Généralités sur l'inférence bayésienne

Nous présentons dans ce chapitre les fondements de l'inférence bayésienne. L'inférence bayésienne est un domaine des statistiques très riche et ouvrant la voie à diverses applications. Nous parlons d'inférence "bayésienne" lorsque l'on se donne une distribution a priori sur ce qu'on cherche à inférer. Ce qu'on cherche à inférer peut être le paramètre θ d'un modèle paramétrique $p(y;\theta)$, mais on peut également utiliser l'inférence bayésienne pour estimer la réalisation cachée d'un modèle à données latentes. Dans un modèle à données latentes, le signal observé est considéré comme la réalisation y d'un processus Y et ce que l'on cherche est la réalisation x d'un processus X, les deux processus étant liés par une loi de probabilité $p(x,y)$.
Ce chapitre se divise en deux sections. La première amène le formalisme bayésien dans sa généralité. Dans la deuxième section, nous nous intéressons au choix de l'a priori et au choix de la loi $p(x,y)$ dans les modèles à données latentes. Concernant l'inférence bayésienne dans un modèle paramétrique $p(y;\theta)$, nous y abordons deux types de lois a priori. Les deux lois a priori abordées sont les lois conjuguées à une famille paramétrique et les mesures de Jeffreys. Les premières présentent un intérêt algorithmique car la loi a posteriori est dans la même famille paramétrique que la loi a priori. Les lois conjuguées sont souvent utilisées en estimation des paramètres par échantillonnage de Gibbs car la règle de Bayes est simple à implémenter [36]. Quant aux mesures de Jeffreys, elles font partie de la catégorie des mesures a priori dites "non informatives". On choisit d'utiliser un a priori non informatif lorsque l'on ne dispose d'aucune connaissance sur le paramètre. Concernant les modèles à données latentes, les lois a priori seront choisies de façon à ce que les modèles $p(x,y)$ permettent d'utiliser les algorithmes d'inférence bayésienne tels que les algorithmes de Baum-Welsh et de Viterbi. Ces modèles devront être suffisamment simples pour pouvoir utiliser ce type d'algorithmes, et suffisamment riches pour pouvoir modéliser certaines propriétés comme la markovianité, la semi-markovianité ou la dépendance longue dans les observations.

1.1 Principe de l'inférence bayésienne

On considère Y une variable aléatoire à valeurs dans un $\mathbb{R}$-espace vectoriel de dimension finie $\mathcal{Y}$ muni de sa tribu borélienne $\mathcal{B}_{\mathcal{Y}}$. Un modèle statistique paramétrique pour la loi de Y est une famille de densités de probabilité $\{y \in \mathcal{Y} \to p(y;\theta) : \theta \in \Theta\}$ par rapport à une mesure ν sur $\mathcal{Y}$. La fonction de $\mathcal{Y} \times \Theta$ dans $\mathbb{R}^+$ qui à (y,θ) associe $p(y;\theta)$ est appelée vraisemblance. L'ensemble Θ est l'ensemble des paramètres du modèle ; on considèrera dans la suite que

$\Theta \subset \mathbb{R}^k$. Muni de sa tribu borélienne $\mathcal{B}_\Theta$, $(\Theta, \mathcal{B}_\Theta)$ est un espace mesurable, il sera également muni d'une mesure de référence. Lorsque Θ est un sous-ensemble discret, la mesure de référence est la mesure de décompte et lorsque Θ est un ouvert non vide de $\mathbb{R}^k$, ce sera la mesure induite par la mesure de Lebesgue $\lambda_{\mathbb{R}^k}$.
Une stratégie de décision est une fonctionnelle φ de $\mathcal{Y}$ dans Θ. On l'appelle aussi estimateur de θ.

1.1.1 Fonction de coût et risque

Une fonction $L : \Theta \times \Theta \to \mathbb{R}^+$ est dite "fonction de coût" si elle vérifie $L(\theta, \hat{\theta}) = 0$ lorsque $\hat{\theta} = \theta$. Soit $\varphi : \mathcal{Y} \to \Theta$ une stratégie de décision. On appelle risque la quantité

$$R(\theta, \varphi) = \mathbb{E}_\theta \left[L(\theta, \varphi(Y)) \right], \tag{1.1}$$

où $\mathbb{E}_\theta$ est l'espérance.
Pour tout $\theta \in \Theta$, le risque est le coût moyen induit par φ.

1.1.2 Des stratégies admissibles aux stratégies bayésiennes

Définition 1.1.1 (Relation de préférence et stratégies admissibles)**.** *Notons Φ l'ensemble des stratégies de décisions. Une relation de préférence est une relation d'ordre sur Φ. La relation "φ_1 préférée à φ_2" (resp. strictement préférée) est notée $\varphi_1 \gtrsim \varphi_2$ (resp. $\varphi_1 > \varphi_2$) et on dit que φ_1 et φ_2 sont équivalentes si $\varphi_1 \gtrsim \varphi_2$ et $\varphi_2 \gtrsim \varphi_1$. On dit que φ est une stratégie admissible s'il n'existe pas de stratégie qui lui soit strictement préférée.*

La relation "$\varphi \gtrsim \varphi' \Leftrightarrow \forall \theta,\ R(\theta, \varphi) \leq R(\theta, \varphi')$" n'est pas une relation d'ordre total ; en effet, pour certaines valeurs de θ, on peut avoir $R(\theta, \varphi) \leq R(\theta, \varphi')$ tandis que pour d'autres valeurs de θ, on a $R(\theta, \varphi) \geq R(\theta, \varphi')$. On peut alors considérer le risque bayésien qui ne dépend pas de θ mais d'une mesure μ sur l'espace des paramètres $(\Theta, \mathcal{B}_\Theta)$ appelée "mesure a priori". On notera f la densité de la mesure a priori par rapport à la mesure de référence de Θ. La mesure a priori n'est pas obligatoirement une mesure de probabilité. De plus, elle peut vérifier $\mu(\Theta) = +\infty$, on dit alors qu'elle est impropre. On exigera par contre que la quantité $p_\mu(y) =^{\text{def}} \displaystyle\int_\Theta p(y;\theta) d\mu(\theta)$ soit finie. Dans ce cas, la densité $\theta \to \dfrac{p(y;\theta) f(\theta)}{p_\mu(y)}$ définit une mesure de probabilité sur $(\Theta, \mathcal{B}_\Theta)$ appelée mesure a posteriori que l'on notera $\mu(.|y)$. Sa densité sera notée $f(.|y)$. La formule :

$$f(\theta|y) = \frac{p(y;\theta) f(\theta)}{p_\mu(y)}$$

est parfois appelée "la règle de Bayes".
Le risque bayésien est ensuite défini par :

$$\rho(\mu, \varphi) = \mathbb{E}_\mu \left[R(\theta, \varphi) \right], \tag{1.2}$$

où $\mathbb{E}_\mu$ est l'intégration sous la mesure μ.
La mesure a priori quantifie la connaissance que l'on a avant toute expérience sur le paramètre

θ. Nous détaillerons à la section 1.2 le choix de cet a priori.
Dans le cadre bayésien, on dit que φ est préférée à φ' si $\rho(\mu,\varphi) \leq \rho(\mu,\varphi')$. Dans ce cas la stratégie admissible φ_μ, si elle existe, est appelée "stratégie bayésienne".
Le risque bayésien s'écrit également :

$$\rho(\mu,\varphi) = \int_{\mathcal{Y}} \mathbb{E}_\mu \left[L(\theta,\varphi(y))|y\right] p_\mu(y) d\nu(y),$$

où $\mathbb{E}_\mu[.|y]$ est l'intégration sous la mesure a posteriori $\mu(.|y)$.
La quantité $r_\mu(y,\varphi) = \mathbb{E}_\mu \left[L(\theta,\varphi(y))|y\right]$ est appelée risque a posteriori et on a le résultat classique suivant :

Proposition 1.1.1. *Si φ est une stratégie de décision telle que :*

$$\forall y \in \mathcal{Y},\ \forall \varphi' \in \Phi,\ r_\mu(y,\varphi) \leq r_\mu(y,\varphi'), \tag{1.3}$$

alors φ est une stratégie bayésienne.

Preuve.
Voir [44]. □

Dans le cas où la stratégie bayésienne est unique, il suffit alors de chercher la décision satisfaisant (1.3).
Donnons quelques exemples de stratégies bayésiennes couramment rencontrées :

- si Θ est continu et si $L(\theta,\hat{\theta}) = (\theta-\hat{\theta})^2$, alors $r_\mu(y,\varphi) = \mathbb{E}_\mu\left[(\theta-\varphi(y))^2|y\right]$. La stratégie bayésienne est l'espérance a posteriori $\hat{\theta}(y) = \mathbb{E}_\mu(\theta|y)$;
- si Θ est discret et si $L(\theta,\hat{\theta})$ vaut 0 si $\theta = \hat{\theta}$ et 1 sinon, alors la stratégie bayésienne est donnée par $\hat{\theta}_{MAP}(y) = \arg\max_\theta \mu(\theta|y)$. On appelle cet estimateur "estimateur du Maximum A Posteriori" (MAP) ;
- soit $\epsilon > 0$. Si Θ est continu et si $L(\theta,\hat{\theta})$ vaut 0 lorsque $\left|\theta-\hat{\theta}\right| < \epsilon$ et 1 sinon, alors la stratégie bayésienne est donnée par $\hat{\theta}_\epsilon(y) = \arg\max_\theta \mu([\theta-\epsilon,\theta+\epsilon]|y)$.

L'estimateur du MAP est généralisé au cas où Θ est continu par $\hat{\theta}_{MAP}(y) = \arg\max_\theta f(\theta|y)$; cependant, il n'est pas défini à partir de la même fonction de perte que dans le cas discret. Le lien avec l'estimateur $\hat{\theta}_\epsilon$ et l'étude de l'estimateur du MAP dans le cas continu figurent dans [15] pages 258-259.
Notons l'exemple classique suivant :

Exemple 1.1.1. Soit $\{p(y;\theta) : \theta \in \Theta\}$ un modèle paramétrique. Considérons les deux estimateurs de θ suivants :

- maximum de vraisemblance : $\hat{\theta}_{ML} = \arg\max p(y;\theta)$;
- maximum a posteriori : $\hat{\theta}_{MAP} = \arg\max f(\theta|y)$.

Si la densité de la loi a priori est $f(\theta) \propto 1$ pour tout $\theta \in \Theta$, alors la densité de la loi a posteriori est $f(\theta|y) \propto \dfrac{p(y;\theta)}{p_\mu(y)}$ et donc $\hat{\theta}_{ML} = \hat{\theta}_{MAP}$.

