issues de la statistique bayésienne, permettant de considérer k comme un paramètre du modèle et de l'estimer. Ceci est l'objet de la méthode présentée dans ce mémoire.

Remarquons tout d'abord que le modèle, présenté tel quel, n'est pas identifiable. En effet, considérons un mélange simple à deux composantes. On a évidemment :

$$pf_{\theta_1} + (1-p)f_{\theta_2} = (1-p)f_{\theta_2} + pf_{\theta_1}$$

Ainsi les couples (p, θ_1, θ_2) et $(1-p, \theta_2, \theta_1)$ définissent la même loi. Il sera donc nécessaire d'imposer certaines contraintes supplémentaires afin d'assurer l'identifiabilité du modèle, par exemple une contrainte d'ordre sur les poids : $p > 1-p$ ou plus généralement $p_1 > ... > p_k$.

Notons que les modèles de mélanges peuvent être utilisés dans diverses applications dépassant le cadre présenté précédemment, par exemple :

- dans un problème quelconque d'inférence statistique ou le recours à une famille paramétrique simple n'est pas satisfaisant, l'introduction de mélanges de distributions permet d'enrichir le modèle considéré. Dans ce cas, les différentes composantes ne s'interprètent plus comme reflétant la présence de groupes dans la population étudiée. Cf. Robert (1996).

- les modèles de mélange sont aussi vus comme un compromis entre certaines méthodes paramétriques et non-paramétriques. Par exemple, pour estimer la densité de la loi d'un échantillon $X_1, ..., X_n$ de variables indépendantes, une approche paramétrique simple consisterait à utiliser une loi normale $\mathcal{N}(\mu, \sigma^2)$ alors qu'une approche non-paramétrique possible est l'utilisation de l'estimateur du noyau :

$$\widehat{f(x)} = \frac{1}{nh_n} \sum_{i=1}^{n} \varphi(\frac{x - X_i}{h_n})$$

où φ est la densité d'une loi $\mathcal{N}(0,1)$. Ainsi, on approche la loi des observations par des lois normales, dans un cas en imposant la présence d'une seule normale, dans l'autre cas en imposant la présence de n normales. Ainsi, le modèle de mélanges peut être vu comme un compromis entre les deux méthodes en ce qu'il utilisera k normales où $1 \leq k \leq n$, où k peut être soit fixé par le statisticien, soit estimé comme nous le verrons dans la seconde partie. Cf. Robert (1996) ou McLachlan & Peel (2000). L'article de Roeder et Wasserman (1997) utilise effectivement des modèles de mélanges pour faire de l'estimation de densité non paramétrique.

- l'utilisation des modèles de mélanges est particulièrement intéressante pour le problème de la classification. On peut supposer qu'il existe effectivement k classes dans la population étudiée, mais ne pas savoir de quel groupe est issu un individu particulier. Si l'on souhaite "deviner" quel est le groupe (ou la classe) d'un individu donné à partir de l'observation de x et que l'on adopte le modèle de mélanges :

$$g(x) = \sum_{i=1}^{k} p_i f_{\theta_i}(x)$$

alors un individu pour lequel $x = X$ est classé de façon optimale dans la classe $i^* = \arg\max_{1 \leq i \leq k} p_i f_{\theta_i}(X)$. Donc, estimer les paramètres du modèle permet de proposer une classification des individus. De plus, l'estimation de k apporte aux méthodes plus classiques une estimation du nombre possible de classes dans la population.

- enfin, les modèles de mélanges sont utilisés lorsqu'on est en présence d'observations "aberrantes". Supposons par exemple que l'on étudie le modèle $\mathcal{N}(\theta, 1)$ (avec θ proche de 0) et que l'on ait une observation aberrante, valant 10, dans l'échantillon. On peut alors utiliser le mélange "$p_1\mathcal{N}(\theta, 1) + (1 - p_1)\mathcal{N}(0, 100)$" pour éviter que l'observation aberrante ne vienne entacher d'erreur l'estimation de θ. La figure 2 illustre ceci.

1.1.3 Première approche pour l'estimation par méthodes bayésiennes

L'utilisation de méthodes bayésiennes pour estimer les paramètres d'un mélanges de distributions se justifie tout d'abord par l'impossibilité, au moins en première approche, d'utiliser les méthodes classiques comme le maximum de vraisemblance : en général, la vraisemblance d'un modèle de mélanges n'est pas bornée...

D'un point de vue bayésien, la densité du modèle de mélange présenté précédemment (avec k fixé) devient :

$$g(x|\xi) = \sum_{i=1}^{k} p_i f(x|\theta_i)$$

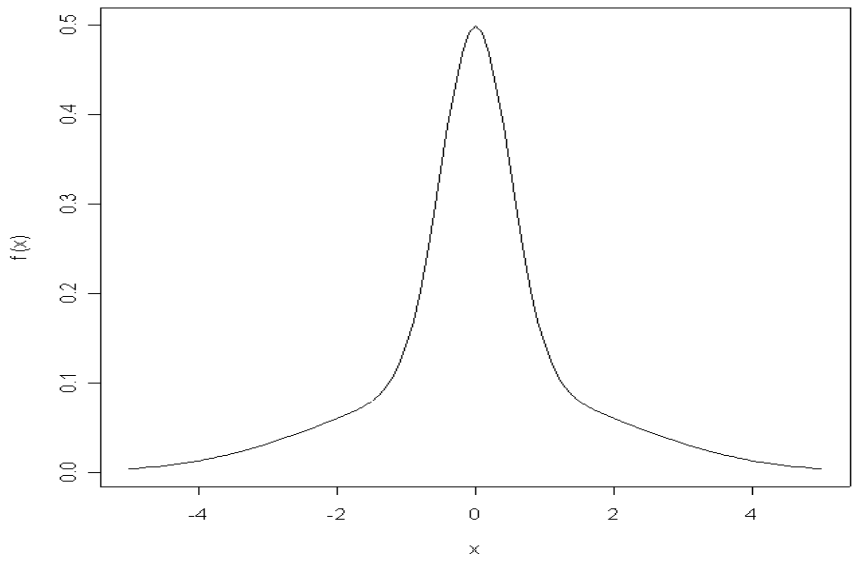

FIG. 2 – Mélange de deux lois normales de même moyenne mais de variance différentes ("chapeau mexicain") : "$0.5\mathcal{N}(0, 0.5) + 0.5\mathcal{N}(0, 2)$"

soit une vraisemblance, pour n observations $x_1, ..., x_n$:

$$\prod_{j=1}^{n} g(x_j|\xi) = \prod_{j=1}^{n} \sum_{i=1}^{k} p_i f(x_j|\theta_i)$$

où $\xi = (\theta_1, ..., \theta_k, p_1, ..., p_k)$. On spécifie aussi une loi *a priori* sur le paramètre ξ, $\pi(\xi)$. Remarquons qu'il est en général impossible de spécifier une loi *a priori* impropre sur le paramètre ξ (si on le fait, la loi *a posteriori* peut elle aussi ne pas être une mesure de probabilités !). Ceci interdit l'utilisation des lois non-informatives usuelles (loi de Jeffreys). Le choix des lois *a priori* sera donc crucial dans un modèle de mélange.

Dès lors, la loi *posteriori* est donnée par :

$$\pi(\xi|x_1, ..., x_n) = \frac{\pi(\xi) \prod_{j=1}^{n} \sum_{i=1}^{k} p_i f(x_j|\theta_i)}{\int \pi(\xi) \prod_{j=1}^{n} \sum_{i=1}^{k} p_i f(x_j|\theta_i) d\xi}$$

le principal problème pour la mise en pratique des méthodes bayésiennes sur des modèles de mélanges finis apparaît alors : le calcul de la loi *a posteriori* est souvent impossible sous forme exacte (notamment l'intégrale au dénominateur), ce qui interdit aussi le calcul explicite des estimateurs bayésiens, de plus les approximations numériques sont souvent trop longues en pratiques (présence de k^n termes au numérateur et au dénominateur).

En réalité, les problèmes posés ci-dessus n'ont pu être résolus qu'avec le développement des méthodes de Monte-Carlo par Chaînes de Markov (MCMC).

1.2 Les méthodes MCMC

1.2.1 Principe général

Le but des algorithmes MCMC est de simuler une chaîne de Markov de loi de probabilité stationnaire donnée de densité f, notée par exemple $(x_n)_{n\in\mathbb{N}}$. Si la chaîne est simulée de façon à satisfaire les hypothèses voulues (irréductibilité, apériodicité) on pourra appliquer le théorème ergodique.

Par exemple, pour g avec $\mathbb{E}_f|g(X)| < \infty$ on a :

$$\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} g(x_n) = \mathbb{E}_f(g(X))$$

Intuitivement, on souhaite donc que (x_n) "visite toutes les zones probables" sous f. Il faudra donc simuler x_n jusqu'à $n = n_2$ où "n_2 est grand". De plus, le choix de x_0 va avoir une influence importante sur les premières valeurs de la suite (x_n) produite. En particulier, si x_0 est située dans une zone "peu probable" selon f, il en sera probablement de même pour les premières valeurs x_1, x_2,... Donc il conviendra d'enlever les n_1 premières réalisations de (x_n). n_1 est appelé *burn-in period*. Ainsi, on estimera $\mathbb{E}_f(g(X))$ par :

$$\frac{1}{n}\sum_{i=n_1}^{n_2} g(x_n)$$

En pratique, le choix de n_1 et n_2 est basé sur les observations de (x_n) et de la convergence des quantités étudiées.

1.2.2 Algorithme de Hastings-Metropolis

Etant donné une loi de densité f par rapport à une mesure dominante μ, l'algorithme de Hastings a pour but de simuler la réalisation d'une chaîne de Markov ergodique de loi stationnaire de densité f, et ceci même si f n'est connue qu'à une constante multiplicative près. Ceci pourrait notamment être le cas de la loi *a posteriori* dans le cas de l'inférence bayésienne dans les modèles de mélanges présentée précédemment :

$$f(\xi) = \pi(\xi|x_1,...,x_n) = \frac{1}{C}\pi(\xi)\prod_{j=1}^{n}\sum_{i=1}^{k} p_i f(x_j|\theta_i)$$

Pour ceci, il est nécessaire de disposer d'une probabilité de transition Q de densité $q(x|y)$ par rapport à μ, vérifiant :

- on peut simuler rapidement une variable aléatoire de loi $q(.|y)$,
- soit q est connue sous forme analytique, soit q est symétrique, c'est-à-dire que $\forall x, \forall y, q(x|y) = q(y|x)$.

L'algorithme est alors le suivant. On se donne x_0 arbitraire et on passe de x_n à x_{n+1} ainsi :

- on simule une variable x'_n selon la loi $q(.|x_n)$,
- on calcule la probabilité d'acceptation

$$a(x_n, x'_n) = \min\left(1, \frac{f(x'_n)}{f(x_n)}\frac{q(x_n|x'_n)}{q(x'_n|x_n)}\right)$$

puis avec une probabilité $a(x_n, x'_n)$ on "accepte" x'_n c'est-à-dire que l'on prend $x_{n+1} = x'_n$, et avec une probabilité $1 - a(x_n, x'_n)$ on rejette x'_n, on prend alors $x_{n+1} = x_n$. Ceci est réalisé le plus simplement possible en simulant la réalisation d'une variable uniforme sur $[0, 1]$, u_n, puis en comparant u_n avec $a(x_n, x'_n)$: acceptation si $u_n < a(x_n, x'_n)$, rejet sinon.

Plusieurs remarques sont alors à faire. Tout d'abord, il faut vérifier si l'algorithme décrit ci-dessus est bien un "algorithme MCMC", c'est-à-dire si il produit bien une chaîne de Markov de probabilité stationnaire de densité f. Si c'est le cas, les x_n produits nous serviront ensuite à étudier les propriétés de f à l'aide de théorèmes limites sur les chaînes de Markov, comme le théorème ergodique.

En fait, il est évident que (x_n) est une chaîne de Markov, de probabilité de transition $P(x, y) = a(x, y)q(y|x) + (1 - a(x, y))\delta_x(y)$. Ceci permet de démontrer que f est bien une densité invariante pour (x_n), en utilisant le fait que f satisfait la *detailed balance condition* :

$$f(y)P(y,x) = f(x)P(x,y)$$

Ceci se vérifie facilement :

$$f(y)P(y,x) = f(y)\left(\min(1, \frac{f(x)}{f(y)}\frac{q(y|x)}{q(x|y)})q(x|y) + (1-a(x,y))\delta_x(y)\right)$$

$$... = \min(f(y)q(x|y), f(x)q(y|x)) + f(y)(1-a(x,y))\mathbb{I}(y=x)$$

$$... = f(x)\min(\frac{f(y)}{f(x)}\frac{q(x|y)}{q(y|x)}, 1)q(y|x) + f(x)(1-a(y,x))\delta_y(x)$$

$$\Rightarrow f(y)P(y,x) = f(x)P(x,y)$$

Démontrons donc maintenant que f est invariante. Il suffit pour cela de démontrer que, quel que soit A ensemble mesurable :

$$\int P(x,A)f(x)dx = \int_A f(x)dx$$

Or :

$$\int P(x,A)f(x)dx = \int\int_A P(x,y)f(x)dydx = \int\int \mathbb{I}_A(y)P(x,y)f(x)dydx$$

On utilise alors la *detailed balance condition* $f(y)P(y,x) = f(x)P(x,y)$ et on obtient :

$$\int P(x,A)f(x)dx = \int\int \mathbb{I}_A(y)P(y,x)f(y)dydx$$

Toutes les quantités dans l'intégrale double étant positives on peut appliquer le théorème de Fubini :

$$\int P(x,A)f(x)dx = \int \mathbb{I}_A(y)f(y)\left(\int P(y,x)dx\right)dy$$

Or $P(.,.)$ étant une probabilité de transition on a $\forall y, \int P(y,x)dx = 1$ donc :

$$\int P(x,A)f(x)dx = \int_A f(y)dy$$

ce qui est exactement ce qu'il fallait démontrer.

Enfin, si $\forall x, \forall y, q(x,y) > 0$ la chaîne de Markov (x_n) est irréductible. Elle est apériodique car il est toujours possible en général que $x_{n+1} = x_n$. Elle est récurrente positive car f est la densité d'une loi de probabilités. On est donc dans les conditions d'applications du théorème ergodique.

1.2.3 Algorithme de Gibbs

Un autre algorithme, dit algorithme de Gibbs, permet de simuler une chaîne de Markov ergodique de loi stationnaire $\nu(\theta_1, ..., \theta_k)$ donnée, à condition de savoir simuler toutes les lois de la forme : $\nu(\theta_i|\theta_1, ..., \theta_{i-1}, \theta_{i+1}, ..., \theta_k)$. Comme nous le verrons rapidement, ce cadre est particulièrement adapté à celui de l'inférence bayésienne.

L'algorithme est le suivant :

- on se donne $(\theta_1^0, ..., \theta_k^0)$ artibraire.
- à l'étape n, on dispose donc de $(\theta_1^n, ..., \theta_k^n)$. On simule alors :

$$\theta_1^{n+1} \sim \nu(\theta_1|\theta_2^n, ..., \theta_k^n)$$

$$\theta_2^{n+1} \sim \nu(\theta_2|\theta_1^{n+1}, \theta_3^n, ..., \theta_k^n)$$

$$\theta_3^{n+1} \sim \nu(\theta_3|\theta_1^{n+1}, \theta_2^{n+1}, \theta_4^n, ..., \theta_k^n)$$

jusqu'à :

$$\theta_k^{n+1} \sim \nu(\theta_k|\theta_1^{n+1}, ..., \theta_{k-1}^{n+1})$$

Les valeurs simulées par l'algorithme de Gibbs seront utilisées exactement de la même façon que celles simulées par la méthode de Hastings-Metropolis. En particulier il faudra définir la *burn-in period* n_1, et n_2.

Il faut enfin maintenant vérifier que ν est bien une loi stationnaire pour la chaîne de Markov ainsi simulée. Tout d'abord on note que le noyau de transition de cette chaîne s'écrit :

$$K(\theta, d\theta') = \nu(\theta_1'|\theta_2, ..., \theta_k)\nu(\theta_2'|\theta_1', \theta_3, ..., \theta_k)...\nu(\theta_k'|\theta_1', ..., \theta_{k-1}')d\theta'$$

Vérifions maintenant que si $\theta^n \sim \nu$ alors $\theta^{n+1} \sim \nu$, c'est-à-dire que pour tout A ensemble mesurable on a :

$$\mathbb{P}(\theta^{n+1} \in A) = \int_A \nu(\theta^{n+1})d\theta^{n+1}$$

Or on a :

$$\mathbb{P}(\theta^{n+1} \in A) = \int \mathbb{I}_A(\theta^{n+1})K(\theta^n, d\theta^{n+1})\nu(\theta^n)d\theta^n$$

$$... = \int \mathbb{I}_A(\theta^{n+1})\nu(\theta_1^{n+1}|\theta_2^n, ..., \theta_k^n)...\nu(\theta_k^{n+1}|\theta_1^{n+1}, ..., \theta_{k-1}^{n+1})\nu(\theta^n)d\theta^{n+1}d\theta^n$$

or :

$$\nu(\theta^n) = \nu(\theta_1^n|\theta_2^n, ..., \theta_k^n)\nu^1(\theta_2^n, ..., \theta_k^n)$$

où ν^j est le loi marginale de toutes les composantes sauf la j-ième.
On peut dès lors sortir de l'intégrale en $d\theta^n d\theta^{n+1}$ l'intégrale en $d\theta_1^n$ qui vaut :

$$\int \nu(\theta_1^n|\theta_2^n, ..., \theta_k^n)d\theta_1^n = 1$$

dès lors on peut regrouper les facteurs :

$$\nu(\theta_1^{n+1}|\theta_2^n, ..., \theta_k^n)\nu^1(\theta_2^n, ..., \theta_k^n) = \nu(\theta_1^{n+1}, \theta_2^n, ..., \theta_k^n)$$

On est alors à :

$$\mathbb{P}(\theta^{n+1} \in A) = \int \mathbb{I}_A(\theta^{n+1})\nu(\theta_2^{n+1}|\theta_1^{n+1}, \theta_3^n, ..., \theta_k^n)...$$

$$...\nu(\theta_k^{n+1}|\theta_1^{n+1}, ..., \theta_{k-1}^{n+1})\nu(\theta_1^{n+1}, \theta_2^n, ..., \theta_k^n)d\theta^{n+1}d(\theta_2^n, ..., \theta_k^n)$$

On recommence la même manipulation :

$$\nu(\theta_1^{n+1}, \theta_2^n, ..., \theta_k^n) = \nu(\theta_2^n|\theta_1^{n+1}, \theta_3^n, ..., \theta_k^n)\nu^2(\theta_1^{n+1}, \theta_3^n, ..., \theta_k^n)$$

ce qui permet d'intégrer en θ_2^n, et ainsi de suite jusqu'à intégrer en θ_k^n. On a alors :

$$\mathbb{P}(\theta^{n+1} \in A) = \int \mathbb{I}_A(\theta^{n+1})\nu(\theta_1^{n+1}, ..., \theta_k^{n+1})d\theta^{n+1} = \int_A \nu(\theta_{n+1})d\theta^{n+1}$$

ce qu'il fallait démontrer.