Remarque : La loi uniforme a longtemps été considérée comme la loi non informative. Le caractère non informatif de cette loi a été énoncé pour la première fois par T. Bayes dans un contexte particulier. En effet, celui-ci considère qu'en absence de connaissance sur le paramètre θ, nous n'avons aucune raison de privilégier un événement $\theta \in A$ plutôt qu'un autre. Mais comme l'a souligné R. A. Fisher, lorsqu'on effectue un changement de paramétrage $\eta = g(\theta)$, ne pas connaître θ est équivalent à ne pas connaître η. Cependant si θ suit une loi uniforme, η ne suit pas en général une loi uniforme. Le contexte particulier dans lequel travaillait T. Bayes fut celui où Θ est discret. Dans ce cas, l'image d'une loi uniforme par une fonctionnelle est encore une loi uniforme. Lorsque l'espace Θ est continu, nous devons alors choisir un autre type de loi non informative. Plus exactement, nous disons qu'une loi est non informative lorsqu'elle maximise la quantité d'information manquante. Nous devons pour cela définir ce qu'est la quantité d'information. Lorsque θ prend ses valeurs dans $\{\theta_1, \ldots, \theta_m\}$ avec les probabilités $\mu(\theta_j) = \mu_j$, celle-ci est définie comme l'entropie de Shannon :

$$H(\mu) = -\sum_{j=1}^{m} \log(\mu_j)\mu_j. \tag{1.4}$$

Nous voyons que cette quantité est bien maximale lorsque θ suit la loi uniforme.
Lorsque Θ est continu et θ suit une loi de densité f par rapport à la mesure de Lebesgue $\lambda_{\mathbb{R}^k}$, l'entropie de Shannon est généralisée par :

$$H(\mu) = -\int_{\Theta} \log f(\theta) f(\theta) d\lambda_{\mathbb{R}^k}(\theta).$$

Cependant, ce n'est pas la bonne mesure d'information que nous devons maximiser. La maximisation de cette quantité fournit des lois uniformes et nous avons souligné que la loi uniforme n'est pas la bonne mesure non informative dans le cas continu. De la même façon, une quantité d'information ne doit pas dépendre du paramétrage, ainsi la quantité d'information sur θ doit être égale à la quantité d'information sur $\eta = g(\theta)$. Cependant, l'entropie de Shannon est invariante par changement de paramétrage uniquement dans le cas discret. Nous verrons dans la sous-section 1.2.1 quelle quantité d'information nous devrons maximiser dans le cas continu et quelle mesure non informative devra être utilisée.

1.2 Choix de l'a priori

1.2.1 Mesures de Jeffreys

Comme nous l'avons discuté dans l'exemple ci-dessus, choisir une loi a priori uniforme comme loi non informative n'est pas judicieux lorsque l'espace des paramètres est continu. Nous devons alors choisir une autre loi non informative. Pour cela, nous allons commencer par définir la quantité d'information "manquante" que l'on doit maximiser. Ensuite, nous montrerons que cette quantité d'information est indépendante du paramétrage.
Soient $\Theta \subset \mathbb{R}^k$ et $\Xi \subset \mathbb{R}^k$ deux ensembles de paramètres, ouverts de $\mathbb{R}^k$. Soit Y une variable aléatoire prennant ses valeurs dans un espace vectoriel de dimension finie $\mathcal{Y}$ muni d'une mesure de référence ν. On considère qu'il existe un $\mathcal{C}^1$-difféomorphisme g de Θ dans Ξ. Rappelons qu'un $\mathcal{C}^1$-difféomorphisme est une application de classe $\mathcal{C}^1$ bijective et dont

l'application réciproque est également de classe $\mathcal{C}^1$. Ainsi, si $\{p(y;\theta) : \theta \in \Theta\}$ est un modèle paramétrique de Y dans le paramétrage Θ, le modèle paramétrique dans le paramétrage Ξ est $\{q(y;\eta) : \eta \in \Xi\}$ où :

$$q(y;\eta) = p(y;\theta), \text{ avec } \eta = g(\theta).$$

Soit μ_Θ une mesure a priori sur Θ de densité f_Θ par rapport à la mesure de Lebesgue. La quantité d'information manquante sur le paramètre θ lorsque la loi a priori est μ_Θ est définie dans [15] pages 157-158, comme l'information de Kullback :

$$\bar{K}(\mu_\Theta(.|y), \mu_\Theta) = \int_\Theta \log\left(\frac{f_\Theta(\theta|y)}{f_\Theta(\theta)}\right) d\mu_\Theta(\theta|y). \tag{1.5}$$

Contrairement à l'entropie de Shannon, cette quantité d'information dépend de l'observation y. Elle s'interprète comme l'information manquante dans la loi a priori et disponible dans l'observation.
Ecrivons cette quantité d'information dans le paramétrage Ξ, μ_Ξ sera la mesure a priori correspondante à μ_Θ dans le paramétrage Ξ et f_Ξ sa densité. On a :

$$f_\Theta(\theta) = |\mathrm{Jac}_\theta(g)|\, f_\Xi(g(\theta)),$$

où $|\mathrm{Jac}_\theta(g)|$ est le déterminant jacobien de g.
On note $p(y) = \displaystyle\int_\Theta p(y;\theta) d\mu_\Theta(\theta)$ (resp. $q(y) = \displaystyle\int_\Theta q(y;\eta) d\mu_\Xi(\eta)$).
On a :

$$\begin{aligned}\bar{K}(\mu_\Xi(.|y), \mu_\Xi) &= \int_\Xi \log\left(\frac{f_\Xi(\eta|y)}{f_\Xi(\eta)}\right) d\mu_\Xi(\eta|y) \\ &= \int_\Xi \log\left(q(y;\eta)\right) \frac{q(y;\eta)}{q(y)} d\mu_\Xi(\eta) - \log\left(q(y)\right).\end{aligned}$$

Effectuant le changement de variable $\eta = g(\theta)$, on a :

$$q(y) = \int_\Theta q(y;g(\theta)) \underbrace{|\mathrm{Jac}_\theta(g)|\, f_\Xi(g(\theta)) d\lambda_{\mathbb{R}^k}(\theta)}_{d\mu_\Theta(\theta)} = p(y),$$

On a également :

$$\begin{aligned}&\int_\Xi \log\left(q(y;\eta)\right) \frac{q(y;\eta)}{q(y)} d\mu_\Xi(\eta) \\ =\ &\int_\Theta \log\left(q(y;g(\theta))\right) \frac{q(y;g(\theta))}{q(y)} |\mathrm{Jac}_\theta(g)|\, f_\Xi(g(\theta)) d\lambda_{\mathbb{R}^k}(\theta) \\ =\ &\int_\Theta \log\left(p(y;\theta)\right) \frac{p(y;\theta)}{p(y)} d\mu_\Theta(\theta).\end{aligned}$$

Ainsi :

$$\bar{K}(\mu_\Xi(.|y), \mu_\Xi) = \bar{K}(\mu_\Theta(.|y), \mu_\Theta). \tag{1.6}$$

Cette quantité ne dépend donc pas du paramètrage choisi.
Soit $y_{1:N} = (y_1, \ldots, y_N)$ un échantillon de réalisations indépendantes de $p(y;\theta)$ et $p(y_{1:N};\theta) = \prod_{n=1}^{N} p(y_n;\theta)$ sa vraisemblance. On définit la quantité d'information moyenne par :

$$\mathcal{J}_N(\mu_\Theta) = \int_{\Theta\times\mathcal{Y}^N} \bar{K}(\mu_\Theta(.|y_{1:N}), \mu_\Theta) p(y_{1:N};\theta) d\mu_\Theta(\theta) d\nu(y_{1:N}).$$

On va choisir comme mesure a priori une mesure maximisant cette quantité d'information. L'expression d'une telle mesure, appelée mesure de Jeffreys, est donnée par la proposition 1.2.1. Considérons pour cela la matrice d'information de Fisher $I_Y(\theta)$ de $p(y;\theta)$. Sous les conditions d'interversion "dérivation" et "intégrale", le coefficient (i,j) de $I_Y(\theta)$ vérifie :

$$\begin{aligned}(I_Y(\theta))_{i,j} &= \mathbb{E}_\theta\left(\frac{\partial}{\partial\theta_i}\log p(Y;\theta) \times \frac{\partial}{\partial\theta_j}\log p(Y;\theta)\right)\\ &= -\mathbb{E}_\theta\left(\frac{\partial^2}{\partial\theta_i\theta_j}\log p(Y;\theta)\right).\end{aligned}$$

Proposition 1.2.1 (Forme des a priori de Jeffreys)**.** *Soient Θ un ouvert non vide de $\mathbb{R}^k$, $\{p(y;\theta) : \theta \in \Theta\}$ une famille de densités de probabilité sur $\mathcal{Y}$ par rapport à une mesure ν, et $y_{1:N} = (y_1, \ldots, y_N)$ un échantillon de $p(y;\theta)$.*
Si les conditions suivantes sont vérifiées :

- *la loi a posteriori $\mu_\Theta(.|y_{1:N})$ converge en loi, lorsque N tend vers l'infini, vers la loi normale de $\mathbb{R}^k$, notée $\hat{\mu}_\Theta(.|y_{1:N})$, dont la moyenne est l'estimateur du maximum de vraisemblance $\hat{\theta} = \hat{\theta}(y_{1:N})$ et dont la matrice de covariance est $\frac{1}{N}\left[I_Y(\hat{\theta})\right]^{-1}$;*
- *pour tout compact K de Θ, on a :*

$$\lim_{N\to+\infty} \int_{K\times\mathcal{Y}^N} \bar{K}(\mu_\Theta(.|y_{1:N}), \hat{\mu}_\Theta(.|y_{1:N})) p(y_{1:N};\theta) d\mu_\Theta(\theta) d\nu(y_{1:N}) = 0.$$

Alors, il existe un entier $N \geq 1$ tel que pour tout $n \geq N$, la quantité

$$\mathcal{J}_n(\mu_\Theta) = \int_{\Theta\times\mathcal{Y}^n} \bar{K}(\mu_\Theta(.|y_{1:n}), \mu_\Theta) p(y_{1:n};\theta) d\mu_\Theta(\theta) d\nu(y_{1:n}),$$

est maximale pour la mesure μ_Θ de densité $\theta \to \sqrt{\det I_Y(\theta)}$ par rapport à la mesure de Lebesgue. Cette mesure est appelée mesure de Jeffreys.