1.2.4 Utilisation en pratique

Notons tout d'abord qu'on peut montrer que l'algorithme de Gibbs peut se mettre sous la forme d'un cas particulier de l'algorithme de Hastings-Metropolis dans lequel les probabilités d'acceptation seraient toujours égales à 1.

Cependant, on voit bien que ces deux algorithmes ne présentent pas les mêmes avantages pratiquent et seront utilisés dans des contextes différents. En particulier, l'algorithme de Hastings-Métropolis ne n'impose que de savoir simuler selon $q(.|.)$ qui est choisie par l'utilisateur. On va voir dans la suite que l'algorithme de Gibbs est particulièrement adapté au cas de l'analyse bayésienne des modèles de mélanges à k fixe.

Notons aussi que certains algorithmes dits hybrides peuvent être utilisés. En effet, chaque étape de l'algorithme de Gibbs :

$$\theta_i^{n+1} \sim \nu(\theta_i|\theta_1^{n+1}, \theta_2^{n+1}, ..., \theta_{i-1}^{n+1}, \theta_{i+1}^{n}, ..., \theta_k^{n})$$

est en fait un noyau de transition pour lequel la loi cible ν est invariante. Si on ne sait pas simuler la loi $\nu(\theta_i|\theta_1^{n+1}, \theta_2^{n+1}, ..., \theta_{i-1}^{n+1}, \theta_{i+1}^{n}, ..., \theta_k^{n})$ dans une de ces étapes on peut remplacer cette étape par l'utilisation d'une probabilité de transition annexe, $q(.|.)$, et utiliser une méthode d'acceptation-rejet comme dans l'algorithme de Hastings-Metropolis. La loi ν sera toujours invariante pour la chaîne simulée. Il faut ensuite vérifier au cas par cas que la convergence a bien lieu (en vérifiant l'irréductibilité, l'apériodicité, ...). C'est une méthode de ce type qui sera utilisé pour l'analyse bayésienne des mélanges de distributions lorsque k est inconnu.

1.3 Inférence bayésienne dans les modèles de mélanges

1.3.1 Les modèles de mélanges comme modèles à données manquantes

Afin de pouvoir appliquer les algorithmes présentés précédemment aux modèles de mélange, il est nécessaire de modifier encore la façon dont est présentée le modèle :

$$g(x|\xi) = \sum_{i=1}^{k} p_i f(x|\theta_i)$$

Pour cela on crée en fait une variable supplémentaire "artificielle", z, qui représente la composante du mélange dont est issu x. En fait, $z{=}i$ où i est le groupe (ou la classe) de l'observation. On suppose donc que la variable étudiée est le couple (x, z) mais que le modèle est à "données manquantes", c'est-à-dire que la partie z n'est pas observée. On a alors :

$$\mathbb{P}(z = i|p_1, ..., p_k) = p_i$$

$$g(x|\xi, z) = f(x|\theta_z)$$

Selon le point de vue bayésien, les z_j peuvent aussi être vus comme des paramètres et le modèle devient alors un modèle hiérarchique, qui s'écrit :

$$g(x|\xi, z) = f(x|\theta_z)$$

$$\forall i \in [1, k], \mathbb{P}(z = i|\xi) = p_i$$

pour lequel il faut choisir une loi *a priori* $\pi(\xi) = \pi(\theta, p)$.

Pour un échantillon de n observations $x_1, ..., x_n$ on a donc :

$$(x_1, ..., x_n)|(z_1, ..., z_n, \xi) \sim \prod_{i=1}^{k} \prod_{j/z_j=i} f(x_j|\theta_i)$$

Notons enfin que selon les auteurs les conventions sont légèrement différentes pour z_j : parfois, $z_j = (0, ..., 0, 1, 0, ..., 0)$ avec le 1 en position i où i est la classe de x_j.

1.3.2 Utilisation de l'algorithme de Gibbs

L'utilisation de l'algorithme de Gibbs nécessite, comme on vient de le voir, de pouvoir simuler certaines lois. Ceci impose, dans notre cadre, de choisir des lois conjuguées pour les paramètres θ_i et p_i. Les lois utilisées ici sont celles proposées dans Robert (1996) et Diebolt et Robert (1994).

Si les $f(.|\theta_i)$ appartiennent à une même famille exponentielle :

$$f(x|\theta_i) = h(x)e^{\theta_i x - \psi(\theta)}$$

alors une loi conjuguée possible pour θ_i est :

$$\pi(\theta_i|\alpha,\beta) = Ce^{\theta_i\alpha - \beta\psi(\theta_i)}$$

et il est possible de trouver une loi conjugée pour les p_i, une loi dite de Dirichlet $\mathcal{D}(\alpha_1,...,\alpha_k)$ de densité :

$$\pi(p_1,...,p_k|\alpha_1,...,\alpha_k) = \frac{\Gamma(\alpha_1+...+\alpha_k)}{\Gamma(\alpha_1)...\Gamma(\alpha_k)}\prod_{i=1}^{k}(p_i^{\alpha_i-1}\mathbb{I}(p_i\geq 0))\mathbb{I}(\sum_{i=1}^{k}p_i=1)$$

par rapport à la mesure uniforme sur $\{(p_1,...,p_k)\in[0,1]^k/p_1+...+p_k=1\}$.

Rappelons que le modèle est sous la forme :

$$\mathbb{P}(z_j=i)=p_i$$

$$(x_1,...,x_n)|(z_1,...,z_n,\xi)\sim\prod_{i=1}^{k}\prod_{j/z_j=i}f(x_j|\theta_i)$$

où $\xi=(\theta_1,...,\theta_k,p_1,...,p_k)$. Or :

$$\prod_{j/z_j=i}f(x_j|\theta_i)=\left(\prod_{j/z_j=i}h(x_j)\right)e^{\theta_i\sum_{j/z_j=i}x_j-\psi(\theta_i)card(\{j/z_j=i\})}$$

$$\Rightarrow\prod_{j/z_j=i}f(x_j|\theta_i)=K_i(x)\pi(\theta_j|\alpha_i+\sum_{j/z_j=i}x_j,\beta_i+card(\{j/z_j=i\}))$$

En notant alors $n_i=card(\{j/z_j=i\})$ et $\overline{x_i}=\frac{1}{n_i}\sum_{j/z_j=i}x_j$ ceci se réécrit simplement :

$$\prod_{j/z_j=i}f(x_j|\theta_i)=K_i(x)\pi(\theta_i|\alpha_i+n_i\overline{x_i},\beta_i+n_i)$$

Donc :

$$(x_1,...,x_n)|(z_1,...,z_n,\xi)\sim K(x)\prod_{i=1}^{k}\pi(\theta_i|\alpha_i+n_i\overline{x_i},\beta_i+n_i)$$

et finalement on obtient la loi *a posteriori* des paramètres sous la forme :

$$\xi|(x_1,...,x_n,z_1,...,z_n)\sim\left(\prod_{i=1}^{k}p_i^{\alpha_i+n_i}\right)\left(\prod_{i=1}^{k}\pi(\theta_i|\alpha_i+n_i\overline{x_i},\beta_i+n_i)\right)$$

En particulier les p_i et les θ_i, sachant les x_j et les z_j, sont indépendants et on trouve les lois nécessaires pour la mise en place de l'algorithme de Gibbs :

$$\forall i,\theta_i|(p_1,...,p_k,x_1,...,x_n,z_1,...,z_n,(\theta_{i'},i'\neq i))\sim\pi(\theta_i|\alpha_i+n_i\overline{x_i},\beta_i+n_i)$$

$$(p_1,...,p_k)|(x_1,...,x_n,z_1,...,z_n,\theta_1,...,\theta_k))\sim\mathcal{D}(\alpha_1+n_1,...,\alpha_k+n_k)$$

$$\forall i, \forall j, \mathbb{P}(z_j = i | p_1, ..., p_k, x_1, ..., x_n, \theta_1, ..., \theta_k) = \frac{p_i f(x_j|\theta_i)}{\sum_{t=1}^{k} p_t f(x_j|\theta_t)}$$

Finalement, l'algorithme de Gibbs s'applique de la façon suivante. On choisit une valeur quelconque ξ^0 et des valeurs z_j^0, et on calcule les valeurs initiales de n_i et $\overline{x_i}$, notées n_i^0 et $\overline{x_i^0}$. Le passage de ξ^n et des valeurs z_j^n à ξ^{n+1} et z_j^{n+1} se fait de la façon suivante :

- on simule :

$$\forall i, \theta_i^{n+1} \sim \pi(\theta_i | \alpha_i + n_i^n \overline{x_i^n}, \beta_i + n_i^n)$$

- on simule :

$$(p_1^{n+1}, ..., p_k^{n+1}) \sim \mathcal{D}(\alpha_1 + n_1^n, ..., \alpha_k + n_k^n)$$

- on simule : $\forall j, z_j^n$ suivant la loi :

$$\mathbb{P}(z_j = i) = \frac{p_i^{n+1} f(x_j|\theta_i^{n+1})}{\sum_{t=1}^{k} p_t^{n+1} f(x_j|\theta_t^{n+1})}$$

- on met à jour les valeurs :

$$n_i^{n+1} = card(\{j / z_j^{n+1} = i\})$$

$$\overline{x_i^{n+1}} = \frac{1}{n_i^{n+1}} \sum_{j/z_j^{n+1}=i} x_j$$

1.3.3 Utilisation de l'algorithme de Hastings-Metropolis

Il serait aussi possible d'utiliser la méthode de Hastings-Metropolis dans le cas des modèles de mélanges. En fait, dans la mesure où cet algorithme ne nécessite pas de savoir simuler la loi d'un paramètre conditionnellement à tous les autres, l'utilisation de cet algorithme impose moins de restrictions sur les lois *a priori*.

Rappelons la formulation du problème sous forme hiérarchique :

$$g(x|\xi, z) = f(x|\theta_z)$$

Notons $\zeta = (\xi, z)$, la loi *a priori* sur ζ, notée $p(\zeta)$ est donnée par :

$$\forall i \in [1, k], \mathbb{P}(z = i|\xi) = p_i$$

et la loi *a priori* $\pi(\xi) = \pi(\theta, p)$.

On peut faire les mêmes choix que dans la partie sur l'algorithme de Gibbs pour $p(.)$, mais ceci n'est pas une obligation. Rappelons cependant qu'il ne faut pas utiliser de lois *a priori* impropres.

Alors :

$$(x_1, ..., x_n)|\zeta \sim \prod_{i=1}^{k} \prod_{j:z_j=i} f(x_j|\theta_i)$$

$$p(\zeta|x) \propto p(\zeta) p(x|\zeta)$$

$$p(\zeta|x) \propto \pi(\xi) \left(\prod_{j=1}^{n} p_j \right) \prod_{i=1}^{k} \prod_{j:z_j=i} f(x_j|\theta_i)$$

$$p(\zeta|x) \propto \pi(\xi) \prod_{i=1}^{k} \prod_{j:z_j=i} p_j f(x_j|\theta_i)$$

Un dernier choix à faire est celui de $q(\zeta'|\zeta)$; en général q choisit d'abord un des paramètres selon une certaine loi de probabilités puis propose une modification de ce paramètre. Ainsi, $q = \sum_{m \in M} q_m$, où q_m ne modifiera qu'un seul paramètre. Notons que dans ce cas, q_m n'est pas vraiment la densité d'une probabilité de transition car en général, $\forall \zeta, q_m(Z|\zeta) < 1$ où Z est l'ensemble où le vecteur de paramètres ζ varie. En fait, afin que q soit une probabilité de transition on a :

$$\forall \zeta, \sum_{m \in M} q_m(Z|\zeta) = 1$$

voire ≤ 1 si on garde la possibilité de ne pas proposer de modification des paramètres à chaque étape. Ici, par exemple, on pourrait prendre $q_1(.|\zeta)$ qui propose une modification de θ, $q_2(.|\zeta)$ qui propose une modification de z et $q_3(.|\zeta)$ pour p, avec $M = \{1, 2, 3\}$.

Reste alors à calculer la probabilité d'acceptation d'une modification de ζ en ζ' :

$$a(\zeta, \zeta') = \min \left(1, \frac{q(\zeta|\zeta')\pi(\xi') \prod_{i=1}^{k} \prod_{j:z'_j=i} p'_j f(x_j|\theta'_i)}{q(\zeta'|\zeta)\pi(\xi) \prod_{i=1}^{k} \prod_{j:z_j=i} p_j f(x_j|\theta_i)} \right)$$

1.3.4 Problèmes rencontrés

Un premier problème apparaît rapidement lorsque l'on analyse les résultats, c'est-à-dire lorsqu'on visualise les lois *a posteriori* des paramètres : celles-ci sont multimodales, ce qui est du au fait que les paramètres de différentes composantes ont pu être échangés au cours des simulations ! En effet, il n'y a pas de "paramétrage naturel" des modèles de mélanges : par exemple, si il y a deux composantes ($k = 2$), laquelle devrait être rattachée à p_1 et θ_1 et laquelle à p_1 et θ_1 ? Pour remédier à ce problème, différentes solutions ont été proposées, la plus simple étant d'imposer une contrainte d'ordre sur les θ_i. Par exemple, si les différentes composantes sont toutes des lois normales, avec $\theta_i = (\mu_i, \sigma_i^2)$ alors on peut imposer $\mu_1 < ... < \mu_k$. En fait, cette méthode ne résout pas complètement le problème. D'autres propositions sont étudiées dans la partie suivante.

Un autre problème important reste celui de la convergence des méthodes par MCMC. Afin que la convergence soit assurée pour les quantités d'intérêt, il faut s'assurer que la chaîne de Markov simulée dans l'espace des paramètres a bien visité les zones de fortes probabilités selon la loi *a posteriori*. Notamment, la façon dont on choisit les valeurs initiales est très importante. Si on les choisit mal, la période de démarrage ou *burn-in period* peut-être bien plus longue que la taille de la chaîne que l'on a simulé, ce qui signifie que les résultats obtenus n'auront aucun sens. Certains auteurs ont proposé d'initialiser les algorithmes en utilisant un algorithme de classification classique et rapide, par exemple la méthode des (k) centres mobiles.

2 Détermination du nombre de composantes par MCMC à saut réversible

Comme on l'a vu en introduction, dans certains cas les composantes $f(.|\theta_i)$ ont une signification "physique", et représentent des "groupes" d'individus : k peut alors être connu à l'avance, mais ce n'est pas toujours le cas ! Il devient alors intéressant de considérer k comme un paramètre à estimer. Il se peut même que k soit pratiquement le seul paramètre d'intérêt du modèle !

2.1 Introduction

2.1.1 Formulation du problème

On s'intéresse donc maintenant à l'étude de modèles de mélanges pour lesquels le nombre de composantes est inconnu. Le modèle reste :

$$\sum_{i=1}^{k} p_i f(x|\theta_i)$$

mais le rôle de k est transformé : k n'est plus une constante, mais un paramètre qu'il faudra lui aussi estimer. On a donc maintenant :

$$g(x|\xi, k) = \sum_{i=1}^{k} p_i f(x|\theta_i)$$

Dans une approche bayésienne, il faudra donc choisir une loi *a priori* sur le paramètre k. En général, celle-ci est une loi de Poisson de paramètre λ, ou parfois une loi uniforme sur un intervalle d'entiers, $[1, 100]$ par exemple.

Le modèle proposé est donc un modèle hiérarchique. En effet :

$$\xi|k \sim \pi(.|k)$$

où $\pi(.|k)$ est une loi sur l'espace dans lequel varie $\xi = (\theta_1, ..., \theta_k, p_1, ..., p_k)$. On voit bien que les lois $\pi(.|k = 2)$ et $\pi(.|k = 3)$ par exemple ne peuvent pas être les mêmes car elles ne sont pas définies sur des espaces de même dimension. Cette formulation sera rendue plus explicite dans la suite où on présente la méthode de Green (1997) pour aborder ce problème.

2.1.2 Objectif et problème principal

Il suffit de connaître la loi de $(k, \xi)|x$ ou mieux les lois de $k|x$ puis de $\xi|k, x$. En particulier la loi de $k|x$ sera cruciale pour le choix du nombre de composantes.

Le problème est alors le même que dans le cas de l'analyse bayésienne des modèles de mélanges à k fixé : ces lois sont donnés par une formule pas exploitable en pratique, et il va falloir recourir à des méthodes par MCMC.

2.2 Les algorithmes MCMC à sauts réversibles

2.2.1 Présentation générale

Le problème rencontré ici reste donc le même que pour les modèles de mélanges à k fixé : la loi *a posteriori* en général pas être connue explicitement. Il faut donc utiliser des méthodes MCMC afin de simuler une chaîne de Markov de loi stationnaire cette loi *a posteriori*.

Seulement, cette loi est définie sur un espace particulier. Si Θ_k est l'espace du paramètre $\xi_k = (\theta_1, ..., \theta_k, p_1, ..., p_k)$, alors la loi *a posteriori* sur $t = (k, \xi)$ est définie sur :

$$\Theta = [\{1\} \times \Theta_1] \bigcup [\{2\} \times \Theta_2] \bigcup [\{3\} \times \Theta_3] \bigcup ...$$

Ainsi, la chaîne de Markov simulée doit être capable de "sauter" d'un espace Θ_k dans un autre, alors même que ces espaces ne sont pas de même dimension.

Le premier à avoir réussi à mettre cette méthode en pratique est Peter Green, en adaptant l'algorithme de Hastings-Metropolis à des sauts d'un espace Θ_k à un autre. Ainsi, les sauts peuvent être accepté, ou rejetés, comme n'importe quel mouvement dans l'algorithme de Hastings-Metropolis. Afin d'assurer la convergence de la méthode, les sauts doivent satisfaire une condition de réversibilité ce qui a conduit Green à appeler la méthode "MCMC à sauts réversibles", ou *reversible jump* MCMC, abrégée en RJMCMC. Il a tout d'abord présenté l'idée dans la discussion d'un article de Grenander et Miller (1994) et l'a développé dans *Reversible jump Markov chain Monte Carlo computation and Bayesian model determination* (1995).

Dans cet article, cette méthode est utilisée dans différents types de problèmes bayésiens dans lesquels la dimension du paramètre n'est pas connue *a priori* : choix des regresseurs dans une regression, segmentation d'image, évaluation de la tendance d'une série avec un nombre inconnu de changements de tendance. Le problème de l'estimation d'un modèle de mélanges avec un nombre inconnu de composantes était seulement cité comme exemple.

Richardson et Green, dans *On Bayesian analysis of mixtures with an unknown number of components* (1997), ont consacré un article complet à l'application de cette méthode aux modèles de mélanges. La suite de cette partie est consacrée à la description de leur méthode et à la présentation de leurs résultats.

2.2.2 Principe des RJMCMC

Cette section présente la méthode des RJMCMC telle qu'elle est exposée par Green (1995).

L'algorithme proposé est donc une adaptation de celui de Hastings-Metropolis. Notons $\pi(t)$ pour simplifier la loi *a posteriori* du paramètre t (qui contient k, les θ_i et les p_i), c'est-à-dire que l'on omet de noter le "sachant l'observation x" dans toute cette sous-partie 2.2.2. A chaque étape, différents types de modification des paramètres sont proposées, puis rejetées ou acceptées :

- mise à jour des poids p_i,
- mise à jour des paramètres θ_i,
- mise à jour des z_j,
- changement du nombre de composantes k.

C'est ce dernier type de modification qui pose un problème. En fait, plusieurs changements peuvent être proposés : augmenter ou diminuer k de 1, de 2...

Concentrons-nous pour le moment sur la probabilité de transition q dans ce dernier cas uniquement (modification de k). On note l'ensemble de toutes les modifications possibles M, et comme on l'a fait précédemment on écrit :

$$q(A|t) = \sum_{m \in M} q_m(A|t)$$

où $q_m(A|t)$ est la probabilité de proposer l'ensemble A en partant de l'état t si on tente une modification de type m. On rappelle que :

$$\sum_{m \in M} q_m(\Theta|t) = 1$$

ou ≤ 1 si on garde la possibilité de ne pas proposer de modifications à certaines étapes.

Notons que pour un état t donné il est toujours possible qu'il y ait un type de modification m tel que : $q_m(\Theta, t) = 0$. C'est même toujours le cas en pratique : en effet, m peut être une modification des paramètres θ ou des poids p mais m peut aussi impliquer un changement du nombre de composantes k : par exemple $1 \leftrightarrow 2$, $1 \leftrightarrow 3$, $2 \leftrightarrow 3$... Il est logique que si $k = 1$, la modification $m = 2 \leftrightarrow 3$ ne puisse pas être proposée...

Le but est maintenant de calculer la probabilité d'acceptation $a_m(t, t')$ qu'il faut associer à q_m si l'on souhaite simuler une chaîne de Markov de probabilité stationnaire π. Pour cela, on peut se rappeler que la *detailed balance condition*, si elle n'est pas une condition nécessaire, est une condition suffisante pour assurer ceci. Rappelons que dans un algorithme de Hastings-Metropolis simple, la probabilité de transition de la chaîne de Markov simulée était :

$$P(t, B) = a(t, B)q(B|t) + (1 - a(t, B))\mathbb{I}_B(t)$$

elle pourra s'écrire dans ce cas :

$$P(t, B) = \left(\sum_{m \in M} \int_B q_m(dt'|t) a_m(t, t') \right)$$

$$+\mathbb{I}_B(t) \left(\sum_{m \in M} \int_\Theta q_m(dt'|t)[1 - a_m(t, t')] \right) + \mathbb{I}_B(t)(1 - \sum_{m \in M} q_m(\Theta|t))$$

Le premier terme correspond à l'acceptation de la modification proposée par q, le second terme à son rejet et le troisième terme au cas hypothétique où aucune modification n'aurait été proposée.

Pour simplifier, Green note :

$$s(t) = 1 - \sum_{m \in M} q_m(\Theta|t) + \left(\sum_{m \in M} \int_\Theta q_m(dt'|t)[1 - a_m(t,t')] \right)$$

donc :

$$P(t,B) = \left(\sum_{m \in M} \int_B q_m(dt'|t) a_m(t,t') \right) + \mathbb{I}_B(t) s(t)$$

La *detailed balance condition* sera assurée si, $\forall A, \forall B$ mesurables dans Θ on a :

$$\int_A \pi(dt) P(t,B) = \int_B \pi(dt) P(t,A)$$

soit :

$$\sum_m \int_A \pi(dt) \int_B q_m(dt'|t) a_m(t,t') + \int_{A \cap B} \pi(dt) s(t)$$

$$= \sum_m \int_B \pi(dt') \int_A q_m(dt|t') a_m(t',t) + \int_{B \cap A} \pi(dt') s(t')$$

Les deux intégrales simples sont évidemment les mêmes. Donc une condition suffisante, et simple, pour assurer la condition désirée est la réversibilité de chaque saut proposé, soit $\forall m \in M, \forall A, \forall B$:

$$\int_A \pi(dt) \int_B q_m(dt'|t) a_m(t,t') = \int_B \pi(dt') \int_A q_m(dt|t') a_m(t',t)$$

Or, supposons que la mesure $\pi(dt) q_m(dt'|t)$ ait une densité $f_m(t,t')$ par rapport à une mesure ν_m symétrique sur l'espace produit Θ^2, on a alors :

$$\int_A \pi(dt) \int_B q_m(dt'|t) a_m(t,t') = \int_A \int_B \nu_m(dt,dt') f_m(t,t') a_m(t,t')$$

et symétriquement :

$$\int_B \pi(dt') \int_A q_m(dt|t') a_m(t',t) = \int_B \int_A \nu_m(dt',dt) f_m(t',t) a_m(t',t)$$

ainsi, la condition d'égalité s'écrit est assurée si a_m vérifie :

$$\forall m \in M, \forall (t,t') \in \Theta^2 : a_m(t,t') f_m(t,t') = a_m(t',t) f_m(t',t)$$

Pour ceci il suffit de prendre :

$$a_m(t,t') = min\left(1, \frac{f_m(t',t)}{f_m(t,t')}\right)$$

que l'on peut réécrire formellement :

$$a_m(t,t') = min\left(1, \frac{\pi(dt') q_m(dt|t')}{\pi(dt) q_m(dt'|t)}\right)$$

En fait, cette forme peut être rendue plus explicite à condition de préciser la forme des q_m.

On se place dans le cas où q_m propose un saut de l'espace $[\{k_1\} \times \Theta_{k_1}]$ de dimension d_1 à l'espace $[\{k_2\} \times \Theta_{k_2}]$ de dimension $d_2 > d_1$, ou le saut de $[\{k_2\} \times \Theta_{k_2}]$ dans $[\{k_1\} \times \Theta_{k_1}]$. Rappelons que $t = (k, \xi)$ et que dans le couple (k_1, ξ) le ξ est de dimension d_1 alors qu'il est de dimension d_2 dans dans le couple

(k_2, ξ). On supposera que $\pi(dt|k = k_1)$ admet une densité $f_k(t)$ par rapport à la mesure de Lebesgue dans Θ_{k_1}, on notera par commodité :

$$\pi(dt) = \pi(k)\pi(dt|k) = \pi(k)f_k(t)dt = \pi(t)dt$$

On note $j((k_1, \xi_1))$ la probabilité de choisir le type de saut m lorsqu'on est en (k_1, ξ_1), et la probabilité de choisir le saut m (en sens inverse) lorsqu'on est en (k_2, ξ_2) sera notée $j((k_2, \xi_2))$. En fait :

$$j((k_1, \xi_1)) = \int_\Theta q_m(dt'|(k_1, \xi_1))$$

$$j((k_2, \xi_2)) = \int_\Theta q_m(dt|(k_2, \xi_2))$$

Il faut maintenant expliciter la façon dont sera réalisé un saut. On l'a vu, intuitivement le problème est que la dimension des deux espaces entre lesquels on saute n'est pas la même. La façon d'accomplir ce saut proposée par Green est la suivante :

- on génère un vecteur aléatoire de dimension $d = d_2 - d_1$, noté $u^{(1)}$, selon une loi de densité $q_1(u^{(1)})$ par rapport à la mesure de Lebesgue.
- on prend ensuite $t' = g(t, u^{(1)})$ où g est une fonction déterministe. g doit vérifier :

$$\forall t, \forall t', \exists! u^{(1)} : g(t, u^{(1)}) = t'$$

$$\exists g_{inv}, \forall t, \forall u^{(1)} : g_{inv}(g(t, u^{(1)})) = t$$

Autrement dit, pour un saut de Θ_{k_1} dans Θ_{k_2} il faut "compléter" la dimension par l'introduction de $u^{(1)}$ qui est aléatoire et utiliser g, alors que pour un saut inverse de Θ_{k_2} dans Θ_{k_1} il faut utiliser g_{inv} et la transformation proposée est déterministe.

Maintenant que q_m est décrite il reste à déterminer la mesure symétrique ν_m et la densité f_m. Green définit ν_m par :

$\forall A \subset \{k_1\} \times \Theta_{k_1}, \forall B \subset \{k_2\} \times \Theta_{k_2}$ mesurables :

$$\nu_m(A \times B) = \nu_m(B \times A) = \lambda\{(t, u^{(1)})/\xi_1 \in A, g(t, u^{(1)}) \in B\}$$

où λ est la mesure de Lebesgue sur $\mathbb{R}^{d_2}$, et :

$\forall A \subset \Theta, \forall B \subset \Theta$ mesurables :

$$\nu_m(A \times B) = \nu_m\{(A \cap \Theta_{k_1}) \times (B \cap \Theta_{k_2})\} + \nu_m\{(A \cap \Theta_{k_2}) \times (B \cap \Theta_{k_1})\}$$

Notons que ν_m est symétrique, ce qui était nécessaire. Alors on a :

$$\pi(dt)q_m(dt'|t) = f(t, t')\nu_m(dt, dt')$$

avec f définie de la façon suivante : $\forall t = (k_1, \xi_1) \in \Theta_{k_1}, \forall t' = (k_2, \xi_2) \in \Theta_{k_2}$

$$f(t, t') = \pi(t)j(t)q_1(u^{(1)})$$

$$f(t', t) = \pi(t')j(t') \left| \frac{\partial g(t, u^{(1)})}{\partial (t, u^{(1)})} \right|$$

et $f(x, x') = 0$ dans tous les autres cas. Dans ces cas on obtient donc une formule plus explicite pour la probabilité d'acceptation :

$$a(t, t') = \min\left(1, \frac{\pi(t')j(t')}{\pi(t)j(t)q_1(u^{(1)})} \left| \frac{\partial g(t, u^{(1)})}{\partial (t, u^{(1)})} \right|\right)$$

La preuve de cette assertion n'est donnée explicitement ni dans Green (1995) ni dans Richardson et Green (1997). En fait, ce résultat peut être vu de façon intuitive en utilisant la formule générique :

$$a_m(t,t') = min\left(1, \frac{\pi(dt')q_m(dt|t')}{\pi(dt)q_m(dt'|t)}\right)$$

et la définition de q_m.

Le résultat est néanmoins démontré rigoureusement à partir de la *detailed balance condition* dans Waagepetersen et Sorensen (2000) ou dans Saint-Pierre et Garel (2001). Rappelons que cette condition s'écrivait :

$$\int_A \pi(dt) \int_B q_m(dt'|t)a_m(t,t') = \int_B \pi(dt') \int_A q_m(dt|t')a_m(t',t)$$

On a :

$$\int_A \pi(dt) \int_B q_m(dt'|t)a_m(t,t') = ...$$

$$... = \int_A \pi(t)dt \int du^{(1)} q_1(u^{(1)})j(t)a_m(t,g(t,u^{(1)}))\mathbb{I}_B(g(t,u^{(1)}))$$

$$... = \int\int q_1(u^{(1)})j(t)a_m(t,g(t,u^{(1)}))\mathbb{I}_B(g(t,u^{(1)}))\pi(t)\mathbb{I}_A(t)du^{(1)}dt$$

et :

$$\int_B \pi(dt') \int_A q_m(dt|t')a_m(t',t) = ...$$

$$... = \int_B \pi(t')dt' \int_A j(t')a_m(t',t)d\delta_{g_{inv}(t')}(t)$$

$$... = \int \pi(t')dt'\mathbb{I}_B(t')j(t')a_m(t',g_{inv}(t'))\mathbb{I}_A(g_{inv}(t'))$$

Il suffit alors d'obtenir le changement de variable $t' = g(t,u^{(1)})$ pour obtenir :

$$... = \int\int \pi(g(t,u^{(1)}))\mathbb{I}_A(t)\mathbb{I}_B(g(t,u^{(1)}))j(g(t,u^{(1)}))a_m(g(t,u^{(1)}),t)\left|\frac{\partial g(t,u^{(1)})}{\partial(t,u^{(1)})}\right| du^{(1)}dt$$

La relation d'égalité entre les deux résultats est effectivement satisfaite par la solution donnée par Green :

$$a(t,t') = \min\left(1, \frac{\pi(g(t,u^{(1)}))j(g(t,u^{(1)}))}{\pi(t)j(t)q_1(u^{(1)})}\left|\frac{\partial g(t,u^{(1)})}{\partial(t,u^{(1)})}\right|\right)$$

2.3 La méthode retenue par Richardson et Green (1997)

C'est donc dans *On Bayesian analysis of mixtures with an unknown number of components* (1997) que Richardson et Green ont appliqué la première fois cette idée de RJMCMC pour estimer le nombre de composantes dans un mélange de distributions. Cette sous-partie présente les choix qui ont été faits par les auteurs dans cet article pour les points suivants :

- le modèle estimé.
- les lois *a priori*.
- les types de mouvement ($m \in M$) et les transitions associées (q_m), ainsi que les calculs des probabilités d'acceptation associées (a_m).

Notons que l'aspect théorique des RJMCMC ayant été présenté dans Green (1995), l'article de Richardson et Green détaille simplement une application de cette méthode pour un modèle de mélanges particulier, en précisant cependant que la méthode peut être adaptée sans difficulté à d'autres modèles. Il s'agit ici d'un mélange de lois normales unidimensionnelles.

Pour ce qui est du choix de M et des q_m, les auteurs précisent que les choix ont été faits de façon arbitraire, ou du moins qu'aucune optimisation des performances de l'algorithme n'a été menée à travers ce choix : les résultats obtenus sont néanmoins satisfaisants. La recherche de q_m "optimaux" a suscité depuis la parution d'un certain nombre d'articles, comme ceci sera montré dans la suite.

2.3.1 Cadre général, notations

On adopte ici les notations retenues par Richardson et Green qui sont légèrement différentes. Les observations sont notées y_i pour $i \in \{1, ..., n\}$, et $y = (y_1, ..., y_n)$. Le nombre de composantes est donc un paramètre du modèle, toujours noté k et les paramètres de chaque composante sont toujours noté θ_j, $j \in \{1, ..., k\}$, et $\theta = (\theta_1, ..., \theta_n)$. Les variables d'allocations sont toujours notées $z_i \in \{1, ..., k\}, j \in \{1, ..., n\}$. Les poids sont maintenant notés w_j pour $j \in \{1, ..., k\}$, et $w = (w_1, ..., w_k)$.

$$y_i|k, \theta, w \sim \sum_{j=1}^{k} w_j f(.|\theta_j) = f(y_i|k, \theta, w)$$

$$\forall j \in \{1, ..., k\}, \mathbb{P}(z_i = j|k, w) = w_j$$

$$y_i|k, \theta, z \sim f(.|\theta_{z_i})$$

Devant le nombre de lois considérées, une notation générique est adoptée : $p(.)$. La loi jointe des paramètres et des variables et la suivante :

$$p(k, w, z, \theta, y) = p(k)p(w|k)p(z|w, k)p(\theta|z, w, k)p(y|\theta, z, w, k)$$

$p(z|w, k)$ a été complètement définie précédemment, il reste à préciser $p(k)$, $p(w|k)$, $p(\theta|z, w, k) = p(\theta|k)$ et $p(y|\theta, z, w, k) = p(y|\theta, z)$ c'est-à-dire $f(.|\theta)$.