Preuve.
Voir [47] pages 127-128. □

La mesure de Jeffreys maximise l'information manquante moyenne lorsque la taille de l'échantillon est suffisamment grande, on la considérera donc comme non informative.
Montrons maintenant qu'une mesure de Jeffreys est transformée en une autre mesure de Jeffreys par changement de paramétrage. Sachant que l'information de Fisher $I_Y(\theta)$ dépend du paramétrage de la loi de Y, on notera cette matrice $I_{Y,\Theta}$ (resp. $I_{Y,\Xi}$) lorsque la loi est paramétrée par Θ (resp. Ξ). Si η suit la loi de Jeffreys de densité $\eta \to \sqrt{\det I_{Y,\Xi}(\eta)}$ par rapport

à la mesure de Lebesgue, alors en utilisant le théorème de changement de variable, $\theta = g^{-1}(\eta)$ suit la loi de densité $\theta \to \sqrt{\det I_{Y,\Xi}(g(\theta))}\,|\mathrm{Jac}_\theta(g)|$. De plus :

$$\begin{pmatrix} \dfrac{\partial \log p(y;\theta)}{\partial \theta_1} \\ \vdots \\ \dfrac{\partial \log p(y;\theta)}{\partial \theta_k} \end{pmatrix} = (\mathrm{Jac}_\theta(g))^T \begin{pmatrix} \dfrac{\partial \log q(y;\eta := g(\theta))}{\partial \eta_1} \\ \vdots \\ \dfrac{\partial \log q(y;\eta := g(\theta))}{\partial \eta_k} \end{pmatrix},$$

où $\mathrm{Jac}_\theta(g)$ est la matrice jacobienne de g. Ainsi :

$$I_{Y,\Theta}(\theta) = (\mathrm{Jac}_\theta(g))^T\, I_{Y,\Xi}(g(\theta))\mathrm{Jac}_\theta(g).$$

On en déduit que la loi de θ est la loi de Jeffreys de densité $\theta \to \sqrt{\det I_{Y,\Theta}(\theta)}$ par rapport à la mesure de Lebesgue.
Nous verrons au chapitre 6 une autre façon d'introduire les mesures de Jeffreys ainsi que leur relation avec les lois uniformes.

1.2.2 Lois a priori conjuguées

Définition 1.2.1 (Lois conjuguées)**.** *Soit $\{p(y;\theta) : \theta \in \Theta\}$ un modèle paramétrique. Une loi a priori μ appartenant à un modèle paramétrique est "conjuguée" à ce modèle si la loi a posteriori $\mu(\theta|y) \propto \mu(\theta)p(y;\theta)$ appartient au même modèle paramétrique que la loi a priori.*

Le paramètre de la loi a priori est couramment appelé hyperparamètre. Lorsqu'on utilise les lois conjuguées, la règle de Bayes revient à remettre à jour l'hyperparamètre.
Le tableau 1.1 donne quelques exemples de lois conjuguées pour des familles de lois usuelles, IG désigne l'inverse d'une loi gamma. Nous donnons dans ce tableau les lois a priori conjugées et les lois a posteriori $\mu(\theta|y_{1:N}) \propto \mu(\theta)\prod_{n=1}^{N} p(y_n;\theta)$ correspondantes, où $y_{1:N} = (y_1, \ldots, y_N)$ est un échantillon de réalisations indépendantes de $p(y;\theta)$.

Famille paramétrique	Loi a priori conjuguée	Loi a posteriori
Loi normale $\mathcal{N}_{\mathbb{R}}(m,s)$, paramètre m	$\mu(m) \sim \mathcal{N}_{\mathbb{R}}(\mu,\gamma)$	$\mu(m\|y_{1:N}) \sim \mathcal{N}_{\mathbb{R}}\left(\dfrac{\mu s+\gamma\sum_{n=1}^{N} y_n}{s+N\gamma}, \dfrac{\gamma s}{s+N\gamma}\right)$
Loi normale $\mathcal{N}_{\mathbb{R}}(m,s)$, paramètre s	$\mu(s) \sim IG(a,b)$	$\mu(s\|y_{1:N}) \sim IG\left(a+\dfrac{N}{2}, \dfrac{2b}{2+b\sum_{n=1}^{N}(y_n-m)^2}\right)$
Loi exponentielle $\mathcal{E}(\lambda)$	$\mu(\lambda) \sim \Gamma(a,b)$	$\mu(\lambda\|y_{1:N}) \sim \Gamma\left(a+N, \dfrac{b}{1+b\sum_{n=1}^{N} y_n}\right)$
Loi binômiale de paramètre $q \in [0,1]$	$\mu(q) \sim \beta(a,b)$	$\mu(q\|y_{1:N}) \sim \beta\left(a+\sum_{n=1}^{N} y_n, b+NK-\sum_{n=1}^{N} y_n\right)$

TAB. 1.1 – Familles conjuguées de familles paramétriques et paramètre de la loi a posteriori fonction de celui de la loi a priori.

1.2.3 Modèles à données latentes

Considérons un couple de processus $Z = (X_s, Y_s)_{s\in\mathcal{S}}$ où Y est observable et X ne l'est pas, chaque X_s prend ses valeurs dans un ensemble fini $\mathcal{X} = \{\omega_1, \ldots, \omega_K\}$ et chaque Y_s prend ses valeurs dans un $\mathbb{R}$-espace vectoriel de dimension finie. La réalisation y de Y représente donc le signal observé et nous souhaitons estimer la réalisation x de X. Les valeurs prises par X_s sont appelées "classes" ou "étiquettes" et la réalisation x est parfois appelée segmentation du signal y. $\mathcal{S}$ est appelé l'ensemble des sites, il peut être un sous-ensemble de $\mathbb{N}$ comme dans le cas des chaînes, un ensemble muni d'une structure d'arbre dans le cas des arbres ou un sous-ensemble de $\mathbb{Z}^p$ dans le cas des champs, il représente ainsi la structure "topologique" du signal. Les deux processus sont liés par une densité de probabilité du type $p(x,y;\theta)$, où θ est le paramètre du modèle. La loi a posteriori $p(x|y;\theta)$ représente la connaissance que l'on a sur x à partir de l'observation y. Les modèles statistiques $p(x,y;\theta)$ seront choisis de façon à ce que la loi a posteriori $p(x|y;\theta)$ soit calculable dans un temps raisonnable pour des processus de "grande taille". Par exemple, pour une image de taille 256×256 que l'on souhaite segmenter en 2 classes, il existe $2^{256\times256} \approx 2\times10^{19728}$ valeurs pour $p(x|y;\theta)$. Dans le cas général, le calcul de $p(x|y;\theta)$ par la formule de Bayes est de complexité algorithmique trop élevée. Dans le cas où Z est choisi comme un vecteur à composantes indépendantes, $p(x|y;\theta) = \prod_{s\in\mathcal{S}} p(x_s|y_s;\theta)$ et chaque $p(x_s|y_s;\theta)$ ne prend que K valeurs, le calcul de $p(x|y;\theta)$ se fait très simplement. Cependant ce modèle ne permet pas de prendre en compte les éventuelles dépendances au sein des états cachés et des observations. Ainsi le modèle devra être choisi suffisamment simple pour permettre le calcul rapide de la loi a posteriori, et suffisamment riche pour modéliser les situations de dépendance les plus réalistes possible.

Remarque : Dans la démarche bayésienne classique, il est courant de se donner la loi a priori $p(x;\theta)$ qui représente les dépendances au sein du processus caché et la loi d'attache aux données $p(y|x;\theta)$, ainsi $p(x,y;\theta) = p(x;\theta)p(y|x;\theta)$. Cependant, comme nous le verrons, se donner directement la loi jointe $p(x,y;\theta)$ permet de modéliser des situations de dépendance

plus complexes.
Donnons quelques exemples de modèles à données latentes, on omettra le paramètre θ. Dans les modèles présentés, on prendra $\mathcal{S} = \{1, \ldots, N\}$ et on notera $z_{1:N}$ la réalisation de Z.

Mélange indépendant

Soit $\mathcal{S} = \{1, \ldots, N\}$ et considérons les variables $Z_n = (X_n, Y_n)$ indépendantes. Nous avons :

$$p(z_{1:N}) = \prod_{n=1}^{N} p(x_n, y_n) = \prod_{n=1}^{N} p(x_n)p(y_n|x_n).$$

Chaînes de Markov cachées à bruit indépendant

La loi jointe $p(z_{1:N})$ est donnée par :

$$p(z_{1:N}) = p(x_1)p(y_1|x_1) \prod_{n=1}^{N-1} p(x_{n+1}|x_n)p(y_{n+1}|x_{n+1}),$$

ainsi :

- $p(x_{1:N}) = p(x_1) \prod_{n=1}^{N-1} p(x_{n+1}|x_n)$, soit X est une chaîne de Markov ;
- $p(y_{1:N}|x_{1:N}) = \prod_{n=1}^{N} p(y_n|x_n)$.

Chaînes de Markov couples

La loi jointe $p(z_{1:N})$ est donnée par :

$$p(z_{1:N}) = p(x_1, y_1) \prod_{n=1}^{N} p(x_{n+1}, y_{n+1}|x_n, y_n).$$

Dans ce modèle, X n'est plus obligatoirement une chaîne de Markov, les variables aléatoires Y_n ne sont plus obligatoirement indépendantes conditionnellement à X et l'égalité $p(y_n|x_{1:N}) = p(y_n|x_n)$ n'a plus obligatoirement lieu. Nous reviendrons sur ce modèle au chapitre 3.

Chaînes de Markov triplets

Dans les chaînes de Markov triplets, nous introduisons un troisième processus U, dit processus auxiliaire, tel que le triplet (X, U, Y) soit une chaîne de Markov. Ce modèle généralise celui des chaînes de Markov couples ; une chaîne de Markov couple est une chaîne de Markov triplet telle que $U = X$. De plus, si on note $V = (X, U)$, le processus (V, Y) est une chaîne de Markov couple ; ainsi les algorithmes d'inférence bayésienne étudiés au chapitre 2 dans le cas des chaînes couples restent utilisables dans les chaînes de Markov triplets.
Comme nous le verrons dans les chapitres suivants, U peut avoir différentes interprétations. Il peut modéliser la non stationnarité de X [65, 68], ou la semi-markovianité, auquel cas U_n est le temps de séjour restant pour X dans la valeur x_n [68]. D'autres interprétations de U sont présentées dans [1, 2, 22, 24, 65]. Ce modèle est particulièrement riche car aucune des chaînes X, U, Y, (X, U), (X, Y) ou (U, Y) n'est nécessairement une chaîne de Markov.

Conclusion

Nous avons présenté dans ce chapitre les notions classiques de l'inférence bayésienne. L'approche classique de l'inférence bayésienne consiste à se donner une loi a priori sur un paramètre ou une réalisation cachée que l'on veut estimer. Cette loi a priori représente la connaissance avant toute expérience sur le paramètre. L'estimateur de ce paramètre inconnu est ensuite choisi selon un critère d'optimalité en se donnant une fonction de perte. Le choix du critère d'optimalité dépend de la nature du problème considéré, ce qui confère aux méthodes bayésiennes une grande souplesse. Nous avons introduit, dans une deuxième section, le choix de la loi a priori. La loi a priori peut être choisie conformément à la connaissance que l'on a sur le paramètre ou la réalisation cachée. Ainsi, si on ne dispose d'aucune connaissance, on est amené à considérer des mesures a priori non informatives telles que les mesures de Jeffreys. La forme de la loi a priori peut être également choisie de façon subjective, de façon à pouvoir calculer facilement la loi a posteriori. C'est le cas notamment des lois a priori conjuguées, mais c'est aussi le cas de certaines lois $p(x)$ dans les modèles à données latentes. Cependant, comme nous l'avons vu aux travers des exemples des chaînes de Markov couples et triplets, se donner la loi jointe $p(x, y)$ au lieu de se donner la loi a priori $p(x)$ et la loi conditionnelle $p(y|x)$ permet de considérer des situations de dépendance plus complexes. Cette loi jointe devra être choisie suffisamment simple de façon à pouvoir calculer facilement la loi a posteriori $p(x|y)$ grâce aux algorithmes que nous verrons au chapitre suivant. Elle devra également être choisie suffisamment riche de façon à modéliser une gamme variée de comportement. Nous verrons au cours de cette thèse différents modèles à données latentes, dont certains originaux, pour lesquels la loi a posteriori est calculable, ce qui rend possible l'utilisation des méthodes bayésiennes de recherche du processus caché x.