En fait, un niveau hiérarchique supplémentaire est rajouté par les auteurs. Ceci permettra notamment de diminuer l'influence des lois *a priori* sur les résultats obtenus, ce point est discuté dans la présentation des résultats, ou de les calibrer de façon adéquate. Les auteurs introduisent l'hyperparamètre λ qui sera paramètre de la loi *a priori* de k, δ pour w et η pour θ. L'intérêt est le suivant : si les lois *a posteriori* semblent trop sensible à des choix particuliers de η, par exemple, le niveau hiérarchique supérieur permet de munir η lui-même d'une loi *a priori* ce qui a pour effet de minimiser cette dépendance.

On obtient finalement la loi jointe des observations, paramètres et hyperparamètres :

$$p(\lambda, \delta, \eta, k, w, z, \theta, y) = p(\lambda)p(\delta)p(\eta)p(k|\lambda)p(w|k, \delta)p(z|w, k)p(\theta|k, \eta)p(y|\theta, z)$$

On peut poser $t = (\lambda, \delta, \eta, k, w, z, \theta)$ pour recoller aux notations des parties précédentes et alléger les notations ; on a :

$$p(t|y) \propto p(\lambda, \delta, \eta, k, w, z, \theta, y)$$

2.3.2 Spécification du modèle

Détaillons maintenant les choix retenus par les auteurs pour les dernières lois qui n'ont pas encore été spécifiées.

Tout d'abord, il faut préciser la loi du paramètre k. Le choix retenu dans les autres applications des RJMCMC par Green était une loi de Poisson de paramètre λ, notée ici $\mathcal{P}(\lambda)$. Cependant, afin de faciliter la présentation, Green retient ici une loi uniforme sur $\{1, ..., k_{max} = \lambda\}$:

$$k \sim \mathcal{U}(\{1, ..., k_{max}\})$$

$$\mathbb{P}(k = k_0) = \frac{1}{k_{max}} \mathbb{I}_{\{1,...,k_{max}\}}(k_0)$$

Ensuite, comme on l'a mentionné précédemment, les observations suivent des lois normales unidimensionnelles :

$$y_i|\theta, z \sim \mathcal{N}(\mu_{z_i}, \sigma^2_{z_i})$$

donc on peut détailler ici $\theta_j = (\mu_j, \sigma_j^2) \in \mathbb{R} \times \mathbb{R}_+$. Ainsi :

$$f(y_i|\theta, z) = \frac{1}{\sqrt{2\pi\sigma^2_{z_i}}} e^{-\frac{(y_i - \mu_{z_i})^2}{2\sigma^2_{z_i}}}$$

Il faut donc maintenant préciser les lois *a priori* sur $\theta_j = (\mu_j, \sigma_j^2)$. On suppose en fait que tous les μ_j et tous les σ_j^2 sont indépendants et :

$$\mu_j|\eta, k \sim \mathcal{N}(\xi, \kappa^{-1})$$

$$\sigma_j^2|\eta, k \sim \mathcal{IG}(\alpha, \beta)$$

avec l'hyperparamètre $\eta = (\xi, \kappa^{-1}, \alpha, \beta)$, et $\mathcal{IG}$ est une loi gamma inverse c'est-à-dire que :

$$\sigma_j^{-2}|\eta, k \sim \Gamma(\alpha, \beta)$$

$$p(\sigma_j^{-2}|\eta, k) = \frac{\beta^\alpha}{\Gamma(\alpha)} (\sigma_j^{-2})^{\alpha-1} e^{-\beta\sigma_j^{-2}} \mathbb{I}_{\mathbb{R}_+}(\sigma_j^{-2})$$

On obtient par le calcul :

$$p(\sigma_j^2|\eta, k) = \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{1}{(\sigma_j^2)^{\alpha+1}} e^{-\frac{\beta}{\sigma_j^2}} \mathbb{I}_{\mathbb{R}_+}(\sigma_j^{-2})$$

Une restriction cependant est à apporter : comme on l'a vu dans la première partie il est nécessaire d'imposer des contraintes d'identifications sur les composantes. Ici, les contraintes retenues par les auteurs sont :

$$\mu_1 < ... < \mu_k$$

Notons que, pour ce qui est des paramètres θ et p, le choix des lois *a priori* est assez semblable à celui fait par Robert (1996) et présentée dans la première partie. La raison va être présentée assez rapidement : en fait, lors des simulations, Green et Richardson n'utilisent la méthode de Hastings-Metropolis que pour les changements de k. Afin de simplifier le problème, pour la modification des autres paramètres, c'est la méthode de Gibbs qui a été retenue. Le choix de lois *a priori* fait ici permet d'appliquer l'algorithme de Gibbs, ceci sera détaillé bien sûr par la suite.

Le choix retenu pour le vecteur des poids $w = (w_1, ..., w_k)$ est une loi de Dirichlet symétrique de paramètre δ :

$$w|\delta, k \sim \mathcal{D}(\delta, ..., \delta)$$

déjà présentée dans la première partie.

Reste enfin à spécifier les lois des hyperparamètres. En fait, dans chaque application de la méthode proposée par les auteurs, $\lambda = k_{max}$ sera fixé à 30, ce qui n'aura aucune conséquence comme on le verra par la suite.

Pour le choix des autres hyperparamètres, les auteurs précisent avoir voulu être "le moins informatif possible", rappelons que l'utilisationde loi complètement non-informatives dans ce contexte est impossible. Afin que la loi (*a priori*) des μ_j soit relativement plate sur l'intervalle où varient les données, les auteurs prennent ξ égal au milieu de cet intervalle et κ un petit multiple de $1/R^2$ où R est la longueur de cet intervalle.

L'hyperparamètre δ est fixé, égal à 1 dans toutes les applications. Si on revient à la définition de la loi de Dirichlet (donnée dans la première partie), ceci signifie simplement que w est de loi uniforme sur le simplexe de $\mathbb{R}^k$.

Reste finalement à déterminer α et β. Après avoir mené de premières simulations à α et β fixés, les auteurs se sont aperçus que la loi *a posteriori* obtenue sur les paramètres σ_j^2 dépendait fortement des valeurs de α et β.

Rappelons que :

$$\mathbb{E}(\sigma_j^{-2}|\alpha,\beta,k) = \frac{\alpha}{\beta}$$

$$\mathbf{V}(\sigma_j^{-2}|\alpha,\beta,k) = \frac{\alpha}{\beta^2}$$

Afin de diminuer cette dépendance, les auteurs ont donc décidé d'introduire un niveau hiérarchique supplémentaire en prenant en fait α fixe mais :

$$\beta|g,h \sim \Gamma(g,h)$$

Les auteurs remarquent qu'en fixant $\alpha > 1 > g$ on introduit une information *a priori* sur les σ_j^2 en les contraignant à être du même ordre de grandeur, ce qui est en fait souhaitable si on souhaite avoir des résultats cohérents. Les auteurs retiennent donc ce choix et prennent h égal à un petit multiple de $1/R^2$.

Finalement, une fois introduit ce dernier niveau hiérarchique, les résultats obtenus qui seront présentés plus loin se sont avérés assez peu sensibles aux choix particuliers faits pour g, h, α, κ.

Notons enfin que, pour toutes les mises en oeuvre sur données réelles que les auteurs font à la fin de l'article, on a : $\kappa = 1/R^2$, $\alpha = 2$, $g = 0.2$, $h = 10/R^2$, $\delta = 1$ et $k_{max} = 30$.

2.3.3 Méthode de simulation : choix des types de mouvements

Les auteurs distinguent six types principaux de mouvements :
(a) modification des poids w ;
(b) modification des paramètres des lois normales, (μ, σ) ;
(c) modification des variables d'allocation z ;
(d) modification de l'hyperparamètre β ;
(e) modification de k : séparer une composante en deux ou en réunir deux en une ;
(f) modification de k : créer ou supprimer une composante vide.

Plutôt que de choisir un type de mouvement au hasard, les auteurs utilisent un algorithme qui passe systématiquement par les étapes (a), (b), (c), (d), (e) et (f) dans l'ordre à chaque étape ou *sweep*. En fait, le choix de lois *a priori* qu'ils ont effectué leur permet de traiter les étapes (a) à (d) par la méthode proposée par Diebolt et Robert (1994), c'est-à-dire par un algorithme de Gibbs (présenté en détail dans la première partie). En revanche, les étapes (e) et (f) ne peuvent être traitées que par un algorithme de Hastings-Metropolis, c'est pour ces étapes que la méthode des RJMCMC développée par Green (1995) se révèle utile.

Examinons donc tout d'abord les étapes (a) à (d) qui nécessitent de connaître la loi conditionnelle de chaque paramètre, sachant tous les autres et sachant les observations. La loi jointe des paramètres, hyperparamètres et observations était écrite sous la forme :

$$p(t,y) = p(\lambda,\delta,\eta,k,w,z,\theta,y)$$

$$p(t,y) = p(\lambda)p(\delta)p(\eta)p(k|\lambda)p(w|k,\delta)p(z|w,k)p(\theta|k,\eta)p(y|\theta,z)$$

Comme on a retenu $\lambda = k_{max}$ et $\delta = (g,h)$ constants on a $p(\lambda) = p(\delta) = 1$, et on a $\eta = (\xi, \kappa^{-1}, \alpha, \beta)$ avec tous ces hyperparamètres constants sauf β qui suit une loi gamma. On obtient finalement :

$$p(\lambda,\delta,\eta,k,w,z,\theta,y) = \frac{1}{k_{max}}\mathbb{I}_{\{1\leq k\leq k_{max}\}}\frac{h^g}{\Gamma(g)}\beta^{g-1}e^{-h\beta}\left(\prod_{j=1}^{k} w_j^{\delta-1}\right)...$$

$$...\left(\prod_{j=1}^{k} w_j^{n_j}\right)\left(\prod_{j=1}^{k}\frac{\beta^\alpha}{\Gamma(\alpha)}(\sigma_j^{-2})^{\alpha-1}e^{-\beta\sigma_j^{-2}}\right)\left(\prod_{j=1}^{k}\frac{e^{\frac{-(\mu_j-\xi)^2}{2\kappa^{-1}}}}{\sqrt{2\pi\kappa^{-1}}}\right)...$$

$$... \left(\prod_{i=1}^{n} \frac{e^{\frac{-(y_i-\mu_{z_i})^2}{2\sigma_{z_i}^2}}}{\sqrt{2\pi\sigma_{z_i}^2}} \right) \mathbb{I}_{\{\beta>0, w_1>0, ..., w_k>0, w_1+...+w_k=1, \kappa^{-1}>0, \sigma_1^2>0, ..., \sigma_k^2>0\}}$$

où $n_j = card(\{i = 1, ..., n : z_i = j\})$. Comme on impose en plus la contrainte $\mu_1 < ... < \mu_k$, il faudrait encore en principe multiplier cette expression par $\mathbb{I}_{\mu_1<...<\mu_k} k!$

L'étape (a) consiste donc à modifier les paramètres de poids w. Le calcul de la loi conditionnelle sachant tous les autres paramètres et les observations est direct à partir de la formule ci-dessus (on omet les indicatrices pour simplifier l'écriture) :

$$p(w|...) \propto \left(\prod_{j=1}^{k} w_j^{\delta-1} \right) \left(\prod_{j=1}^{k} w_j^{n_j} \right)$$

$$p(w|...) \propto \prod_{j=1}^{k} w_j^{\delta-1+n_j}$$

$$\Rightarrow w|... \sim \mathcal{D}(\delta + n_1, ..., \delta + n_k)$$

L'étape (b) consiste à modifier les paramètres des lois normales (μ, σ). On a d'après la formule de la loi jointe, pour le paramètre μ :

$$p(\mu|...) \propto \left(\prod_{j=1}^{k} e^{\frac{-(\mu_j-\xi)^2}{2\kappa^{-1}}} \right) \left(\prod_{i=1}^{n} e^{\frac{-(y_i-\mu_{z_i})^2}{2\sigma_{z_i}^2}} \right)$$

$$p(\mu|...) \propto \exp\left[-\frac{1}{2} \sum_{j=1}^{k} \left(\frac{(\mu_j - \xi)^2}{\kappa^{-1}} + \sum_{i:z_i=j} \frac{(y_i - \mu_j)^2}{2\sigma_j^2} \right) \right]$$

$$p(\mu|...) \propto \exp\left\{ -\frac{1}{2} \sum_{j=1}^{k} \left[\mu_j^2 \left(\frac{1}{\kappa^{-1}} + \sum_{i:z_i=j} \frac{1}{\sigma_j^2} \right) - 2\mu_j \left(\frac{\xi}{\kappa^{-1}} + \sum_{i:z_i=j} \frac{y_i}{\sigma_j^2} \right) \right] \right\}$$

$$\Rightarrow \mu_j|... \sim \mathcal{N}\left(\frac{\sigma_j^{-2} \sum_{i/z_i=j} y_i + \kappa\xi}{\sigma_j^{-2} n_j + \kappa}, \frac{1}{\sigma_j^{-2} n_j + \kappa} \right)$$

notée $\mathcal{N}(m, s^2)$, sous la contrainte $\mu_1 < ... < \mu_k$, ce qui signifie en fait que la loi jointe des μ_j s'écrit :

$$\mathbb{I}_{\mu_1<...<\mu_k} k! \prod_{j=1}^{k} \frac{1}{\sqrt{2\pi s^2}} \exp\left(-\frac{1}{2s^2} (\mu_j - m)^2 \right)$$

et pour le paramètre σ :

$$p(\sigma|...) \propto \left(\prod_{j=1}^{k} \Gamma(\alpha) (\sigma_j^{-2})^{\alpha-1} e^{-\beta\sigma_j^{-2}} \right) \left(\prod_{i=1}^{n} \frac{e^{\frac{-(y_i-\mu_{z_i})^2}{2\sigma_{z_i}^2}}}{\sqrt{2\pi\sigma_{z_i}^2}} \right)$$

$$p(\sigma|...) \propto \prod_{j=1}^{k} (\sigma_j^{-2})^{\alpha-1} \left(\frac{1}{\sqrt{2\pi\sigma_j^2}} \right)^{n_j} \exp\left(-\beta\sigma_j^2 - \sigma_j^2 \sum_{i:z_i=j} \frac{(y_i - \mu_j)^2}{2} \right)$$

$$p(\sigma|...) \propto \prod_{j=1}^{k} (\sigma_j^{-2})^{\alpha-1-\frac{n_j}{2}} \exp\left[-\sigma_j^{-2}\left(\beta + \frac{1}{2}\sum_{i:z_i=j}(y_i-\mu_j)^2\right)\right]$$

$$\Rightarrow \sigma_j^{-2}|... \sim \Gamma(\alpha + \frac{1}{2}n_j, \beta + \frac{1}{2}\sum_{i/z_i=j}(y_i-\mu_j)^2)$$

L'étape (c) consiste à modifier les variables d'allocation z. On a :

$$\Rightarrow \mathbb{P}(z_i = j|...) \propto \frac{w_j}{\sigma_j} e^{-\frac{(y_i-\mu_j)^2}{2\sigma_j^2}}$$

Enfin l'étape (d) consiste à modifier l'hyperparamètre β. On a :

$$p(\beta|...) \propto \beta^{g+k\alpha} \exp\left(-h\beta - \beta\sum_{j=1}^{k}\sigma_j^{-2}\right)$$

$$\Rightarrow \beta|... \sim \Gamma(g + k\alpha, h + \sum_{j=1}^{k}\sigma_j^{-2})$$

Examinons maintenant les étapes (e) et (f). Ces étapes sont les deux qui font intervenir la méthode des RJMCMC développée par Green (1995) : il faut ici détailler quelles seront les types de mouvements $q_m, m \in M$ et calculer les probabilités d'acceptation. On peut se demander pourquoi les auteurs ont retenu les deux étapes e et f : une seule de ces deux étapes aurait permis à l'algorithme de modifier le paramètre k. Nous reviendrons sur ce choix dans la suite car depuis, différents auteurs ont proposé des améliorations de la méthode qui jouent sur la présence ou non de ces deux types d'étapes e et f.

L'étape (e) consiste à séparer une composante en deux ou à en regrouper deux en une seule. Donc cette étape ne peut modifier k qu'en lui ajoutant ou lui soustrayant 1. Les types de mouvements possibles seront donc :

$$M = \{1 \leftrightarrow 2, 2 \leftrightarrow 3, ..., k_{max} - 1 \leftrightarrow k_{max}\}$$

Lorsqu'on est en k, seuls les mouvements possibles seront donc $k-1 \leftrightarrow k$ et $k \leftrightarrow k+1$, choisis respectivement avec une probabilité d_k (d pour *death*, "mort" d'une composante) et b_k (b pour *birth*, "naissance" d'une composante). Bien sûr comme $k \in \{1, ..., k_{max}\}$ on a $b_{k_{max}} = d_1 = 0$. Le choix des autres d_k et b_k est arbitraire, les auteurs choisissent : $d_{k_{max}} = b_1 = 1$ et pour $k \in \{2, ..., k_{max} - 1\}, b_k = d_k = \frac{1}{2}$.

Détaillons tout d'abord comment deux composantes sont regroupées (si en $k+1$ on a choisi la modification $k \leftrightarrow k+1$). Deux composantes j_1 et j_2 sont choisies au hasard (de façon uniforme) telles que $\mu_{j_1} < \mu_{j_2}$ et telles qu'il n'y ait pas d'autre μ_j dans l'intervalle $[\mu_{j_1}, \mu_{j_2}]$. Ces deux composantes seront regroupées, en une nouvelle composantes que l'on notera j^*.

Tout d'abord toutes les variables d'allocation sont modifiées : tous les z_i valant j_1 ou j_2 vaudront maintenant j^*. Ceci induit un changement d'indexation des composantes : par exemple si $j_1 = 4$ et $j_2 = 5$ alors $j^* = 4$ et il faut renommer la composante 6 composante 5, la 7 devient 6 et ainsi de suite jusqu'à la $k+1$ qui devient k.

Les auteurs imposent ensuite que w_{j^*}, μ_{j^*} et $\sigma^2_{j^*}$ vérifient :

$$w_{j^*} = w_{j_1} + w_{j_2}$$

$$w_{j^*}\mu_{j^*} = w_{j_1}\mu_{j_1} + w_{j_2}\mu_{j_2}$$

$$w_{j^*}(\mu^2_{j^*} + \sigma^2_{j^*}) = w_{j_1}(\mu^2_{j_1} + \sigma^2_{j_1}) + w_{j_2}(\mu^2_{j_2} + \sigma^2_{j_2})$$

ce qui définit complètement la nouvelle composante.

Le mouvement inverse, c'est-à-dire le passage de k à $k+1$ doit vérifier lui aussi ces équations. Cette fois, comme la dimension du paramètre va augmenter, il est nécessaire de simuler des variables aléatoires comme cela a été décrit dans la présentation générale des RJMCMC. En fait, la dimension augmente de 3 il faut donc simuler trois variables aléatoires indépendantes :

$$u_1 \sim \mathcal{B}e(2,2), u_2 \sim \mathcal{B}e(2,2), u_3 \sim \mathcal{B}e(1,1)$$

où $\mathcal{B}e(p,q)$ avec $p>0$, $q>0$ dénote une loi Beta de densité :

$$be(x|p,q) = \frac{x^{p-1}(1-x)^{q-1}}{\mathcal{B}(p,q)} \mathbb{I}_{]0,1[}(x)$$

où $\mathcal{B}(p,q)$ est la fonction Beta :

$$\mathcal{B}(p,q) = \frac{\Gamma(p)\Gamma(q)}{\Gamma(p+q)}$$

Notons $u = (u_1, u_2, u_3)$. La loi de probabilité q suivant laquelle on génère u, nécessaire au calcul des probabilités d'acceptation dans le cadre des RJMCMC, est donc complètement déterminée. On a sa densité :

$$q(u) = be(u_1|2,2)be(u_2|2,2)be(u_3|1,1)$$

On choisit cette fois une composante j^* au hasard qu'on va séparer en deux : j_1 et j_2 en réindexant les composantes suivantes. On pose :

$$w_{j_1} = u_1 w_{j^*}$$

$$w_{j_2} = (1-u_1) w_{j^*}$$

ainsi la contrainte $w_{j^*} = w_{j_1} + w_{j_2}$ est respectée, puis :

$$\mu_{j_1} = \mu_{j^*} - u_2 \sigma_{j^*} \sqrt{\frac{w_{j_2}}{w_{j_1}}}$$

$$\mu_{j_2} = \mu_{j^*} + u_2 \sigma_{j^*} \sqrt{\frac{w_{j_1}}{w_{j_2}}}$$

ainsi la contrainte $w_{j^*}\mu_{j^*} = w_{j_1}\mu_{j_1} + w_{j_2}\mu_{j_2}$ est respectée, et enfin :

$$\sigma^2_{j_1} = u_3(1-u_2^2)\sigma^2_{j^*}\frac{w_{j^*}}{w_{j_1}}$$

$$\sigma^2_{j_2} = (1-u_3)(1-u_2^2)\sigma^2_{j^*}\frac{w_{j^*}}{w_{j_2}}$$

ainsi la contrainte $w_{j^*}(\mu^2_{j^*} + \sigma^2_{j^*}) = w_{j_1}(\mu^2_{j_1} + \sigma^2_{j_1}) + w_{j_2}(\mu^2_{j_2} + \sigma^2_{j_2})$ est respectée.

Ainsi, la fonction g qui permet de passer des paramètres t et du vecteur aléatoire u dans le modèle à k composantes au paramètre $t' = g(t,u)$ dans le modèle à $k+1$ composantes est entièrement déterminée par les équations précédentes.

La dernière modification à faire pour achever la création des deux composantes est d'actualiser les variables d'allocation z_i pour les i tels que $z_i = j^*$. Elles sont tirées au hasard suivant la loi :

$$\mathbb{P}(z_i = j|...) \propto \frac{w_j}{\sigma_j} e^{-\frac{(y_i-\mu_j)^2}{2\sigma_j^2}}$$

pour $j \in \{j_1, j_2\}$.

Ainsi toutes les modifications possibles dans l'étape (e) ont été définies, reste encore à calculer les probabilités d'acceptation. En fait, on a de façon générique (suivant les notations que l'on avait dans la

partie précédente, où a_m désignait la probabilité d'acceptation d'une modification, et q_m est la loi selon laquelle on propose la modification) :

$$a_m(t,t') = min\left(1, \frac{p(dt')q_m(dt|t')}{p(dt)q_m(dt'|t)}\right)$$

Donc on peut écrire : $a_m(t,t') = \min(1,A)$, avec :

$$A = \frac{p(dt')q_m(dt|t')}{p(dt)q_m(dt'|t)}$$

et donc :

$$a_m(t',t) = min(1, A^{-1})$$

On a développé précédemment comment Green (1995) a obtenu pour les RJMCMC, dans le cas de l'augmentation de la dimension du paramètre (donc ici la séparation de deux composantes) :

$$A = \frac{p(t')j(t')}{p(t)j(t)q(u)}\left|\frac{\partial g(t,u)}{\partial(t,u)}\right|$$

où $j(t)$ est la probabilité de proposer une modification de type m c'est-à-dire $k \leftrightarrow k+1$ lorsqu'on est en t, et est donc donnée par b_k, de même $j(t') = d_k$

Ici on a :

$$\frac{p(t')}{p(t)} = \frac{\mathcal{L}'}{\mathcal{L}}\frac{p(k+1)}{p(k)}(k+1)\frac{w_{j_1}^{\delta-1+l_1}w_{j_2}^{\delta-1+l_2}}{w_{j^*}^{\delta-1+l_1+l_2}\mathcal{B}(\delta,k\delta)}...$$

$$...\sqrt{\frac{\kappa}{2\pi}}\exp\left[-\frac{1}{2}\kappa\left((\mu_{j_1}-\xi)^2+(\mu_{j_2}-\xi)^2-(\mu_{j^*}-\xi)^2\right)\right]...$$

$$...\frac{\beta^\alpha}{\Gamma(\alpha)}\left(\frac{\sigma_{j_1}^2\sigma_{j_2}^2}{\sigma_{j^*}^2}\right)^{-\alpha-1}\exp\left(-\beta(\sigma_{j_1}^{-2}+\sigma_{j_2}^{-2}-\sigma_{j^*}^{-2})\right)$$

où l_1 est le nombre d'observations allouées à j_1, l_2 le nombre d'observations allouées à j_2 et $\frac{\mathcal{L}'}{\mathcal{L}}$ est le ratio de la nouvelle vraisemblance sur l'ancienne, c'est-à-dire si on note respectivement z_i et z_i' les variables d'allocation dans les modèles à $k+1$ et k composantes et de même (μ_i,σ_i) (μ_i',σ_i') les paramètres on a :

$$\frac{\mathcal{L}'}{\mathcal{L}} = \frac{\prod_{i=1}^{n}\frac{\exp\left(\frac{-(y_i-\mu'_{z'_i})^2}{2\sigma'^2_{z'_i}}\right)}{\sqrt{2\pi\sigma'^2_{z'_i}}}}{\prod_{i=1}^{n}\frac{\exp\left(\frac{-(y_i-\mu_{z_i})^2}{2\sigma^2_{z_i}}\right)}{\sqrt{2\pi\sigma^2_{z_i}}}} = \frac{\prod_{i:z'_i=j_1,j_2}\frac{\exp\left(\frac{-(y_i-\mu'_{z'_i})^2}{2\sigma'^2_{z'_i}}\right)}{\sqrt{2\pi\sigma'^2_{z'_i}}}}{\prod_{i:z_i=j^*}^{n}\frac{\exp\left(\frac{-(y_i-\mu_{j^*})^2}{2\sigma^2_{j^*}}\right)}{\sqrt{2\pi\sigma^2_{j^*}}}}$$

et :

$$\frac{j(t')}{j(t)q(u)} = \frac{d_{k+1}}{b_k P_{alloc} be(u_1|2,2)be(u_2|2,2)be(u_3|1,1)}$$

où P_{alloc} est la probabilité de l'allocation entre j_1 et j_2 des variables regroupées sous j^*, c'est-à-dire que :

$$P_{alloc} = \prod_{i:z_i=j^*}\left(\frac{\frac{w_{z'_i}}{\sigma_{z'_i}}\exp\left(-\frac{(y_i-\mu_{z'_i})^2}{2\sigma^2_{z'_i}}\right)}{\frac{w_{j_1}}{\sigma_{j_1}}\exp\left(-\frac{(y_i-\mu_{j_1})^2}{2\sigma^2_{j_1}}\right)+\frac{w_{j_2}}{\sigma_{j_2}}\exp\left(-\frac{(y_i-\mu_{j_2})^2}{2\sigma^2_{j_2}}\right)}\right)$$

et :

$$\left|\frac{\partial g(\xi_2)}{\partial(\xi_1, u^{(1)})}\right| = \frac{w_{j^*}|\mu_{j_1} - \mu_{j_2}|\sigma_{j_1}^2\sigma_{j_2}^2}{u_2(1-u_2^2)u_3(1-u_3)\sigma_{j^*}}$$

On obtient donc une expression exacte de A. Enfin, pour le mouvement en sens inverse (regroupement de deux composantes) la probabilité d'acceptation est alors :

$$min(1, A^{-1})$$

L'étape (f) enfin consiste à créer ou supprimer une composante vide. Ceci est en fait plus simple à décrire. Ici, l'ensemble des modifications possibles est choisi similaire au précédent :

$$M = \{1 \leftrightarrow 2, 2 \leftrightarrow 3, ..., k_{max} - 1 \leftrightarrow k_{max}\}$$

En k, le mouvement $k \leftrightarrow k+1$ (création d'une composante vide) est choisi avec la probabilité b_k et le mouvement $k-1 \leftrightarrow k$ (suppression d'une composante vide) avec la probabilité d_k, les b_k et d_k étant les mêmes que dans l'étape e.

Pour la création d'une composante, on introduit donc j^* avec $w_{j^*} \sim \mathcal{B}e(1,k)$, $\mu_{j^*} \sim \mathcal{N}(\xi, \kappa^{-1})$ et $\sigma_{j^*}^{-2} \sim \Gamma(\alpha, \beta)$. Enfin, toutes les variables de poids doivent être renormalisées de façon à ce que leur somme soit 1, c'est-à-dire qu'on remplace w_j par $w_j(1-w_{j^*})$ pour tout $j \neq j^*$.

Pour le mouvement inverse, on choisit au hasard une composante vide j^* (si il y en a une) de façon uniforme et on l'efface, il suffit de renormaliser les poids en remplaçant w_j par $\frac{w_j}{\sum_{j' \neq j^*} w_{j'}}$.

La probabilité d'acceptation est toujours déterminée par $a_m(t,t') = \min(1, A)$ où

$$A = \frac{p(k+1)}{p(k)} \frac{1}{\mathcal{B}(k\delta, \delta)} w_{j^*}^{\delta-1}(1-w_{j^*})^{n+k\delta-k}(k+1)...$$

$$...\frac{d_{k+1}}{(k_0+1)b_k}\left[\frac{1}{be(w_{j^*}|1,k)}(1-w_{j^*})^{k-1}\right]$$

pour la création d'une composante et $\min(1, A^{-1})$ pour la destruction. Dans cette expression la première ligne correspond au rapport des lois jointes des observations et paramètres. On constate que la vraisemblance des observations n'apparaît pas du tout : en effet elle n'est pas modifiée dans un mouvement de ce type. La seconde ligne contient le jacobien de la transformation (entre crochets) et le rapport du choix du type de modification particulier retenu : $\frac{d_{k+1}}{(k_0+1)b_k}$ où k_0 est le nombre de composantes vides avant la création d'une nouvelle [1].

Ainsi tous les types de modifications sont clairement définis. Pour vérifier la validité de l'algorithme, les auteurs concluent en remarquant que la chaîne simulée est clairement irréductible et apériodique.

2.4 Les résultats obtenus

La fin de l'article de Richardson et Green est consacrée à la mise en pratique de cet algorithme et à des propositions d'améliorations de la méthode. Rappelons notamment que les propositions de modifications des paramètres dans les étapes (e) et (f) n'ont fait l'objet d'aucune optimisation selon les auteurs : ainsi les performances de l'algorithme devraient donc pouvoir être améliorées. L'article est suivi d'une discussion dans lesquelles d'autres propositions ou critiques sont faites, certaines ayant trouvé des prolongements dans d'autres articles par la suite. Ces commentaires et propositions seront l'objet de la partie 3. La fin de la partie 2 est consacrée à la présentation des résultats obtenus par Richardson et Green.

[1] Remarquons que dans l'article original de Richardson et Green, la puissance du dernier $(1-w_{j^*})$ n'est pas $k-1$ mais k, il s'agit d'une erreur de calcul repérée par Tobias Rydén que les auteurs ont corrigé sur leur site web peu de temps après la parution de l'article.

2.4.1 Les données

L'algorithme proposé a donc été appliqué à trois ensembles de données différents :

- le premier concerne des mesures de l'activité enzymatique dans le sang d'un groupe de 245 individus. L'enzyme étudiée étant impliquée dans le métabolisme de substances carcinogéniques, la classification est ici un objectif important car elle peut permettre de détecter différents groupes, génétiquement différents, dans la population. On a pour cet ensemble de données : $R = 2.86$ et $\xi = 0.45$, donc $\kappa = 0.122$ et $h = 1.22$.

- le second concerne la mesure de l'acidité dans 155 lacs du Wisconsin. On a ici : $R = 4.18$ et $\xi = 5.02$, donc $\kappa = 0.057$ et $h = 0.573$.

- le dernier, dit "galaxy data set", contient la mesure de vélocité de 82 galaxies. On a ici : $R = 25.11$ et $\xi = 21.73$, donc $\kappa = 0.0016$ et $h = 0.016$. Le "galaxy data set" figure en annexe.

Ces trois jeux de données avaient déjà été étudiés par d'autres auteurs dans le cadre de modèle de mélanges, en particulier le "galaxy data set" (au moins trois autres articles). Ceci permet une première comparaison des résultats obtenus avec des méthodes plus classiques.

2.4.2 Interprétation des résultats

Des simulations selon la loi *a posteriori* ont été menées dans chacun des trois cas ci-dessus pas les auteurs. Deux remarques sont à faire :

- les auteurs ont réalisé 200000 itérations de l'algorithme décrit ci-dessus (ou 200000 *sweeps*), dont les 100000 premières on été considérées comme *burn-in period*. Autrement dit, $n_1 = 100000$ et $n_2 = 200000$. Aucun critère quantitatif n'est proposé pour le choix de ces valeurs, Richardson et Green affirment d'ailleurs qu'elles sont vraisemblablement supérieures à ce qui est nécessaire.

- dans les trois cas, le nombre de composantes n'a jamais dépassé 24, donc le choix de $k_{max} = 30$ n'a été d'aucune conséquence.

Pour ce qui est du paramètre k, l'interprétation des résultats est extrêmement simple puisqu'on obtient les probabilités *a posteriori* de chaque valeur possible.

Par exemple, pour le premier jeu de données (enzymes), on a :

p(k=...|y)	Valeur obtenue
1	0.000
2	0.023
3	0.290
4	0.317
5	0.206
6	0.095
7	0.041
8	0.017
9	0.007
≥ 10	0.003

On peut représenter ces probabilités sur un histogramme, ce qui est fait par exemple dans Green (1995) (cf. figure 3).