Chapitre 2

Inférence bayésienne dans les modèles de Markov cachés

Dans ce chapitre, nous détaillons l'estimation des paramètres et des états cachés dans certains modèles à données latentes classiques. Dans toute la suite, on considère un modèle paramétrique $\{z \to p(z;\theta) : \theta \in \Theta\}$ pour le couple de processus $Z = (X, Y) = (X_u, Y_u)_{u\in\mathcal{S}}$, où $\mathcal{S}$ est un ensemble fini de sites. Dans un modèle à données latentes, nous observons y une réalisation de Y, et nous devons estimer x la réalisation cachée du processus X. Le modèle paramétrique représente la relation probabiliste entre la réalisation cachée et l'observation. La réalisation x de X sera estimée à partir de la loi a posteriori $p(x|y;\theta)$ selon un critère d'optimalité déterminé par une fonction de perte L. Dans la suite, chaque X_u prendra ses valeurs dans un ensemble fini $\mathcal{X} = \{\omega_1, \ldots, \omega_K\}$ et chaque Y_u dans un $\mathbb{R}$-espace vectoriel $\mathcal{Y}$ de dimension finie. Nous nous limiterons aux cas où $z \to p(z;\theta)$ est une densité par rapport à la mesure produit $(\nu \otimes \lambda_{\mathcal{Y}})^{\otimes|\mathcal{S}|}$, où ν est la mesure de décompte sur $\mathcal{X}$ et $\lambda_{\mathcal{Y}}$ est la mesure de Lebesgue sur $\mathcal{Y}$. Les fonctions de perte que nous utiliserons sont :

1. $L(x, \hat{x}) = \sum_{u\in\mathcal{S}} I(x_u \neq \hat{x}_u)$;
2. $L(x, \hat{x}) = I(x \neq \hat{x})$,

où $I(A) = 1$ si A est vraie et 0 sinon.
Pour la première fonction de perte, l'estimateur obtenu est celui du "Maximum des Marginales a Posteriori" (MPM). Il est défini par $(\hat{x}_{MPM})_u = \arg\max_{x_u} p(x_u|y;\theta)$ pour tout $u \in \mathcal{S}$. Pour la seconde fonction de perte, l'estimateur bayésien est celui du "Maximum A Posteriori" (MAP), défini par $\hat{x}_{MAP} = \arg\max_x p(x|y)$. Dans le cas général, le calcul des estimateurs du MPM et du MAP nécessitent de connaître toutes les probabilités a posteriori $p(x|y;\theta)$, soit $K^{|\mathcal{S}|}$ valeurs. Comme nous le verrons, lorsque le modèle est suffisamment simple, il existe des algorithmes permettant le calcul du MPM et du MAP même pour des ensembles d'indices très riches, pouvant dépasser un million d'éléments. Ces algorithmes, appelés algorithmes d'inférence bayésienne, s'appuient sur la factorisation de la loi $p(z;\theta)$. Nous introduirons la relation entre factorisation et dépendances au travers des modèles graphiques. Dans un second temps, nous aborderons les algorithmes d'inférence bayésienne dans les cas plus spécifiques des chaînes et des arbres de Markov couples [39, 67, 68, 81, 96, 98]. Pour finir, nous présenterons dans cette section deux algorithmes d'estimation du paramètre θ : l'algorithme "Expectation Maximisation" (EM) et l'algorithme "Iterative Conditional Estimation" (ICE). Le premier

est parmi les méthodes les plus utilisées dans les modèles de Markov cachés. Le deuxième, que nous retiendrons dans la suite de notre thèse, se prête mieux aux divers modèles généralisant les chaînes de Markov cachées proposés dans notre manuscrit.

2.1 Algorithmes d'inférence bayésienne et modèles graphiques de dépendance

Les modèles graphiques de dépendance abordés dans cette section sont détaillés dans [59, 60, 70].

2.1.1 Graphes de dépendance non orientés et markovianité

Nous appelons graphe un couple $G = (\mathcal{S}, \mathcal{E})$, où $\mathcal{S}$ est un ensemble fini et $\mathcal{E}$ est un sous-ensemble de $\mathcal{S} \times \mathcal{S}$. L'ensemble $\mathcal{S}$ sera appelé ensemble des sommets ou sites du graphe et $\mathcal{E}$ l'ensemble des arêtes. Le graphe sera qualifié de non orienté lorsque pour $(u, v) \in \mathcal{E}$, le couple (v, u) appartient aussi à $\mathcal{E}$, il sera qualifié d'orienté dans le cas contraire. Les graphes considérés par la suite seront non orientés. Dans un graphe non orienté, il existe une arête entre u et v si $(u, v) \in \mathcal{E}$, on dira alors que v (resp. u) est voisin de u (resp. v). L'ensemble des voisins de u sera noté $\mathcal{V}_u$ et appelé voisinage de u. On dira qu'il existe un chemin entre u et v s'il existe des sites $u_1, \ldots, u_n$ et des arêtes entre u et u_1, u_1 et u_2, ..., u_n et v. On dira alors que le chemin de u à v passe par u_k pour $k \in \{1, \ldots, n\}$. Deux ensembles de sites a et b sont séparés par un ensemble de sites c si tout chemin d'un site de a vers un site de b passe par au moins un site de c. Un sous-graphe c de G est une clique si et seulement si :

- ou bien il existe un sommet u tel que $c = (\{u\}, \{(u, u)\})$, c est alors appelé "singleton" et sera noté par abus $c = \{u\}$;
- ou bien deux sommets de c sont deux sommets mutuellement voisins dans G.

Markovianité et graphes de dépendance

Soit $G = (\mathcal{S}, \mathcal{E})$ un graphe non orienté et soit $Z = (Z_u)_{u \in \mathcal{S}}$ un ensemble indexé par $\mathcal{S}$ de variables aléatoires, qui sera appelé "champ aléatoire", à valeurs dans $\mathcal{Z} = \prod_{u \in \mathcal{S}} \mathcal{Z}_u$ muni d'une mesure de référence $\nu_{\mathcal{Z}} = \bigotimes_{u \in \mathcal{S}} \nu_{\mathcal{Z}_u}$. On distingue trois types de dépendance pouvant se déduire de la lecture d'un graphe :

- la markovianité par paires ;
- la markovianité globale ;
- la markovianité locale.

Définition 2.1.1. *Soit $G = (\mathcal{S}, \mathcal{E})$ un graphe et $Z = (Z_u)_{u \in \mathcal{S}}$ un champ aléatoire. Z satisfait vis-à-vis du graphe G :*

- *la propriété de markovianité par paires si pour tous sites u et v, s'il n'existe pas d'arête entre u et v, alors Z_u et Z_v sont indépendantes conditionnellement à l'ensemble de variables aléatoires $\{Z_t : t \notin \{u, v\}\}$;*
- *la propriété de markovianité globale si pour trois sous-ensembles a, b et c non vides et disjoints de $\mathcal{S}$, si c sépare a et b, alors les ensembles de variables aléatoires $Z_a = (Z_t)_{t \in a}$*

et $Z_b = (Z_t)_{t\in b}$ sont indépendants conditionnellement à l'ensemble de variables aléatoires $Z_c = (Z_t)_{t\in c}$;
- *la propriété de markovianité locale si pour tout u de voisinage $\mathcal{V}_u$, les ensembles de variables aléatoires $\{Z_u\}$ et $\{Z_v : v \neq u, v \notin \mathcal{V}_u\}$ sont indépendants conditionnellement à l'ensemble de variables aléatoires $\{Z_t : t \in \mathcal{V}_u\}$.*

Lorsque le champ aléatoire Z satisfait la propriété de markovianité par paires vis-à-vis du graphe G, G est appelé graphe de dépendance de Z.

Remarque : Il est important de noter que, sous l'hypothèse de markovianité globale, Z_a et Z_b peuvent être indépendants conditionnellement à Z_c sans que c ne sépare les ensembles a et b.

Proposition 2.1.1 (Equivalence des markovianités)**.** *Soit $Z = (Z_u)_{u\in\mathcal{S}}$ un champ aléatoire indexé sur l'ensemble fini de sites $\mathcal{S}$ d'un graphe G et à valeurs dans un ensemble mesurable $\mathcal{Z} = \prod_{u\in\mathcal{S}} \mathcal{Z}_u$ muni d'une mesure de référence $\nu_{\mathcal{Z}} = \bigotimes_{u\in\mathcal{S}} \nu_{\mathcal{Z}_u}$.*
- *Si Z satisfait la propriété de markovianité globale vis-à-vis de G, alors il satisfait la propriété de markovianité locale vis-à-vis de G ;*
- *si Z satisfait la propriété de markovianité locale vis-à-vis de G, alors il satisfait la propriété de markovianité par paires vis-à-vis de G.*

Soit p la densité de Z par rapport à $\nu_{\mathcal{Z}}$. Si $p(z)$ est strictement positif pour tout $z \in \mathcal{Z}$, alors les trois markovianités sont équivalentes.

Preuve.
Voir [70] pages 32-33. □

Nous donnons ci-après les deux exemples les plus couramment utilisés de modèles markoviens. Lorsque l'ensemble $\mathcal{S}$ est un sous-ensemble de $\mathbb{N}$, les champs aléatoires seront qualifiés de "chaîne" ou "processus".

Exemple 2.1.1 (Chaîne de Markov)**.** Un processus $Z = (Z_n)_{1\leq n\leq N}$ est une chaîne de Markov si pour tout $n > 1$, les ensembles $\{Z_k : k < n\}$ et $\{Z_k : k > n\}$ sont indépendants conditionnellement à Z_n.

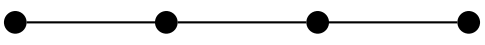

FIG. 2.1 – Graphe de dépendance d'une chaîne de Markov.