Ainsi, le nombre de composantes est très vraisemblablement 3, 4 ou 5 (avec probabilité 0.813), la valeur de k la plus probable est 4. Toutes les valeurs nécessaires pour le "choix de modèle" peuvent être calculées à partir de ces valeurs, par exemple les facteurs de Bayes :

$$\mathcal{B}_{k_1,k_2} = \frac{p(k=k_1|y)}{p(k=k_2|y)} \frac{p(k=k_2)}{p(k=k_1)}$$

Pour ce qui est de l'estimation de la densité obtenue, les auteurs proposent plusieurs approches :

- on peut choisir, à l'aide des valeurs de $p(k|y)$, une valeur de k particulière k_0. Ensuite, on ne considère plus que les étapes de la simulation où $k = k_0$. On retient alors l'estimateur :

$$\widehat{f}(y_i|k=k_0,\widehat{\theta},\widehat{w}) = \sum_{j=1}^{k_0} \widehat{w}_j f(.|\widehat{\theta}_j)$$

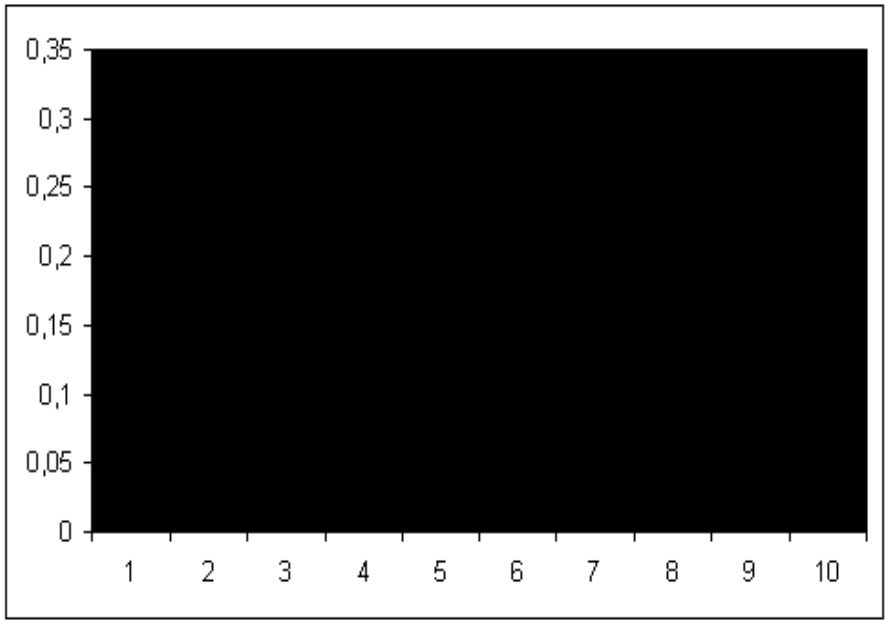

FIG. 3 – Loi *a posteriori* de k.

où $\widehat{w}$ et $\widehat{\theta}$ sont des résumés des lois *a posteriori* des poids et des paramètres, par exemple les moyennes des valeurs pour lesquelles $k = k_0$:

$$\widehat{\theta} = \frac{\sum_{m=n_1}^{n_2} \mathbb{I}_{k^m=k_0} \theta^m}{\sum_{m=n_1}^{n_2} \mathbb{I}_{k^m=k_0}}$$

Le problème est que, comme on le verra par la suite, les densités *a posteriori* sont souvent multimodales et les résultats obtenus par cette méthode ne sont pas très satisfaisants.

- une deuxième approche consiste, toujours à $k = k_0$ fixé, à estimer :

$$\mathbb{E}[f(.|k, w, \theta)|k = k_0, y]$$

en calculant l'estimateur :

$$\widetilde{f}(.|k = k_0, \theta, w) = \frac{\sum_{m=n_1}^{n_2} \mathbb{I}_{k^m=k_0} \sum_{j=1}^{k_0} w_j^m f(.|\theta_j^m)}{\sum_{m=n_1}^{n_2} \mathbb{I}_{k^m=k_0}}$$

- enfin la dernière proposition de Richardson et Green consiste à sortir de l'optique paramétrique pour se concentrer sur l'estimation de la densité, en estimant :

$$\mathbb{E}[f(.|k, w, \theta)|y]$$

autrement dit, on n'impose plus une valeur fixe de k. L'estimateur proposé est le suivant :

$$\widetilde{f}(.|k, \theta, w) = \frac{\sum_{k_0=0}^{k_{max}} \widetilde{f}(.|k = k_0, \theta, w) card\{l \in \{n_1, ..., n_2\} : k^l = k_0\}}{n_2 - n_1}$$

L'inconvénient des deux dernières méthodes est que la quantité approchée, par exemple $\mathbb{E}[f(.|k,w,\theta)|k = k_0, y]$, n'est pas elle-même la densité d'un mélange.

Une proposition pour remédier à cela est faite par Christian Robert dans la discussion de l'article de Richardson et Green : il propose de n'utiliser la méthode des RJMCMC que dans le but de choisir k (donc comme outil de sélection de modèles), puis une fois ce choix fait d'utiliser l'algorithme de Gibbs à $k = k_0$ fixé, comme cela a été décrit dans la première partie, pour étudier les paramètres.

2.4.3 Sensibilité au choix des lois *a priori*

Richardson et Green étudient ensuite en détail la dépendance des résultats obtenus aux lois *a priori* choisies.

Ils étudient tout d'abord la sensibilité de la loi *a posteriori* de k aux lois *a priori*. Ils refont pour cela les mêmes simulations en prenant différentes valeurs pour les hyperparamètres α, β, g, h et comparent es résultats obtenus. La loi *a posteriori* de k s'étant avérée très sensible à a valeur du ration α/β, les auteurs ont choisi comme on l'a vu un modèle hiérarchique avec β lui-même aléatoire. Une fois ce choix fait, la loi *a posteriori* de k s'est avérée assez insensible aux choix des valeurs particulières de α, g et h, tant que la quantité $\sqrt{\frac{g}{h\alpha}}$ (ordre de grandeur des σ_j) restait comprise entre $\frac{R}{5}$ et $\frac{R}{20}$. Les choix pour les simulations "définitives" ont été faits de façon à ce que cette valeur soit $\frac{R}{10}$. La loi *a priori* sur les μ_j, $\mathcal{N}(\xi, \kappa^{-1})$ a une légère influence sur la loi *a posteriori* de k. Les auteurs observent qu'en réduisant légèrement la variance κ^{-1} on favorise un plus grand nombre de composantes car on accepte plus facilement deux composantes avec des moyennes très proches. Mais si κ^{-1} devient trop petit, on finit par diminuer le nombre de composantes car on refuse la présence de composantes dont la moyenne serait située près des valeurs extrêmes des données.

C'est ensuite la dépendance de la loi *a posteriori* des paramètres (σ^2, μ) aux lois *a priori* qui est examinée par les auteurs. Tout d'abord, ils constatent que lorsque κ^{-1} diminue, les moyennes μ ont tendance à être moins dispersées, mais ils remarquent aussi que cet effet est beaucoup plus sensible lorsque k est grand. Les σ_j^2 sont, comme k, particulièrement sensibles aux valeurs de $\frac{\beta}{\alpha}$, là aussi particulièrement lorsque k est grand. En revanche, une fois β rendu aléatoire, les résultats obtenus sur σ varient peu avec les valeurs de α, g et h.

2.4.4 Applications à la classification bayésienne

Comme on l'a vu en introduction, un des intérêts des modèles de mélange est que les résultats obtenus peuvent être utilisés pour faire de la classification, c'est-à-dire pour rattacher les individus à un groupe particulier, ici à une composante particulière.

Ceci est examiné par Richardson et Green à la fin de l'article *On Bayesian analysis of mixtures with an unknown number of components* (1997).

On peut bien sûr classer sans grande difficulté les observations y_i pour $i = 1, ..., n$: on dispose pour cela, grâce à la simulation MCMC, d'une approximation de

$$\mathbb{P}(z_i = j|y, k = k_0)$$

une fois que l'on a fixé le nombre de classes $k = k_0$. Le problème soulevé par les auteurs est que lorsqu'on est en présence d'une nouvelle observation, disons y^*, on ne pourra obtenir une approximation de :

$$\mathbb{P}(z^* = j|y, y^*, k = k_0)$$

qu'en recommençant complètement les simulations sur ce nouvel échantillon ! Ils proposent alors l'approximation (très intuitive) :

$$\mathbb{P}(z^* = j|y, y^*, k = k_0) = \int\int \mathbb{P}(z^* = j|y, y^*, k = k_0, \theta, w)p(\theta, w|y, y^*, k = k_0)d\theta dw$$

$$... = \int\int \mathbb{P}(z^* = j|y^*, k = k_0, \theta, w)p(\theta, w|y, y^*, k = k_0)d\theta dw$$

$$... \approx \int\int \mathbb{P}(z^* = j|y^*, k = k_0, \theta, w) p(\theta, w|y, k = k_0) d\theta dw$$

On peut estimer cette dernière intégrale par la quantité :

$$\frac{\frac{w_j}{\sqrt{2\pi\sigma_j^2}} \exp\left(-\frac{(y^* - \mu_j)^2}{2\sigma_j^2}\right)}{\sum_{J=1}^{k_0} \frac{w_J}{\sqrt{2\pi\sigma_J^2}} \exp\left(-\frac{(y^* - \mu_J)^2}{2\sigma_J^2}\right)}$$

On classe enfin les observations dans la classe à laquelle leur probabilité d'appartenance est la plus forte :

$$\widehat{z_i} = \arg\max_j \mathbb{P}(z_i = j|y, k = k_0)$$

$$\widehat{z^*} = \arg\max_j \mathbb{P}(z_i = j|y, y^*, k = k_0)$$

3 Critiques et extensions

Dans la dernière partie de l'article, les auteurs proposent quatre voies pour prolonger, améliorer ou approfondir leur travail :

- la première concerne les performances du simulateur. La question est de savoir si les types de mouvements retenus sont optimaux.

- la deuxième concerne la présentation et l'interprétation des lois *a posteriori*. Pour les auteurs, de nouvelles voies devraient être explorées, en particulier l'étude de lois *a posteriori* jointes de différents paramètres.

- la troisième concerne les lois *a priori*. En particulier, il y a des cas où il pourrait être intéressant de ne pas choisir des μ_j indépendants *a priori*.

- enfin la quatrième concerne le modèle lui-même. Tout les auteurs n'ont analysé que des mélanges de lois normales. Pour eux, il faudrait étudier d'autres distributions. En particulier, dans certains cas une densité unimodale peut-être très bien approchée par un mélange de deux lois normales avec des moyennes très proches, ou par une seule loi plus "plate" que la loi normale. De même, on peut vouloir imposer de nouvelles contraintes sur les paramètres (par exemple, certains modèles de mélange en génétique imposent des contraintes sur les poids).

Dans la discussion et des articles parus depuis, certains de ces problèmes sont abordés : les performances du simulateur et la présence de lois non normales en particulier. En revanche, d'autres points sont critiqués dans la discussion, en particulier le problème d'échange des labels des paramètres au cours des simulations (déjà abordé pour le cas où k est fixe dans la première partie), l'absence d'un critère quantitatif pour déterminer quand la convergence de la méthode est effectuée (valeurs de n_1 et n_2), ainsi que tout simplement le choix de méthodes bayésiennes dans un tel cadre. On va détailler maintenant chacun de ces points.

3.1 Performances du simulateur

3.1.1 Commentaires des auteurs sur les résultats obtenus

Les auteurs examinent plusieurs critères. Tout d'abord, le simulateur a-t'il bien exploré toutes les valeurs probables de k ? Pour ceci, il peut être intéressant de regarder quelle est la proportion de mouvements de types (e) (séparation ou regroupement de deux composantes) qui ont été acceptés : elle varie entre 8% et 14% selon les cas.

La présence de composantes vides pourrait biaiser l'inférence sur k (modifications de type f), ceci n'est pas le cas selon les auteurs car au cours des simulations de nombre moyen de composantes vides est resté assez faible (selon les cas entre 0.1 et 0.57).

Le choix des valeurs initiales n'a que peu d'influence : les auteurs ont commencé toutes les simulations avec $k = 1$, la première séparation de composantes a toujours été acceptée en moins de 5 étapes. De même, en commençant avec $k = 20$ il fallait moins de 100 étapes pour descendre à $k = 10$.

Les auteurs remarquent cependant que l'inférence sur k, notamment le calcul de facteurs de Bayes, n'a de sens que pour les valeurs de k qui ont été visitées assez souvent par la chaîne de Markov simulée.

Pour ce qui est des valeurs des paramètres, les auteurs remarquent que leur simulateur pourrait même avoir des performances supérieures à une MCMC fonctionnant à k fixé. Ils utilisent pour ceci un exemple simple avec 200 "observations" :

$$\forall i \in \{1, ..., 50\}, x_i \sim \mathcal{N}(2.5, 1)$$

$$\forall i \in \{51, ..., 100\}, x_i \sim \mathcal{N}(4, 1)$$

$$\forall i \in \{101, ..., 150\}, x_i = -x_{i-100}$$

$$\forall i \in \{151, ..., 200\}, x_i = -x_{i-100}$$

et font "comme si" $(x_1, ..., x_{200}$ était un échantillon de variables indépendantes et identiquement distribuées provenant d'un mélange de distributions normales. Ils étudient alors le cas (volontairement mal spécifié) où $k = 3$, par deux méthodes différentes :

- par MCMC avec $k = 3$ fixe.
- par RJMCMC, en ne retenant que les étapes où $k = 3$.

Il y a en réalité quatre composantes dans ce "mélange". L'algorithme simple, une fois regroupée deux composantes, en fait les deux de moyenne -4 et -2.5 ou les deux de moyennes 2.5 et 4, a énormément de mal à changer de configuration. En revanche, l'algorithme utilisant les RJMCMC change fréquemment de configuration puisqu'il peut fréquemment passer à 4 composantes et ensuite revenir à 3 en regroupant deux composantes qui ne seront pas toujours les mêmes.

3.1.2 Une proposition d'amélioration de la méthode

Brooks, Giudici et Roberts (2000) remarquent que les simulateurs MCMC ont de très bonnes performances et que ce qui peut ralentir la convergence des méthodes utilisant les RJMCMC est la trop grande proportion de rejet des propositions de mouvements où k varie. Leur article étudie donc de nouvelles propositions de modifications dont le but est d'augmenter cette proportion.

La probabilité d'acceptation étant de la forme

$$a_m(t, t') = min(1, A)$$

$$a_m(t', t) = min(1, A^{-1})$$

où

$$A = \frac{p(t')j(t')}{p(t)j(t)q(u)} \left| \frac{\partial t'}{\partial(t, u)} \right|$$

il peut être judicieux de chercher des propositions de modifications pour lesquelles A est le plus proche possible de 1. Les auteurs proposent différentes méthodes visant à augmenter la proportion de "sauts" d'une valeur de k à une autre acceptés. Les résultats obtenus améliorent sensiblement les performances des simulateurs.

3.1.3 Une approche "concurrente" : méthode par diffusion avec sauts

Un algorithme différent a été proposé par David Phillips et Adrian Smith (1996) pour résoudre par une approche bayésienne le problème de la détermination du nombre de composantes dans un mélange. Il repose que des propositions déjà faites par d'autres auteurs, notamment Grenander et Miller (1994). Notons que l'idée quand à l'inférence est exactement la même que celle de Richardson et Green, seule la façon de simuler dans la loi *a posteriori* diffère.

On présente rapidement en 3.1.3 leur méthode, puis une généralisation proposée par M. Stephens (2000), et enfin en 3.1.4 la comparaison entre ces méthodes et les RJMCMC menée par Olivier Cappé, Christian Robert et Tobias Rydén (2003).

Présentation de la méthode de Phillips et Smith Tout d'abord, comme pour les RJMCMC la méthode proposée ne se limite pas aux modèles de mélanges mais peut être utilisée pour l'inférence bayésienne dans n'importe quel cas où la dimension même du paramètre à estimer est inconnue. Des exemples similaires à ceux utilisés par Green (1995) sont présentés : détection d'un nombre inconnu d'objets sur une image, détection d'un nombre inconnu de changements de tendance sur une série temporelle, choix du nombre de regresseurs dans un modèle linéaire, et l'estimation des paramètres d'un modèle de mélange avec un nombre inconnu de composantes.

Le modèle Le modèle retenu est pratiquement le même que celui utilisé par Richardson et Green (1997), avec quelques simplifications. La seule différence repose sur la façon dont est simulé un échantillon suivant la loi *a posteriori* : au lieu d'utiliser une chaîne de Markov, qui est un processus à temps discret, Phillips et Smith simulent un processus à temps continu dont la loi stationnaire est la loi *a posteriori* recherchée : une diffusion avec sauts, les sauts correspondant au changement du nombre de composantes. Une autre différence notable avec Richardson et Green est que Smith et Phillips n'introduisent pas les variables indicatrices de la classe des observations, z_i.

Le modèle est le suivant :

$$y_i|k,\theta,w \sim \sum_{j=1}^{k} w_j f(.|\theta_j) = f(y_i|k,\theta,w)$$

où $\theta_j = (\mu_j, \sigma_j^2)$ et $f(.|\theta_j)$ est la densité d'une loi $\mathcal{N}(\mu_j, \sigma_j^2)$,

$$\mu_j \sim \mathcal{N}(\mu, \tau^2)$$

avec la contrainte $\mu_1 < ... < \mu_k$,

$$\sigma_j^2 \sim \mathcal{IG}(\gamma, \delta)$$

$$(w_1, ..., w_k)|k \sim \mathcal{D}(1, ..., 1)$$

c'est-à-dire une loi uniforme sur le simplexe de $\mathbb{R}^k$. Enfin, Phillips et Smith désiraient faire le choix habituel $k \sim \mathcal{P}(\lambda)$. En fait, comme $k = 0$ est interdit, ils retiennent :

$$\forall k' \geq 1, \mathbb{P}(k = k') = \frac{\lambda^{k'}}{(e^{\lambda} - 1)k'!}$$

Contrairement à Richardson et Green, Phillips et Smith n'introduisent pas un niveau hiérarchique supplémentaire, ici μ, τ, γ, δ et λ sont constantes.

Enfin on note :

$$\theta = (\theta_1, ..., \theta_k) \in \Theta_k$$

Le simulateur Les auteurs cherchent donc à simuler selon la loi *a posteriori* $p(\theta, k|y)$ notée pour simplifier $\pi(\theta, k)$. Ils utilisent pour ceci une diffusion pouvant sauter d'un espace Θ_k dans un autre. On la notera $\{(\theta, k)^{(t)}, t \in \mathbb{R}_+\}$.