Considérons le processus marginal $Z_{1:N} = (Z_n)_{1\leq n\leq N}$. Comme $(Z_1, \ldots, Z_{N-2})$ et Z_N sont indépendants conditionnellement à Z_{N-1}, alors :

$$p(z_{1:N}) = p(z_{1:N-1}) \times p(z_N|z_1, \ldots, z_{N-1}) = p(z_{1:N-1}) \times p(z_N|z_{N-1}).$$

En réitérant le raisonnement pour le processus marginal $Z_{1:N-1}$ puis pour $Z_{1:N-2}$ et ainsi de suite, on en déduit :

$$p(z_{1:N}) = p(z_1) \prod_{n=1}^{N} p(z_{n+1}|z_n) \text{ pour tout } N \geq 1.$$

Exemple 2.1.2 (Champ de Markov sur $\mathbb{Z}^2$)**.** Soit $G = (\mathcal{S}, \mathcal{E})$ un graphe tel que $\mathcal{S} = \{1, \ldots, M\} \times \{1, \ldots, N\}$. Un champ aléatoire Z est un champ de Markov s'il satisfait la propriété de markovianité locale vis-à-vis de G.
Si les voisins d'un site (m, n) sont les sites $(m+1, n)$, $(m-1, n)$, $(m, n+1)$ et $(m, n-1)$, on obtient le graphe représenté à la figure 2.2.

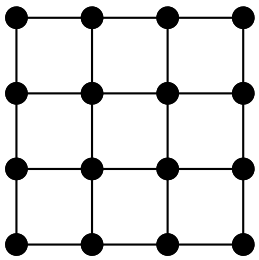

FIG. 2.2 – Graphe de dépendance d'un champ de Markov par rapport aux quatres plus proches voisins.

2.1.2 Factorisation d'une loi selon un graphe

Nous précisons dans cette sous-section les liens entre la markovianité globale d'un champ aléatoire et la factorisation de sa loi.

Définition 2.1.2 (Factorisation d'une distribution)**.** *Soit $Z = (Z_u)_{u \in \mathcal{S}}$ un champ aléatoire indexé sur l'ensemble fini des sites $\mathcal{S}$ d'un graphe G et à valeurs dans $\mathcal{Z} = \prod_{u \in \mathcal{S}} \mathcal{Z}_u$ muni de la mesure de référence $\nu_{\mathcal{Z}} = \bigotimes_{u \in \mathcal{S}} \nu_{\mathcal{Z}_u}$. La densité p de la loi de $Z = (Z_u)_{u \in \mathcal{S}}$ par rapport à la mesure $\nu_{\mathcal{Z}}$ se factorise selon le graphe $G = (\mathcal{S}, \mathcal{E})$ si elle s'écrit :*

$$p(z) = \prod_{c \in \mathcal{C}} f_c(z_c),$$

où $\mathcal{C}$ est l'ensemble des cliques du graphe et pour toute clique c, f_c est une fonction de $\prod_{u \text{ site de } c} \mathcal{Z}_u$ dans $\mathbb{R}^+$.

Proposition 2.1.2. *Si la distribution de Z se factorise selon le graphe G, alors Z satisfait la propriété de markovianité globale vis-à-vis de ce graphe.*

Preuve.
Voir [70]. □

La réciproque de cette proposition est donnée par le théorème d'Hammersley-Clifford. Rappelons tout d'abord la définition d'un champ de Gibbs.

Définition 2.1.3 (Champ de Gibbs)**.** *Soit $G = (\mathcal{S}, \mathcal{E})$ un graphe et soit $\mathcal{C}$ l'ensemble de ses cliques.*
Un champ aléatoire $Z = (Z_u)_{u\in\mathcal{S}}$ est un champ de Gibbs vis-à-vis du graphe $G = (\mathcal{S}, \mathcal{E})$ s'il existe une famille d'applications à valeurs réelles appelée "potentiel de Gibbs" $\{\phi_c : c \in \mathcal{C}\}$ telle que la densité de sa loi par rapport à une mesure de référence s'écrit :

$$p(z) \propto \exp\left(-\sum_{c\in\mathcal{C}} \phi_c(z_c)\right).$$

Si Z est un champ de Gibbs, alors il se factorise selon le graphe considéré.

Théorème 2.1.1 (Théorème d'Hammersley-Clifford)**.** *Soit $Z = (Z_u)_{u\in\mathcal{S}}$ un champ aléatoire indexé sur l'ensemble de sites $\mathcal{S}$ d'un graphe G et à valeurs dans un ensemble mesurable $\mathcal{Z} = \prod_{u\in\mathcal{S}} \mathcal{Z}_u$ muni d'une mesure de référence $\nu_{\mathcal{Z}} = \bigotimes_{u\in\mathcal{S}} \nu_{\mathcal{Z}_u}$. Soit p la densité de Z par rapport à $\nu_{\mathcal{Z}}$.*
Si Z est un champ de Gibbs vis-à-vis de G, alors il vérifie la propriété de markovianité globale vis-à-vis du graphe G.
Si $p(z)$ est strictement positif pour tout $z \in \mathcal{Z}$ et si Z satisfait la propriété de markovianité locale, alors Z est un champ de Gibbs pour le graphe G.

Preuve. Pour la preuve, voir [54]. □

Ainsi, sous la condition de positivité de la loi de Z, nous avons l'équivalence entre les quatres propriétés :
- la loi de Z se factorise selon le graphe G ;
- la loi de Z satisfait la propriété de markovianité globale ;
- la loi de Z satisfait la propriété de markovianité locale ;
- la loi de Z satisfait la propriété de markovianité par paires.

2.2 Algorithmes d'inférence bayésienne dans les modèles de Markov couples

Nous présentons les algorithmes d'inférence bayésienne dans le cas des chaînes et des arbres de Markov couples. Un champ aléatoire $Z = (X_u, Y_u)_{u\in\mathcal{S}}$ est une chaîne de Markov couple si $\mathcal{S} = \{1, \ldots, N\}$ et si le processus Z est une chaîne de Markov. On notera alors $x_{1:N}$ et $y_{1:N}$ les réalisations respectives de X et de Y. Les algorithmes d'inférence étudiés dans cette section utilisent la factorisation de la loi selon son graphe de dépendance. Il permettent ainsi, comme

nous allons le voir, le calcul rapide des estimateurs du MPM et du MAP. Le modèle de Markov couple généralise le modèle classique de Markov cachées à bruit indépendant [81, 96, 99, 104]. Il permet de modéliser certaines situations ne pouvant pas être prises en compte par ce dernier. Par ailleurs, l'égalité $p(y_n|x_{1:N}) = p(y_n|x_n)$ n'a pas toujours lieu dans les chaînes couples. Les chaînes de Markov cachées à bruit indépendant étant des chaînes couples particulières, les algorithmes d'inférence bayésienne peuvent être encore utilisés dans les chaînes de Markov cachés à bruit indépendant.

2.2.1 Algorithme de Baum-Welsh

Soit $Z = (X_n, Y_n)_{n\in\{1,\dots,N\}}$ une chaîne de Markov telle que chaque X_n prend ses valeurs dans $\mathcal{X} = \{\omega_1, \dots, \omega_K\}$ et chaque marginale Y_n prend ses valeurs dans un espace vectoriel de dimension finie $\mathcal{Y}$. L'algorithme de Baum-Welsh permet de calculer les lois a posteriori $p(x_n|y_{1:N})$ et $p(x_n, x_{n+1}|y_{1:N})$ à partir d'un échantillon observé $y_{1:N}$ de Y. Nous allons voir dans cette sous-section deux algorithmes de Baum-Welsh. La première version est la plus classique ; elle est toutefois sujette à des problèmes numériques. La deuxième version, dite "conditionnelle", permet d'éviter ces problèmes.

Algorithme de Baum-Welsh classique

L'algorithme de Baum-Welsh est constitué de deux étapes. La première, appelée "étape directe", consiste à calculer les sommations successives $\sum_{x_k} p(x_{k:N}, y_{1:N})$ pour k de 1 à n et dans la seconde, appelée "étape rétrograde", on calcule les sommations successives $\sum_{x_k} p(x_{n:k}, y_{1:N})$ pour k de N à $n+1$. On obtient ainsi les quantités $p(x_n, x_{n+1}, y_{1:N})$ et $p(x_n, y_{1:N})$. Comme Z est une chaîne de Markov, les étapes directes et rétrogrades s'écrivent facilement.

1. Etape directe :
   - initialisation : $\alpha_1(x_1) = p(x_1, y_1)$;
   - itération pour n de 1 à $N-1$:

$$\alpha_{n+1}(x_{n+1}) = \sum_{x_n} \alpha_n(x_n) p(x_{n+1}, y_{n+1}|x_n, y_n). \tag{2.1}$$

2. Etape rétrograde :
   - initialisation : $\beta_N(x_N) = 1$;
   - itération pour n de N à 2 :

$$\beta_{n-1}(x_{n-1}) = \sum_{x_n} \beta_n(x_n) p(x_n, y_n|x_{n-1}, y_{n-1}). \tag{2.2}$$

Finalement, on a :
- $p(x_n|y_{1:N}) \propto p(x_n, y_{1:N}) = \alpha_n(x_n)\beta_n(x_n)$;
- $p(x_n, x_{n+1}|y_{1:N}) \propto p(x_n, x_{n+1}, y_{1:N}) = \alpha_n(x_n)\beta_{n+1}(x_{n+1})p(x_{n+1}, y_{n+1}|x_n, y_n)$;
- $p(x_{n+1}|x_n, y_{1:N}) = \dfrac{\beta_{n+1}(x_{n+1})}{\beta_n(x_n)} p(x_{n+1}, y_{n+1}|x_n, y_n)$.

Algorithme de Baum-Welsh conditionnel

L'algorithme de Baum-Welsh conditionnel a été proposé par P. Devijver dans le cas des chaînes de Markov cachées classiques dans [40] et généralisé aux chaînes couples par S. Derrode dans [39]. Comme $\alpha_n(x_n) = p(x_n, y_{1:n})$ et $\beta_n(x_n) = p(y_{n+1:N}|x_n, y_n)$; ainsi, si le processus Y est à valeurs réelles, et si n est suffisamment grand, $\alpha_n(x_n)$ devient très petit. De même si n est suffisamment petit par rapport à N, $\beta_n(x_n)$ devient très petit. Ainsi, lorsque l'on programme l'algorithme de Baum-Welsh de cette façon, on rencontre des problèmes numériques car l'ordinateur considère comme nulle les valeurs trop petites. Afin de remédier à ce problème, nous modifions l'algorithme de Baum-Welsh en divisant les quantités $\alpha_n(x_n)$ et $\beta_n(x_n)$ par des quantités du même ordre de grandeur.
On pose :

$$\tilde{\alpha}_n(x_n) = \frac{\alpha_n(x_n)}{p(y_{1:n})} = p(x_n|y_{1:n}),$$

X_n étant à valeurs finies, cette quantité ne pose aucun problème numérique. On a alors $p(x_n, y_{1:N}) = p(y_{1:n})\tilde{\alpha}_n(x_n)\beta_n(x_n)$, ainsi $p(x_n|y_{1:N}) = \dfrac{\tilde{\alpha}_n(x_n)\beta_n(x_n)}{p(y_{n+1:N}|y_{1:n})}$. Comme $p(x_n|y_{1:N})$ ne pose aucun problème numérique, on pose :

$$\tilde{\beta}_n(x_n) = \frac{\beta_n(x_n)}{p(y_{n+1:N}|y_{1:n})} = \frac{p(y_{n+1:N}|x_n, y_n)}{p(y_{n+1:N}|y_{1:n})}.$$

L'algorithme de Baum-Welsh modifié s'écrit :

1. Etape directe :
   - initialisation : $\tilde{\alpha}_1(x_1) = p(x_1|y_1)$;
   - itération :

$$\tilde{\alpha}_{n+1}(x_{n+1}) = \frac{1}{p(y_{n+1}|y_{1:n})} \sum_{x_n} \tilde{\alpha}_n(x_n) p(x_{n+1}, y_{n+1}|x_n, y_n), \tag{2.3}$$

   et

$$p(y_{n+1}|y_{1:n}) = \sum_{x_{n+1}} \sum_{x_n} \tilde{\alpha}_n(x_n) p(x_{n+1}, y_{n+1}|x_n, y_n) \ ;$$

2. Etape rétrograde :
   - initialisation : $\tilde{\beta}_N(x_N) = 1$;
   - itération :

$$\tilde{\beta}_n(x_n) = \frac{\displaystyle\sum_{x_{n+1}} \tilde{\beta}_{n+1}(x_{n+1}) p(x_{n+1}, y_{n+1}|x_n, y_n)}{\displaystyle\sum_{x_{n+1}} \sum_{x_n} \tilde{\alpha}_n(x_n) p(x_{n+1}, y_{n+1}|x_n, y_n)} \ ; \tag{2.4}$$

On a ainsi :
- $p(x_n|y_{1:N}) = \tilde{\alpha}_n(x_n)\tilde{\beta}_n(x_n)$;
- $p(x_n, x_{n+1}|y_{1:N}) \propto \tilde{\alpha}_n(x_n)\tilde{\beta}_{n+1}(x_{n+1}) p(x_{n+1}, y_{n+1}|x_n, y_n)$;
- $p(x_{n+1}|x_n, y_{1:N}) \propto \dfrac{\tilde{\beta}_{n+1}(x_{n+1})}{\tilde{\beta}_n(x_n)} p(x_{n+1}, y_{n+1}|x_n, y_n)$.

2.2.2 Algorithme de Viterbi

L'algorithme de Viterbi permet de calculer l'estimateur du MAP de manière itérative et rapide. Soit $Z = (X_n, Y_n)_{n\in\{1,\dots,N\}}$ une chaîne de Markov couple.

– Initialisation :

$$\hat{x}_N(x_{N-1}) = \arg\max_{x_N} p(x_N, y_N|x_{N-1}, y_{N-1}) \ ;$$

et

$$\psi_{N-1}(x_{N-1}) = \max_{x_N} p(x_N, y_N|x_{N-1}, y_{N-1}) \ ;$$

– Itération (étape rétrograde) pour n de $N-1$ à 1 :

$$\hat{x}_n(x_{n-1}) = \arg\max_{x_n} [p(x_n, y_n|x_{n-1}, y_{n-1})\psi_n(x_n)] \ ;$$

et

$$\psi_{n-1}(x_{n-1}) = \max_{x_n} [p(x_n, y_n|x_{n-1}, y_{n-1})\psi_n(x_n)] \ ;$$

– Etape directe :

$$(\hat{x}_{MAP})_1 = \arg\max_{x_1} p(x_1, y_1)\psi_1(x_1) \text{ et } (\hat{x}_{MAP})_{n+1} = \hat{x}_{n+1}((\hat{x}_{MAP})_n).$$

On obtient ainsi, après les étapes "rétrograde" et "directe", $\hat{x}_{MAP}$ maximisant $p(x_{1:N}|y_{1:N})$. Dans [39], il est également proposé un algorithme de Viterbi conditionnel afin d'éviter les problèmes numériques.

2.2.3 Algorithme de Baum-Welsh adapté aux arbres de Markov

Soit $\mathcal{S}$ un ensemble fini de sites. Soit $\mathcal{S}_1, \dots, \mathcal{S}_P$ une partition de $\mathcal{S}$ telle que $\mathcal{S}_1 = \{1\}$, $|\mathcal{S}_1| \leq |\mathcal{S}_2| \leq \dots |\mathcal{S}_P|$. A chaque $u \in \mathcal{S}_k$ pour $k \neq 1$, il existe un unique élément noté $\rho(u)$ dans $\mathcal{S}_{k-1}$. Cet élément est appelé parent de u. Les éléments v tels que u soit parent de v sont appelés fils de u, on note $c(u)$ leur ensemble. Les voisins d'un site u sont le site parent $\rho(u)$ et l'ensemble des fils $c(u)$. L'ensemble $\mathcal{E}$ des arêtes est l'ensemble des couples (u, v) tels que $u \in c(v)$ ou $v \in c(u)$. Un champ aléatoire $Z = (Z_u)_{u\in\mathcal{S}}$ est un arbre de Markov si son graphe de dépendance est $(\mathcal{S}, \mathcal{E})$; sa distribution s'écrit alors :

$$p(z) = p(z_1) \prod_{u\neq 1} p(z_u|z_{\rho(u)}).$$

Considérons $Z = (X_u, Y_u)_{u\in\mathcal{S}}$ un arbre de Markov couple. Comme pour les chaînes couples, les arbres de Markov couples généralisent les arbres de Markov cachés dans lesquels le champ caché X est un arbre de Markov. Dans [94], on donne des conditions pour qu'un arbre de Markov couple soit un arbre de Markov caché.
L'algorithme de Baum-Welsh dans le cas des arbres se déroule en deux temps appelés "récursion ascendante" et "récursion descendante" et fonctionne de la manière suivante.

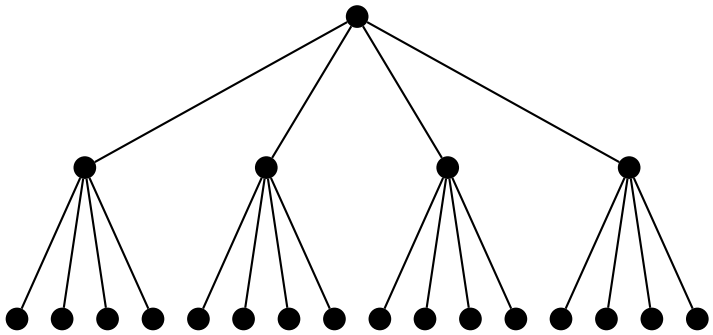

FIG. 2.3 – Graphe de dépendance d'un arbre de Markov.

1. Etape ascendante :
   - initialisation :
     Pour $u \in \mathcal{S}_P$, $\beta_u(x_u) = 1$;
   - itération pour k de $P-1$ à 1 :
     Pour $u \in \mathcal{S}_k$ et $t \in c(u)$, $\beta_{t,u}(x_u) = \sum_{x_t} \beta_t(x_t) p(x_t, y_t | x_u, y_u)$ et $\beta_u(x_u) = \prod_{t \in c(u)} \beta_{t,u}(x_u)$.
2. Etape descendante :
   - initialisation :
     $\alpha_1(x_1) = p(x_1, y_1)$;
   - itération pour k de 2 à P :
     Pour $u \in \mathcal{S}_k$, $\alpha_u(x_u) = \sum_{x_{\rho(u)}} \alpha_{\rho(u)}(x_{\rho(u)}) \frac{\beta_{\rho(u)}(x_{\rho(u)})}{\beta_{u,\rho(u)}(x_{\rho(u)})} p(x_u, y_u | x_{\rho(u)}, y_{\rho(u)})$.

La distribution $p(x|y)$ est également une distribution markovienne et on a :
- $p(x_u|y) \propto p(x_u, y) = \alpha_u(x_u)\beta_u(x_u)$;
- $p(x_u|x_{\rho(u)}, y) = \frac{\beta_u(x_u)}{\beta_{u,\rho(u)}(x_{\rho(u)})} p(x_u, y_u | x_{\rho(u)}, y_{\rho(u)})$.

La version conditionnelle de l'algorithme de Baum-Welsh dans le cas des arbres figure dans [49].

Nous n'étudierons pas les modèles d'arbre de Markov dans cette thèse. En revanche, tous les modèles de chaînes abordés dans cette thèse peuvent se généraliser au cas des arbres.
Dans la section suivante, nous abordons l'estimation des paramètres d'un modèle à données latentes $p(z;\theta)$.

2.3 Estimation des paramètres

2.3.1 Algorithme EM

L'algorithme "Expectation Maximisation" (EM) est parmi les plus utilisés pour estimer les paramètres d'une chaîne de Markov cachée ; ainsi que ceux de différents autres modèles à données latentes. Cet algorithme est issu des travaux de A. P. Dempster, N. M. Laird et D. B. Rubin [38] sur l'estimation par maximum de vraisemblance à partir de données incomplètes. Nous allons exposer dans cette sous-section l'algorithme EM dans sa généralité puis nous détaillerons l'algorithme EM dans le cas des chaînes de Markov cachées à bruit indépendant

de type exponentiel.
On considère un couple de processus $(X,Y)=(X_u,Y_u)_{u\in\mathcal{S}}$ tel que chaque X_u soit à valeurs dans $\mathcal{X}=\{\omega_1,\ldots,\omega_K\}$ et chaque Y_u à valeurs dans un espace vectoriel de dimension finie $\mathcal{Y}$. On notera $p(x,y;\theta)$ la distribution de (X,Y) de paramètre θ. Nous observons la réalisation y de Y et nous devons estimer le paramètre θ.
Soit

$$Q(\theta|\theta_q)=\mathbb{E}_{\theta_q}\left(\log\left(p(X,y;\theta)\right)|y\right)=\sum_{x\in\mathcal{X}^{|\mathcal{S}|}}\log\left(p(x,y;\theta)\right)p(x|y;\theta_q), \tag{2.5}$$

l'espérance sachant $Y=y$ de la log-vraisemblance en données complètes lorsque le paramètre vaut θ_q. Le but de l'algorithme EM est de construire une suite $(\theta_q)_{q\in\mathbb{N}}$ telle que :

$$\log\left(p(y;\theta_{q+1})\right)\geq\log\left(p(y;\theta_q)\right).$$

On a :

$$\log\left(p(y;\theta)\right)=Q(\theta|\theta_q)-\sum_{x\in\mathcal{X}^{|\mathcal{S}|}}\log\left(p(x|y;\theta)\right)p(x|y;\theta_q).$$

La fonction $\theta\rightarrow\sum_{x\in\mathcal{X}^{|\mathcal{S}|}}\log\left(p(x|y;\theta)\right)p(x|y;\theta_q)$ est maximale pour $\theta=\theta_q$, soit :

$$\sum_{x\in\mathcal{X}^{|\mathcal{S}|}}\log\left(p(x|y;\theta)\right)p(x|y;\theta_q)\leq\sum_{x\in\mathcal{X}^{|\mathcal{S}|}}\log\left(p(x|y;\theta_q)\right)p(x|y;\theta_q).$$

Ainsi

$$\log\left(p(y;\theta)\right)\geq Q(\theta|\theta_q)-\sum_{x\in\mathcal{X}^{|\mathcal{S}|}}\log\left(p(x|y;\theta_q)\right)p(x|y;\theta_q),$$

et donc si $\theta_{q+1}=\arg\max Q(\theta|\theta_q)$, on en déduit :

$$\begin{aligned}\log\left(p(y;\theta_{q+1})\right) &\geq Q(\theta_{q+1}|\theta_p)-\sum_{x\in\mathcal{X}^{|\mathcal{S}|}}\log\left(p(x|y;\theta_q)\right)p(x|y;\theta_q),\\ &\geq Q(\theta_q|\theta_q)-\sum_{x\in\mathcal{X}^{|\mathcal{S}|}}\log\left(p(x|y;\theta_q)\right)p(x|y;\theta_q)=\log\left(p(y;\theta_q)\right).\end{aligned}$$

L'algorithme EM fonctionne de la manière suivante :
- étape E (Expectation) :
  calcul de $Q(\theta|\theta_q)=\mathbb{E}_{\theta_q}\left(\log\left(p(X,y;\theta)\right)|y\right)$;
- étape M (Maximisation) :
  maximisation de $\theta\rightarrow Q(\theta|\theta_q)$.