On étudie tout d'abord la dynamique des sauts. Aux questions déjà soulevées par Green s'ajoutent de nouvelles dues au temps continu : quand proposer des sauts ?

Si à la date t on est en (θ, k), la probabilité de quitter Θ_k est donnée par *l'intensité de saut* $q((\theta, k), (\phi, h))$ où $h \neq k$. En fait, la probabilité (infinitésimale) de sauter entre les dates t et $t + dt$ de (θ, k) vers un nouvel état inclus dans $A \subset \Theta_h$ est donnée par :

$$dt \int_A q((\theta, k), (\phi, h))d\phi$$

Les auteurs notent $\mathcal{J}(\theta, k)$ l'ensemble des (ϕ, h) qui peuvent être atteint en un seul saut, appelé "ensemble de saut". Ces ensembles doivent être "réversibles", c'est-à-dire que :

$$(\phi, h) \in \mathcal{J}(\theta, k) \Rightarrow (\theta, k) \in \mathcal{J}(\phi, h)$$

Pour décider quand aura lieu un saut, il faut calculer l'intensité marginale de saut :

$$r(\theta, k) = \sum_h \int_{(\phi,h)\in\mathcal{J}(\theta,k)} q((\theta, k), (\phi, h))d\phi$$

Ainsi, la probabilité (infinitésimale) de quitter Θ_k entre les dates t et $t + dt$ de (θ, k) est donnée par :

$$r(\theta, k)dt$$

Les temps de sauts sont simulés en générant aléatoirement $e_1, e_2, ...$ indépendantes selon une loi exponentielle de paramètre 1, $\mathcal{E}(1)$, et en choisissant t_i date du i-ème saut telle que :

$$\int_{t_{i-1}}^{t_i} r((\theta, k)^{(t)})dt \geq e_i$$

avec $t_0 = 0$ par convention ($\geq$ et non pas $=$ car on ne pourra bien sûr pas simuler directement un processus à temps continu mais une version discrétisée de ce processus, le pas de temps étant Δ).

Si juste avant la date t_i on est en (θ, k), il faut choisir le nouvel état suivant un noyau de transition Q à déterminer.

Deux propositions sont faites pour Q par les auteurs, l'une s'inspirant de l'algorithme de Gibbs et l'autre reposant sur celui de Hastings-Metropolis. Seule cette deuxième solution est appliquée aux modèles de mélanges :

$$q((\theta, k), (\phi, h)) = Cp_h e^{P_h(\phi)}$$

où $p_k = \frac{\lambda^k}{(e^\lambda - 1)k!}$ est la probabilité *a priori* de k, et $P_h(\phi)$ est le logarithme de la densité *a priori* de $\phi = (\theta, \sigma^2)$ sachant que l'on a h composantes :

$$P_h(\phi) = \prod_{j=1}^{h} \left[\left(\frac{1}{\sqrt{2\pi\tau^2}} e^{-\frac{(\mu_j - \mu)^2}{2\tau^2}} \right) \left(\frac{\delta^\gamma}{\Gamma(\gamma)} \frac{1}{(\sigma_j^2)^{\gamma+1}} e^{-\frac{\delta}{\sigma_j^2}} \right) \right]$$

A chaque date de saut, on simule donc (ϕ, h) suivant la loi *a priori* restreinte à $\mathcal{J}(\theta, k)$. On accepte ensuite (ϕ, h) avec probabilité :

$$\min(1, e^{L_h(\phi) - L_k(\theta)})$$

où L est la vraisemblance, c'est-à-dire que :

$$L_h(\phi) = \prod_{i=1}^{n} \sum_{j=1}^{h} w_j f(y_i|\theta_j)$$

Sinon, on rejette le saut c'est-à-dire qu'on reste à l'état (θ, k). On obtient :

$$Q((\theta,k),(\phi,h)) = \frac{C\min(1, e^{L_h(\phi)-L_k(\theta)}p_h e^{P_h(\phi)})}{r(\theta,k)}$$

Décrivons maintenant comment est réalisé un tel saut. Un seul type de saut est envisagé ici : la création ou la suppression d'une composante, c'est-à-dire que ce saut est proche de l'étape (f) de l'algorithme de Richardson et Green. Il n'y a pas ici de possibilité de regrouper ou séparer deux composantes (étape (e) chez Richardson et Green).

Pour la création d'une composante, tout d'abord on génère $\mu^* \sim \mathcal{N}(\mu, \tau^2)$ puis $\sigma^{*2} \sim \mathcal{IG}(\gamma, \delta)$. Reste à définir la probabilité w^* associé. Si $\mu_j < \mu^* < \mu_{j+1}$ on simule un nouveau triplé (w_j, w^*, w_{j+1}) suivant sa loi conditionnelle aux autres $w_{j'}$ fixés. Si $\mu^* < \mu_1$ on simule un nouveau couple (w^*, w_1) suivant sa loi conditionnelle aux autres $w_{j'}$, de même si $\mu^* > \mu_k$. On réindexe ensuite les différentes composantes. Le calcul explicite de $r(\theta, k)$ donne finalement la probabilité d'acceptation de ce saut.

Pour la destruction d'une composante, on choisit la composante j à effacer (de façon uniforme), on simule un nouveau couple (w_{j-1}, w_{j+1}) suivant sa loi conditionnelle aux autres $w_{j'}$ fixés, on réindexe ensuite les différentes composantes et on calcule la probabilité d'acceptation.

Reste donc à déterminer la dynamique entre les sauts. Elle est donnée par l'équation différentielle stochastique : $\forall t \in]t_{i-1}, t_i[$,

$$d\theta^{(t)} = \frac{dt}{2}\left[\frac{d}{d\theta}(L_k(\theta^{(t)})) + \frac{d}{d\theta}(P_k(\theta^{(t)}))\right] + dW_t^k$$

où $\{W_t^k, t \geq 0\}$ est un mouvement brownien de dimension $dim(\Theta_k)$. Bien sûr, comme on ne simule pas directement le processus à temps continu mais une version discrétisée (avec pas de temps Δ), on utilisera l'équation discrétisée :

$$\theta^{(t+\Delta)} = \theta^{(t)} + \frac{\Delta}{2}\left[\frac{d}{d\theta}(L_k(\theta^{(t)})) + \frac{d}{d\theta}(P_k(\theta^{(t)}))\right] + \sqrt{\Delta}z_k^{(t)}$$

où les $z_k^{(t)}$ sont indépentants et identiquement distribués selon $\mathcal{N}(0,1)$.

Les résultats Les auteurs utilisent appliquent ensuite leur méthode sur deux jeux de données, dont le *galaxy data set* utilisé par Richardson et Green.

On peut comparer ici les résultats des deux méthodes, tout en remarquant que ces comparaisons n'ont que peu de sens puisque les lois simulées ici ne sont pas les mêmes : les deux articles utilisaient en effet des lois *a priori* différentes donc les lois *a posteriori* ne peuvent être les mêmes. En particulier, Phillips et Smith n'ont pas introduit un niveau hiérarchique supplémentaire comme Richardson et Green. Leurs résultats dépendent donc plus fortement des lois *a priori* (Phillips et Smith ont retenu ici : $\mu = 20$, $\tau^2 = 100$, $\delta = 0.5$ et $\lambda = 3$).

p(k=...|y)	Richardson & Green	Phillips & Smith
3	0.061	-
4	0.128	-
5	0.182	-
6	0.199	0.025
7	0.160	0.394
8	0.109	0.321
9	0.071	0.217
10	0.040	0.043

Les auteurs reconnaissent en fait dès le début de l'article qu'aucune comparaison théorique de leur méthode avec les RJMCMC n'avait pas été menée lors de la parution de leur article.

Contribution de Stephens Comme on l'a vu, Phillips et Smith traitent en fait de nombreux exemples dans leur article et ne passent que peu de temps sur celui des mélanges de distributions. Stephens (2000a) propose une description complète de l'adaptation de la méthode aux modèles de mélanges.

Sa méthode est légèrement différente puisqu'il choisit une autre dynamique de saut : il en résulte que les sauts ne sont jamais refusés. En revanche, si le processus saute dans un état très improbable il n'y restera que très peu de temps avant de sauter de nouveau.

Pour Stephens, ceci est un point positif, car en visitant même les zones de faible probabilité de la loi *a posteriori*, son simulateur aurait de meilleures performances que celui de Green et Richardson.

3.1.4 Comparaison simulateurs à temps continu / RJMCMC

Cappé, Robert et Rydén (2003) ont comparé les différents types de simulateurs. Ils remarquent tout d'abord que dans l'approche de Green et Richardson par RJMCMC, deux types de sauts sont proposés :

- regrouper ou séparer des composantes (type (e)),
- créer ou détruire des composantes (type (f)),

alors que les simulateurs à temps continu, qu'il surnomment BDMCMC (*birth and death* MCMC), de Phillips et Smith et de Stephens n'utilisent que des créations ou destructions de composantes. Remarquant que ce choix est arbitraire ils créent des simulateurs plus généraux à temps continu, qu'ils appellent CTMCMC (*continuous time* MCMC) utilisant aussi des regroupements et séparations de composantes.

La partie théorique de leurs comparaisons montre qu'en fait, les CTMCMC ne sont pas "beaucoup plus générales" que les RJMCMC puisque, sous certaines conditions, une CTMCMC peut être approchée par une suite de RJMCMC dont le pas de temps tendrait vers 0.

La partie empirique compare quatre simulateurs (RJMCMC avec sauts de type (f) seulement, de type (e) et (f), BDMCMC et RJMCMC) et montre que si les résultats sont comparables, le classement obtenu est le suivant :

- le meilleur simulateur serait la RJMCMC avec sauts de type (f) seulement,
- presque au même niveau, BDMCMC,
- moins bons : RJMCMC avec sauts de type (e) et (f) et CTMCMC, cette dernière méthode étant en plus très coûteuse en temps de calcul.

3.2 Problème des échanges des labels des paramètres

Pour beaucoup d'auteurs ayant contribué à la discussion, il s'agit la du principal problème de cette méthode. Matthew Stephens utilise un exemple simple pour montrer que la contrainte $\mu_1 < ... < \mu_k$ n'est pas suffisante : il considère le mélange

$$0.33\mathcal{N}(0,1) + 0.33\mathcal{N}(5,1) + 0.34\mathcal{N}(5,4)$$

et utilise un algorithme de Gibbs simple pour obtenir un échantillon de la loi *a posteriori*. Ses résultats montrent clairement ce que l'on pouvait attendre : la contrainte $\mu_1 < \mu_2 < \mu_3$ ne sépare pas correctement les deux composantes ayant la même moyenne, et les labels de ces deux composantes sont échangés au cours des simulations. Ceci est problématique car la méthode retenue ne fonctionne donc pas correctement sur l'exemple du "chapeau méxicain" (figure 2, dans la première partie). Dans le cas du "chapeau méxicain" on peut résoudre ce problème en prenant $\sigma_1^2 < \sigma_2^2 < \sigma_3^2$ comme contrainte à la place de celle sur les composantes. Mais ceci ne résout pas le problème dans le cas de l'exemple proposé par Stephens, où deux composantes ont la même variance.

Gilles Celeux remarque que la contrainte $\mu_1 < ... < \mu_k$ modifie la forme de la loi *a posteriori* des moyennes et peut biaiser leur estimation, en particulier dans le cas où deux composantes auraient des moyennes très proches. Pour lui, il vaudrait mieux ne pas imposer de contraintes pendant les simulations puis de permuter ensuite les résultats des MCMC à l'aide d'un algorithme à définir. Cependant, il remarque aussi que cette méthode pourrait être difficile à implémenter dans le cas où k est inconnu. Il affirme en attendant que le problème de l'échange du label des composantes est une des difficultés de l'analyse bayésienne des modèles de mélanges qui n'avait lors de la parution de l'article de Richardson et Green pas trouvé de solution satisfaisante.

La proposition d'un algorithme permettant de permuter après les simulations a été étudiée depuis par exemple par Celeux, Hurn et Robert (1997) ou Stephens (2000b) qui obtiennent des résultats satisfaisants.

3.3 Généralisations

3.3.1 Composantes non normales

Les auteurs remarquent que k peut avoir été surestimé du fait de la présence de composantes non normales. En effet, comme on l'a mentionné précédemment, dans certains cas une densité unimodale peut-être bien approchée par un mélange de deux lois normales avec des moyennes très proches, ou par une seule loi, plus "plate" que la loi normale. Il faudrait donc être en mesure de généraliser cette méthode à des lois autres que la loi normale. La même remarque est faite par Yung-Hsin Chien et Edward George dans la discussion : l'inférence sur k pourrait être faussée par l'hypothèse de normalité des composantes.

Remarquons que ceci n'est pas un problème si le seul but est l'estimation de densité, c'est-à-dire si on considère les RJMCMC comme une alternative à la méthode du noyau par exemple. Mais dans le cas où le but premier est la classification et où les composantes ont une signification physique, ceci pose un problème important.

3.3.2 Cas multivarié

Dès le début de l'article, Richardson et Green affirmaient que si leur exposé se limitait aux mélanges de gaussiennes unidimensionnelles, leur méthode est plus générale et devrait pouvoir s'adapter au cas multidimensionnel. Dans la discussion, D. M. Titterington et Mike West remarquent qu'une généralisation au cas multivarié n'est pas évidente, et pourrait impliquer de nouveaux types de mouvements. Gilles Celeux note, en particulier, que les contraintes d'identifiabilité des composantes, $\mu_1 < ... < \mu_k$, ne sont pas directement généralisables au cas multivarié. De nombreux auteurs proposent d'introduire une mesure de la proximité de deux composantes afin de favoriser le regroupement de deux composantes "proches".

Cette généralisation a été étudiée par Guillaume Saint-Pierre, cf. Saint-Pierre et Garel (2001). Outre les problèmes de généralisation des contraintes, le principal problème rencontré est la difficulté de la généralisation des formules concernant le saut de type (e) c'est-à-dire la séparation ou la réunion de deux composantes :

$$w_{j^*} = w_{j_1} + w_{j_2}$$

$$w_{j^*}\mu_{j^*} = w_{j_1}\mu_{j_1} + w_{j_2}\mu_{j_2}$$

$$w_{j^*}(\mu_{j^*}^2 + \sigma_{j^*}^2) = w_{j_1}(\mu_{j_1}^2 + \sigma_{j_1}^2) + w_{j_2}(\mu_{j_2}^2 + \sigma_{j_2}^2)$$

Si les deux premières de ces équations se généralisent directement, ce n'est pas le cas de la troisième, puisqu'il faut maintenant passer à des matrices de variance-covariance. De plus, la complexité des calculs des probabilités d'acceptation devient problématique dans ce cas.

3.3.3 *A priori* non informatifs

Même si l'utilisation d'*a priori* totalement non informatifs n'est pas possible dans ce contexte, plusieurs auteurs remarquent qu'il serait intéressant d'utiliser des *a priori* partiellement non informatifs afin de limiter la dépendance des résultats au choix des lois *a priori*.

Par exemple Christian Robert remarque qu'il peut être avantageux de paramétrer différemment le modèle de mélanges : la composante i est vue comme une perturbation locale des précédentes. Dans le cas normal on obtient :

$$p_0\mathcal{N}(\theta_0, \tau_0^2) + \sum_{i=1}^{k-2}(1-p_0)...(1-p_{i-1})p_i\mathcal{N}(\theta_0 + ... + \tau_0...\tau_{i-1}\theta_i, \tau_0^2...\tau_i^2)...$$

$$... + (1-p_0)...(1-p_{k-2})\mathcal{N}(\theta_0 + ... + \tau_0...\tau_{k-2}\theta_{k-1}, \tau_0^2...\tau_{k-1}^2)$$

Cette paramétrisation permet l'utilisation de lois *a priori* non informatives comme :

$$\pi(\theta_0, \tau_0) = \frac{1}{\tau_0}$$

$$\forall i, p_i \sim \mathcal{U}[0,1]$$

$$\forall i > 0, \tau_i \sim \mathcal{U}[0,1]$$

$$\forall i > 0, \theta_i \sim \mathcal{N}(0, \zeta^2)$$

D'autres propositions ont été faites. Roeder et Wasserman (1997) ont proposé des lois *a priori* partiellement non informatives pour les paramètres dans les modèles de mélange à k fixé qui devraient pouvoir s'adapter au cas où k varie. L'idée est de ne plus supposer l'indépendance des paramètres. Par exemple les moyennes sont supposées être proches les unes des autres, groupées autour d'un "centre" A qui suit lui une loi complètement non informative (et donc impropre).

3.4 Critère de convergence

De nombreux auteurs remarquent l'absence de critère quantitatif pour déterminer quand les simulations doivent s'arrêter, c'est-à-dire pour choisir n_2. Ceci est désigné comme la recherche d'un "diagnostic de convergence". Selon S. Brooks, les 200000 étapes de simulations ne sont pas suffisantes, ce que reconnaissent Richardson et Green tout en remarquant que de nouvelles simulations, avec 500000 étapes, n'apportaient pas de modification sensible des résultats.

3.5 Discussion du choix des méthodes bayésiennes

3.5.1 Complexité des MCMC dans ce contexte

Certains auteurs critiquent le choix de méthodes bayésiennes dans ce contexte. Par exemple, Murray Aitkin remarque que la méthode des RJMCMC est beaucoup plus complexe, et coûteuse en temps de calcul, qu'une méthode classique. Celle-ci pourrait consister en l'utilisation du maximum de vraisemblance pour différentes valeurs de k, puis d'une procédure de choix de modèle reposant sur la vraisemblance pour déterminer k.

Le problème est en fait le même que pour les méthodes bayésiennes : les équations définissant l'estimateur du maximum de vraisemblance (EMV) ne peuvent être résolues analytiquement. L'EMV devra donc être approché par l'algorithme EM, à k fixé. Le choix de k est ensuite classiquement réalisé par l'utilisation de critères comme l'AIC ou le BIC.

L'algorithme EM converge en général assez rapidement et cette méthode est finalement moins complexe que les RJMCMC. Cependant, deux problèmes (au moins) sont posés par cette méthode :

- l'algorithme EM peut très bien converger vers un maximum local de la vraisemblance qui ne sera pas le maximum global.

- il n'y a pas, dans le cas des modèles de mélanges, de justification théorique à l'utilisation de l'AIC et du BIC, même si ils donnent en pratique des résultats convenables (quoique surestimant en général le nombre de composantes).

Cette critique sur la complexité est nuancée par K. Mengersen et A. George qui remarquent que la méthode n'est finalement pas très complexe à implémenter.

De même, d'autres auteurs estiment que la méthode est d'un grand intérêt pratique : Walter Gilks présente dans la discussion un exemple de modèle en génétique qui pourrait nécessiter l'utilisation de RJMCMC.

Cependant, cette critique reste valable dans le cas mutidimensionnel, comme ceci a été mentionné précédemment, cf. Saint-Pierre et Garel (2001).

3.5.2 Sensibilité du choix de modèle aux lois *a priori*

Murray Aitkin remarque aussi qu'il faudrait étudier la dépendance de la loi *a posteriori* de k à la loi *a priori* choisie pour k. Dans le cas multidimensionnel, selon Saint-Pierre et Garel (2001), cette dépendance est importante, mais pourrait être réduite par l'introduction d'un niveau hiérarchique supplémentaire, par exemple :

$$k \sim \mathcal{P}(\lambda)$$

où λ ne serait plus une valeur fixe mais suivrait lui-même une loi donnée.

La sensibilité aux autres lois *a priori* a été étudiée en détail par Richardson et Green dans leur article.

On voit ici cependant l'intérêt d'être en mesure d'utiliser des *a priori* non informatifs (au moins partiellement) et pour cela de prendre en compte les propositions (présentées précédemment) de Robert dans la discussion ou de Roeder et Wasserman (1997).

3.5.3 Une comparaison numérique

Geoffrey McLachlan et David Peel proposent une comparaison des résultats obtenus par Richardson et Green avec ceux obtenus, sur les mêmes données, en utilisant les mêmes méthodes que celles proposées par Aitkin (algorithme EM puis critère AIC). Les résultats obtenus, au moins sur k, sont assez similaires à ceux de Richardson et Green, sauf pour le jeu de données sur l'acidité, pour lequel les méthodes bayésiennes donnent $k \in \{2, ..., 6\}$ alors que les méthodes classiques rejettent $k > 3$.

4 Mise en pratique

4.1 Le programme

4.1.1 Choix effectués

Le modèles est exactement le même que celui étudié par Richardson et Green. La seule différence est que je n'ai pas imposé de contrainte d'ordre lors des simulations (ce qui impose une légère modification dans la formule de la probabilité d'acceptation).

J'ai réalisé le programme sous le logiciel R. Le code source figure en annexe.

4.1.2 Problèmes rencontrés

La méthode n'est pas très difficile à programmer en général mais pose certaines problèmes : complexité des formules des probabilités d'acceptation, stockage des résultats (dimension variable !)...

En revanche, la lenteur d'execution du programme obtenu a été un réel problème.

4.2 Résultats sur données simulées

4.2.1 Présentation des données

Je présente ici les résultats obtenus pour un fichier de 30 "observations" simulées suivant une loi :

$$\frac{2}{3}\mathcal{N}(0,1) + \frac{1}{3}\mathcal{N}(4,1)$$

Ces simulations figurent en annexe.

Dans un premier temps, j'ai appliqué la méthode décrite dans la partie 1, c'est-à-dire que l'on garde k fixé, égal à 2. Sinon, j'ai utilisé exactement la même structure de lois que Richardson et Green. Dans un second temps, j'ai utilisé la méthode avec saut réversible (k "inconnu").

4.2.2 Résultats obtenus à k fixé

J'ai utilisé seulement 5000 itérations après la période de démarage (très courte, de l'ordre de quelques étapes).

En prenant comme résumé des simulations les moyennes *a posteriori*, on obtient l'estimation suivante de la loi :

$$0.6445\mathcal{N}(0.1369, 12655) + 0.3555\mathcal{N}(4.3909, 1.1495)$$

Des lois *a posteriori* sont représentées dans les figures 4 (moyenne de la première composante), 5 (variance) et 6 (poids).

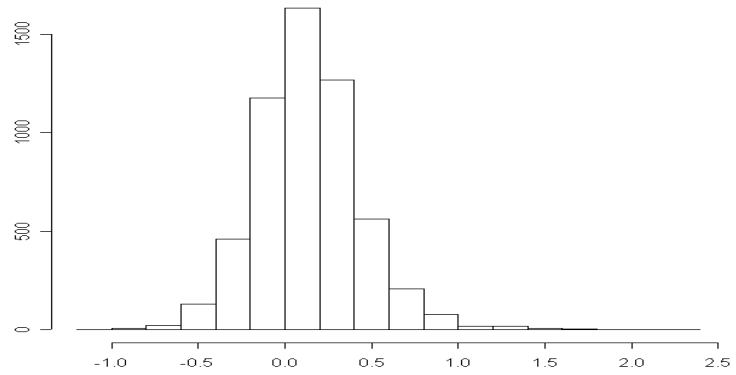

FIG. 4 – Loi *a posteriori* de μ_1

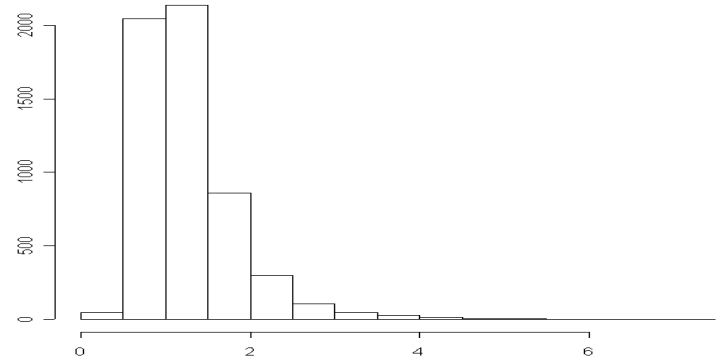

FIG. 5 – Loi *a posteriori* de σ_1^2

4.2.3 Résultats des RJMCMC

100000 simulations ont été réalisées dans ce cas bien que la convergence semble avoir été assez rapide. Les résultats obtenus pour la loi *a posteriori* de k sont représentées dans la figure 7.

Ainsi, la méthode surestime le nombre de composantes dans ce cas. Examinons les résultats conditionnellement à $k = 2$ et $k = 3$. Pour exploiter les résultats, on distingue les composantes par la même méthode que Richardson et Green (contrainte d'ordre sur les moyennes).

Pour $k = 2$, voici les résultats obtenus en estimant les paramètres par leurs espérances *a posteriori* (paramètre estimé / vraie valeur) :

Composante	1	2
Moyenne	0.146 / 0	4.409 / 4
Variance	1.264 / 1	1.110 / 1
Poids	0.648 / 0.667	0.352 / 0.333

Examinons maintenant les résultats obtenus pour $k = 3$ composantes. Ceci pourra peut-être nous aider à comprendre pourquoi $k = 3$ est plus probable au vu des données que $k = 2$.

Composante	1	2	3
Moyenne	-0.585	1.331	4.687
Variance	1.330	1.415	1.037
Poids	0.362	0.341	0.297

Il faut faire très attention en interprétant ces résultats. On pourrait être tenté de croire, par exemple, que la première composante dans $k = 2$ a été "coupée en deux". En fait il n'en est rien, la moyenne

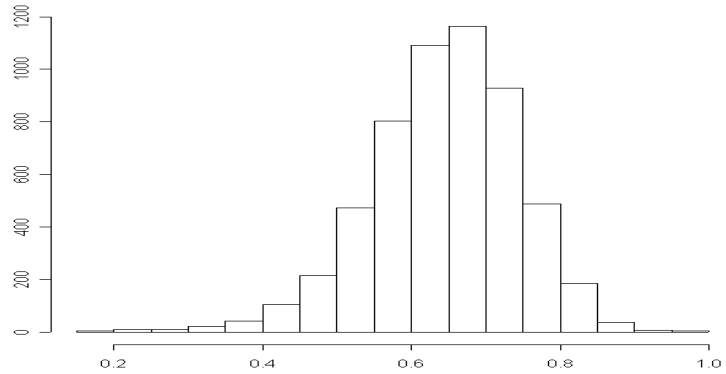

FIG. 6 – Loi *a posteriori* de w_1

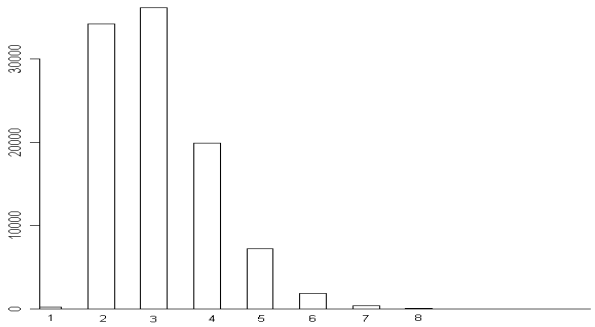

FIG. 7 – Loi *a posteriori* de k pour les données simulées

a posteriori n'a pas vraiment de signification pour la deuxième composante dans $k = 3$. Pour s'en convaincre, il suffit de regarder les lois *a posteriori* des trois moyennes (sous la contrainte d'ordre) : cf. figure 8.

La deuxième composante, dans le cas $k = 2$, est tantôt un "double" de la première, tantôt un double de la troisième.

Afin d'empêcher ce phénomène de se produire trop souvent, on peut utilise une loi *a priori* sur k qui "pénalise" plus les nombres élevés de composantes, en revenant par exemple à la loi de Poisson de paramètre λ, initialement proposée par Richardson et Green. Avec $\lambda = 1$, on obtient déjà des résultats beaucoup plus satisfaisants avec une probabilité *a posteriori* pour $k = 2$ de l'ordre de 0.7, et pour $k = 3$ de l'ordre de 0.25.

Ces résultats donnent ceci dit un facteur de Bayes proche de 1, très légèrement en faveur de 3 composantes.

4.3 Résultats sur données réelles

4.3.1 Intérêt de travailler sur des données réelles

En général, lorsqu'on étudie un modèle statistique on se donne une famille de lois possibles pour les observations :

$$\mathcal{C} = \{\mathbb{P}_\theta, \theta \in \Theta\}$$

et on fait l'hypothèse que la vraie loi des observations $\mathbb{P}^* \in \mathcal{C}$. Ici, on a fait l'hypothèse que les données étaient issues d'un mélange de gaussiennes. Ceci était justifié dans le cas des données simulées suivant cette loi.

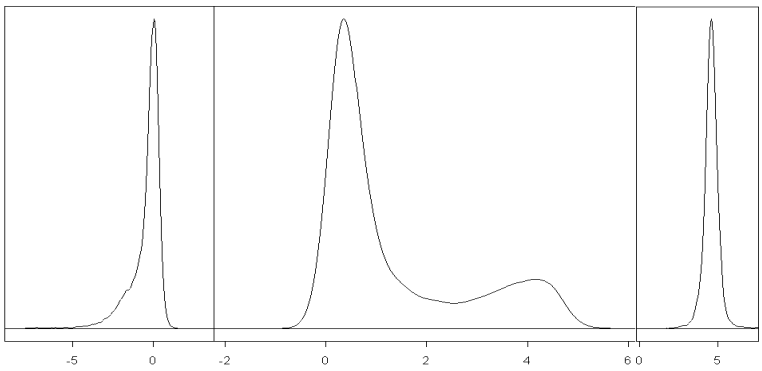

FIG. 8 – Les lois *a posteriori* pour les trois moyennes dans le cas où $k = 3$ pour les données simulées.

Or, pour des données réelles ceci n'est qu'une approximation. En pratique, dans un modèle de mélanges, ceci se traduit par exemple par le fait que deux composantes normales peuvent être utilisées pour représenter une seule classe dont la loi est plus "plate" que la loi normale, ou par la présence d'une observation atypique qui peut favoriser la création d'une nouvelle classe.

Notons en effet que :

- dans les comparaisons entre méthodes classiques et RJMCMC, McLachlan et Peel par exemple affirmaient que le RJMCMC a une tendance à surestimer le nombre de composantes.

- pour les données simulées avec $k = 2$, il y a déjà eu surestimation du nombre de composantes en utilisant l'*a priori* uniforme.

Il est donc intéressant de voir comment réagit la méthode étudiée face à un jeu de données réelles.

4.3.2 Présentation des données

On utilise donc ici le *galaxy data set* déjà étudié par Richardson et Green. Ces données dont disponibles en annexe.

4.3.3 Résultats et commentaires

Ici, seulement 50000 simulations ont été réalisées.

La loi *a posteriori* de k donnée par les simulations est représentée sur la figure 8.

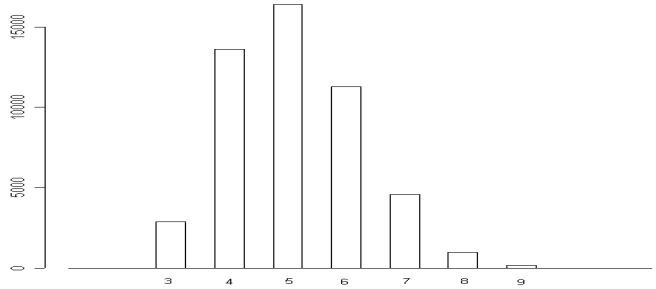

FIG. 9 – Loi *a posteriori* de k pour le *galaxy data set*

Le tableau suivant compare les résultats que j'ai obtenu à ceux obtenus par Richardson et Green avec un *a priori* uniforme.

p(k=...|y)	Richardson & Green	Mes simulations
3	0.061	0.058
4	0.128	0.272
5	0.182	0.328
6	0.199	0.225
7	0.160	0.092
8	0.109	0.020
9	0.071	0.004
10	0.040	0.000

Ainsi, par rapport aux estimations proposées par McLachlan et Peel dans la discussion, on obtient plus de composantes (pour eux le nombre de composantes était compris entre 2 et 6 maximum).

En revanche, en utilisant une loi de Poisson *a priori* ($\lambda = 1$) on obtient des résultats plus proches de ceux obtenus par des méthodes classiques :

- le k le plus probable *a posteriori* est 4 avec une probabilité de 0.56 ;

- il y a une forte évidence en faveur de 3 à 5 composantes, 3 a une probabilité 0.28 et 5 de 0.14, ce qui donne une probabilité totale de 0.98 pour cet intervalle.

- en particulier, la probabilité de $k = 2$ estimée est nulle, après la *burn-in period* la MCMC ne revient plus jamais à deux composantes.

Bibliographie

Brooks, S.P., Giudici, P. & Roberts, G.O., *Efficient construction of reversible jump MCMC proposal distributions*, 2003

Cappé, Olivier, Robert, Christian P. & Rydén, Tobias, *Reversible jump, birth-and-death, and more general continuous time MCMC samplers*, 2002

Celeux, Gilles, Hurn, Merrilee & Robert, Christian, *Computational and Inferential difficulties with mixture posterior distributions*, Journal of the American Statistical Association, 1997

Diebolt, Jean & Robert, Christian P., *Estimation of finite mixture distribution through bayesian sampling*, Journal of the Royal Statistical Society, 1994

Green, Peter J., *Reversible jump Markov chain Monte Carlo computation and Bayesian model determination*, Biometrika, 1995

Green, Peter J. & Richardson, Sylvia, *On Bayesian analysis of mixtures with an unknown number of components*, with discussion, Journal of the Royal Statistical Society, 1997

Green, Peter J. & Richardson, Sylvia, *Correction on "On Bayesian analysis of mixtures with an unknown number of components"*, disponible sur le site web de Peter Green, 1997

Grenander, U. & Miller, M., *Representation of knowledge in complex systems*, with discussion, Journal of the Royal Statistical Society, 1994

McLachlan, Geoffrey & Peel, David, *Finite mixture models*, Wiley series in probability and mathematical statistics, 2000

Phillips, D. B. & Smith, A. F., *Bayesian model comparison via jump diffusions*, in *Markov Chain Monte Carlo in practice*, Chapman and Hall, 1996

Robert, Christian P., *Mixtures of distributions : inference and estimation*, in *Markov Chain Monte Carlo in practice*, Chapman and Hall, 1996

Robert, Christian P. & Casella, George, *Monte Carlo Statistical Methods*, Springer texts in statistics, 1999

Roeder, Kathryn & Wasserman, Larry, *Practical bayesian density estimation using mixture of normals*, Journal of the American Statistical Association, 1997

Saint-Pierre, Guillaume & Garel, Bernard, *Reversible Jump MCMC for multivariate gaussian mixture models*, Technical report, ENSEEIHT, 2001

Stephens, M., *Bayesian analysis of mixture models with an unknown number of components - an alternative to reversible jump methods*, Annals of Statistics, 2000 (a)

Stephens, M., *Dealing with label-switching in mixture models*, Journal of the Royal Statistical Society, 2000 (b)

Waagepetersen, Rasmus & Sorensen, Daniel, *A tutorial on Reversible Jump MCMC with a view toward applications in QTL-mapping* , International Statistical Review, 2001