Il suffit en fait de choisir θ_{q+1} tel que $Q(\theta_{q+1}|\theta_q)\geq Q(\theta_q|\theta_q)$ pour avoir $\log\left(p(y;\theta_{q+1})\right)\geq\log\left(p(y;\theta_q)\right)$. L'algorithme est alors appelé "Generalized Expectation Maximisation"(GEM). Dans [79], on peut trouver d'autres versions de l'algorithme EM. Parmi celles-ci, on peut citer l'algorithme "Stochastic Expectation Maximisation" (SEM) [20, 29] qui est une version stochastique de l'algorithme EM. L'objectif de l'algorithme SEM est d'éviter la convergence de la suite $(\theta_q)_{q\in\mathbb{N}}$ vers un maximum local de $\theta\rightarrow\log\left(p(y;\theta)\right)$ en introduisant une perturbation stochastique.

Donnons maintenant un exemple illustrant le fonctionnement de l'algorithme EM. On considère une chaîne Markov cachée $(X, Y) = (X_n, Y_n)_{1 \leq n \leq N}$ classique avec la distribution :

$$p(x_{1:N}, y_{1:N}) = p(x_1)p(y_1|x_1) \prod_{n=1}^{N-1} p(x_{n+1}|x_n)p(y_{n+1}|x_{n+1}).$$

Prenons le cas particulier où $\mathcal{Y} = \mathbb{R}$ et $p(y_n|x_n = \omega_j)$ est une loi de type exponentiel de paramètre fonctionnel a et de paramètre scalaire λ_j, les détails sur les modèles exponentiels sont donnés au chapitre 4. Nous avons :

$$p(y_n|x_n = \omega_j) = \frac{1}{Z(\lambda_j)} \exp\left(\lambda_j a(y_n)\right), \text{ avec } \int_{\mathcal{Y}} \exp\left(\lambda_j a(y_n)\right) dy_n = Z(\lambda_j).$$

Le paramètre θ que l'on cherche à estimer a pour composantes les paramètres de $p(x_{1:N})$, qui sont la loi initiale $p(x_1)$ et la transition $p(x_{n+1}|x_n)$ que l'on supposera indépendante de n, et le paramètre λ_j de $p(y_n|x_n = \omega_j)$, également supposé indépendant de n.
Considérons l'étape E (on omettra le paramètre θ par mesure de simplicité).
On a :

$$\begin{aligned} \log\left(p(X, y_{1:N})\right) &= \log\left(p(X_1)\right) + \sum_{n=1}^{N-1} \log\left(p(X_{n+1}|X_n)\right) \\ &+ \sum_{n=1}^{N} \log\left(p(y_n|X_n)\right). \end{aligned}$$

Ainsi, en intégrant sous la loi a posteriori $p(x_{1:N}|y_{1:N}; \theta_q)$ et sachant que les quantités $p(x_{n+1} = \omega_j|x_n = \omega_i)$ ne dépendent pas de n, on a :

$$\begin{aligned} Q(\theta|\theta_q) &= \sum_{j=1}^{K} \log\left(p(x_1 = \omega_j)\right) p(x_1 = \omega_j|y_{1:N}; \theta_q) \\ &+ \sum_{i=1}^{K} \sum_{j=1}^{K} \sum_{n=1}^{N-1} \log\left(p(x_{n+1} = \omega_j|x_n = \omega_i)\right) p(x_n = \omega_i, x_{n+1} = \omega_j|y_{1:N}; \theta_q) \\ &+ \sum_{j=1}^{K} \sum_{n=1}^{N} \log\left(p(y_n|x_n = \omega_j)\right) p(x_n = \omega_j|y_{1:N}; \theta_q). \end{aligned}$$

L'étape M consiste alors à maximiser cette quantité. On notera $p(x_1; \theta_{q+1})$, $p(x_{n+1}|x_n; \theta_{q+1})$ la loi initiale et la transition de X sous le paramètre θ_{q+1} et $\lambda_{q+1,j}$ les paramètres de $p(y_n|x_n = \omega_j; \theta_{q+1})$. Les composantes de θ_{q+1} sont alors les K valeurs $(p(x_1 = \omega_j; \theta_{q+1}))_{j \in \{1,\ldots,K\}}$, les K^2 valeurs $(p(x_{n+1} = \omega_j|x_n = \omega_i; \theta_{q+1}))_{(i,j) \in \{1,\ldots,K\}^2}$ et les K valeurs $(\lambda_{q+1,j})_{j \in \{1,\ldots,K\}}$. Nous devons maximiser chacun des trois termes de la somme. La quantité

$$\sum_{j=1}^{K} \log\left(p(x_1 = \omega_j)\right) p(x_1 = \omega_j|y_{1:N}; \theta_q)$$

est maximale pour :

$$p(x_1 = \omega_j; \theta_{q+1}) = p(x_1 = \omega_j|y_{1:N}; \theta_q).$$

Soit $i \in \{1, \ldots, K\}$, la quantité

$$\sum_{j=1}^{K} \sum_{n=1}^{N-1} \log\left(p(x_{n+1} = \omega_j | x_n = \omega_i)\right) p(x_n = \omega_i, x_{n+1} = \omega_j | y_{1:N}; \theta_q) \quad =$$
$$\sum_{j=1}^{K} \log\left(p(x_{n+1} = \omega_j | x_n = \omega_i)\right) \sum_{n=1}^{N-1} p(x_n = \omega_i, x_{n+1} = \omega_j | y_{1:N}; \theta_q)$$

est maximale pour :

$$p(x_{n+1} = \omega_j | x_n = \omega_i; \theta_{q+1}) = \frac{\sum_{n=1}^{N-1} p(x_n = \omega_i, x_{n+1} = \omega_j | y_{1:N}; \theta_q)}{\sum_{n=1}^{N-1} p(x_n = \omega_i | y_{1:N}; \theta_q)}.$$

L'algorithme EM nécessite de connaître les probabilités a posteriori $p(x_n = \omega_i | y_{1:N}; \theta_q)$ et $p(x_n = \omega_i, x_{n+1} = \omega_j | y_{1:N}; \theta_q)$ qui sont calculées par l'algorithme de Baum-Welsh.
Il reste la maximisation du dernier terme. Dans le cas des modèles exponentiels, l'étape M et la maximisation de la vraisemblance sont analogues.
En effet, soit $y_{1:N}$ un échantillon d'une loi de type exponentiel de densité $p(y; \lambda) = \frac{1}{Z(\lambda)} \exp(\lambda a(y))$, alors l'estimateur du maximum de vraisemblance $\hat{\lambda}_{MV}(y_{1:N})$ est solution de l'équation de vraisemblance :

$$\frac{\partial}{\partial \lambda} \log\left(Z(\lambda)\right) = \frac{1}{N} \sum_{n=1}^{N} a(y_n).$$

La maximisation de chacune des quantités

$$\sum_{n=1}^{N} \log\left(p(y_n | x_n = \omega_j)\right) p(x_n = \omega_j | y_{1:N}; \theta_q) =$$
$$-\log\left(Z(\lambda_j)\right) \sum_{n=1}^{N} p(x_n = \omega_j | y_{1:N}; \theta_q) + \lambda_j \sum_{n=1}^{N} a(y_n) p(x_n = \omega_j | y_{1:N}; \theta_q).$$

se fait en résolvant les équations :

$$\frac{\partial}{\partial \lambda_j} \log\left(Z(\lambda_j)\right) = \frac{1}{\sum_{n=1}^{N} p(x_n = \omega_j | y_{1:N}; \theta_q)} \times \sum_{n=1}^{N} a(y_n) p(x_n = \omega_j | y_{1:N}; \theta_q).$$

Considérons les exemples suivants de modèles exponentiels, avec les équations de vraisemblance correspondantes.

- Loi exponentielle standard de densité sur $\mathbb{R}^+$ donnée par $p(y) = \lambda \exp(-\lambda y)$.
  On a $Z(\lambda) = \frac{1}{\lambda}$ et $a(y) = -y$, ainsi l'estimateur du maximum de vraisemblance $\hat{\lambda}_{MV}$ et

l'estimée $\lambda_{q+1,j}$ sont donnés par :

$$\text{MV} : \quad \hat{\lambda}_{MV} = \frac{N}{\sum_{n=1}^{N} y_n}.$$

$$\text{EM} : \quad \lambda_{q+1,j} = \frac{\sum_{n=1}^{N} p(x_n = \omega_j | y_{1:N}; \theta_q)}{\sum_{n=1}^{N} y_n p(x_n = \omega_j | y_{1:N}; \theta_q)}.$$

– Loi normale $\mathcal{N}_{\mathbb{R}}(m, s)$, les estimations de la moyenne et de la variances sont données par :

$$\text{MV} : \begin{cases} \hat{m}_{MV} = \dfrac{1}{N}\displaystyle\sum_{n=1}^{N} y_n, \\ \hat{s}_{MV} = \dfrac{1}{N}\displaystyle\sum_{n=1}^{N} (y_n - \hat{m}_{MV})^2, \end{cases}$$

$$\text{EM} : \begin{cases} m_{q+1,j} = \dfrac{\displaystyle\sum_{n=1}^{N} y_n p(x_n = \omega_j | y_{1:N}; \theta_q)}{\displaystyle\sum_{n=1}^{N} p(x_n = \omega_j | y_{1:N}; \theta_q)}, \\ s_{q+1,j} = \dfrac{\displaystyle\sum_{n=1}^{N} (y_n - m_{q+1,j})^2 p(x_n = \omega_j | y_{1:N}; \theta_q)}{\displaystyle\sum_{n=1}^{N} p(x_n = \omega_j | y_{1:N}; \theta_q)}. \end{cases}$$

– Loi $\Gamma(a, b)$ de densité sur $\mathbb{R}^+$ donnée par $p(y; a, b) = \dfrac{1}{\Gamma(a) b^a} y^{a-1} \exp\left(-\dfrac{y}{b}\right)$, le paramétrage canonique est donné par :

$$\begin{aligned} a(y) &= \begin{pmatrix} \log(y) \\ -y \end{pmatrix}, \\ \lambda(a, b) &= \begin{pmatrix} a - 1 \\ \frac{1}{b} \end{pmatrix} = \begin{pmatrix} \lambda^{(1)} \\ \lambda^{(2)} \end{pmatrix}, \\ \text{et } \log\left(Z(\lambda)\right) &= \log\left(\Gamma(\lambda^{(1)} + 1)\right) - (\lambda^{(1)} + 1) \log(\lambda^{(2)}). \end{aligned}$$

Ainsi :

$$\text{MV} : \begin{cases} \psi(\hat{\lambda}^{(1)}_{MV}+1) - \log(\hat{\lambda}^{(2)}_{MV}) = \dfrac{1}{N}\displaystyle\sum_{n=1}^{N}\log(y_n), \\ \dfrac{\hat{\lambda}^{(1)}_{MV}+1}{\hat{\lambda}^{(2)}_{MV}} = \dfrac{1}{N}\displaystyle\sum_{n=1}^{N} y_n, \end{cases}$$

$$\text{EM} : \begin{cases} \psi(\lambda^{(1)}_{q+1,j}+1) - \log(\lambda^{(2)}_{q+1,j}) = \dfrac{\displaystyle\sum_{n=1}^{N}\log(y_n)p(x_n=\omega_j|y_{1:N};\theta_q)}{\displaystyle\sum_{n=1}^{N}p(x_n=\omega_j|y_{1:N};\theta_q)}, \\ \dfrac{\lambda^{(1)}_{q+1,j}+1}{\lambda^{(2)}_{q+1,j}} = \dfrac{\displaystyle\sum_{n=1}^{N} y_n p(x_n=\omega_j|y_{1:N};\theta_q)}{\displaystyle\sum_{n=1}^{N}p(x_n=\omega_j|y_{1:N};\theta_q)}, \end{cases}$$

où ψ est la fonction digamma, dérivée logarithmique de la fonction Γ (voir Annexe A). Si on exprime ces équations avec le paramétrage (a, b), on trouve :

$$\text{MV} : \begin{cases} \psi(\hat{a}_{MV}) + \log(\hat{b}_{MV}) = \dfrac{1}{N}\displaystyle\sum_{n=1}^{N}\log(y_n), \\ \hat{a}_{MV}\hat{b}_{MV} = \dfrac{1}{N}\displaystyle\sum_{n=1}^{N} y_n, \end{cases}$$

$$\text{EM} : \begin{cases} \psi(a_{q+1,j}) + \log(b_{q+1,j}) = \dfrac{\displaystyle\sum_{n=1}^{N}\log(y_n)p(x_n=\omega_j|y_{1:N};\theta_q)}{\displaystyle\sum_{n=1}^{N}p(x_n=\omega_j|y_{1:N};\theta_q)}, \\ a_{q+1,j}b_{q+1,j} = \dfrac{\displaystyle\sum_{n=1}^{N} y_n p(x_n=\omega_j|y_{1:N};\theta_q)}{\displaystyle\sum_{n=1}^{N}p(x_n=\omega_j|y_{1:N};\theta_q)}. \end{cases}$$

Malgré l'intérêt indéniable de l'algorithme EM et son très bon comportement dans nombreuses situations réelles, notons qu'il présente également des faiblesses qui vont en partie motiver l'utilisation de l'algorithme ICE. Tout d'abord, la suite $(\theta_q)_{q\in\mathbb{N}}$ construite par EM ne converge pas obligatoirement vers le maximum global de la vraisemblance de Y et il n'existe pas de théorèmes généraux donnant des conditions d'une telle convergence. De plus, l'étape de maximisation peut être délicate, notamment dans certains modèles comprenant des lois Γ ou K.

2.3.2 Algorithme ICE

Soit $Z = (X_u, Y_u)_{u\in\mathcal{S}}$ un modèle à données latentes tel que chaque X_u prend ses valeurs dans l'ensemble fini $\mathcal{X} = \{\omega_1, \ldots, \omega_K\}$ et chaque Y_u dans un espace vectoriel de dimension finie $\mathcal{Y}$. Soit $\theta = (\theta_1, \ldots, \theta_k) \in \mathbb{R}^k$ le paramètre de $p(z;\theta)$. Pour utiliser ICE, on a besoin de deux conditions :

- l'existence d'un estimateur $T = (T_1, \ldots, T_k)$ explicite à partir des données complètes ;
- la possibilité de simulation, pour tout θ, de X selon $p(x|y;\theta)$.

L'algorithme ICE est itératif et fonctionne de la manière suivante :

1. donnée d'un paramètre initial θ_0 ;
2. à partir de θ_q, on calcule :

$$\theta_{q+1,j} = \mathbb{E}_{\theta_q}\left(T_j(X,y)|y\right),$$

pour les composantes T_j pour lesquelles ce calcul est possible. Pour les autres composantes T_i, on pose :

$$\theta_{q+1,i} = \frac{\sum_{l=1}^{L} T_i(x^l, y)}{L},$$

où $x^1, \ldots, x^L$ sont simulés selon $p(x|y;\theta_q)$.

Nous voyons que ICE fonctionne sous des hypothèses très faibles et il n'existe pas de problème de maximisation. Notons que l'estimateur T peut être l'estimateur de vraisemblance ou pas. Soit $Z = (X_n, Y_n)_{1\leq n\leq N}$ une chaîne de Markov cachée à bruit indépendant de distribution :

$$p(x_{1:N}, y_{1:N}) = p(x_1)p(y_1|x_1)\prod_{n=1}^{N-1} p(x_{n+1}|x_n)p(y_{n+1}|x_{n+1}).$$

On supposera dans cet exemple que $p(y_n|x_n = \omega_j)$ est la densité d'une loi $\Gamma(a_j, b_j)$:

$$p(y_n|x_n = \omega_j) = \frac{1}{\Gamma(a_j)b_j^{a_j}} y_n^{a_j-1}\exp\left(-\frac{y_n}{b_j}\right).$$

Nous supposerons que la chaîne X est stationnaire et réversible, ainsi $p(x_n = \omega_i, x_{n+1} = \omega_j) = p(x_n = \omega_j, x_{n+1} = \omega_i)$ et est indépendant de n. Les paramètres du modèle sont :

- les $K^2 - 1$ paramètres $p(x_n = \omega_i, x_{n+1} = \omega_j)$;
- les $2K$ paramètres (a_j, b_j) de la loi $p(y_n|x_n = \omega_j)$.

L'estimateur à partir des données complètes est

$$T = \left((\hat{R}_{i,j}(x_{1:N}, y_{1:N}))_{1\leq i\leq K, 1\leq j\leq K}, (\hat{a}_j(x_{1:N}, y_{1:N}))_{1\leq j\leq K}, (\hat{b}_j(x_{1:N}, y_{1:N}))_{1\leq j\leq K}\right),$$

où $\hat{R}_{i,j}(x_{1:N}, y_{1:N})$ est l'estimateur de $p(x_n = \omega_i, x_{n+1} = \omega_j)$, $\hat{a}_j(x_{1:N}, y_{1:N})$ et $\hat{b}_j(x_{1:N}, y_{1:N})$ sont les estimateurs de a_j et b_j. Les estimateurs $\hat{a}_j(x_{1:N}, y_{1:N})$ et $\hat{b}_j(x_{1:N}, y_{1:N})$ sont donnés par :

$$\hat{b}_j(x_{1:N}, y_{1:N}) = \frac{\hat{s}_j}{\hat{m}_j} \text{ et } \hat{a}_j(x_{1:N}, y_{1:N}) = \frac{\hat{m}_j}{\hat{b}_j},$$

où

$$\begin{aligned}\hat{m}_j &= \frac{\sum_{n=1}^{N} y_n I(x_n = \omega_j)}{\sum_{n=1}^{N} I(x_n = \omega_j)}, \\ \hat{s}_j &= \frac{\sum_{n=1}^{N} (y_n - \hat{m}_j)^2 I(x_n = \omega_j)}{\sum_{n=1}^{N} I(x_n = \omega_j)}.\end{aligned}$$

Enfin, $\hat{R}_{i,j}(x_{1:N}, y_{1:N})$ est classiquement donné par :

$$\hat{R}_{i,j}(x_{1:N}, y_{1:N}) = \frac{1}{2(N-1)} \sum_{n=1}^{N-1} \left[I(x_n = \omega_i, x_{n+1} = \omega_j) + I(x_n = \omega_j, x_{n+1} = \omega_i) \right],$$

où $I(A) = 1$ si A est vraie et 0 sinon.
L'espérance $\theta_{q+1} = \mathbb{E}_{\theta_q}\left(T(X, y_{1:N})|y_{1:N}\right)$ est calculable pour les composantes $\hat{R}_{i,j}$, ce qui donne :

$$\begin{aligned}p(x_n = \omega_i, x_{n+1} = \omega_j; \theta_{q+1}) &= \frac{1}{2(N-1)} \sum_{n=1}^{N-1} \left[p(x_n = \omega_i, x_{n+1} = \omega_j | y_{1:N}; \theta_q) \right. \\ &+ \left. p(x_{n+1} = \omega_i, x_n = \omega_j | y_{1:N}; \theta_q) \right].\end{aligned}$$

L'espérance $\theta_{q+1} = \mathbb{E}\left(T(X, y_{1:N})|y_{1:N}; \theta_q\right)$ n'est pas calculable pour les composantes a_j et b_j. Ainsi, on simule L réalisations $x^{(1)}, \ldots, x^{(L)}$ de X selon la loi a posteriori $p(x_{1:N}|y_{1:N}; \theta_q)$. Ensuite, pour chaque l de 1 à L, on calcule les estimées $\hat{a}_j^{(l)} = \hat{a}_j(x_{1:N}^{(l)}, y_{1:N})$ et $\hat{b}_j^{(l)} = \hat{b}_j(x_{1:N}^{(l)}, y_{1:N})$. Pour finir, on pose :

$$a_{q+1,j} = \frac{1}{L} \sum_{l=1}^{L} \hat{a}_j^{(l)} \text{ et } b_{q+1,j} = \frac{1}{L} \sum_{l=1}^{L} \hat{b}_j^{(l)}.$$

Les algorithmes ICE et EM se valent dans les cas classiques de chaînes de Markov cachées à bruit indépendant et gaussien [14].

Conclusion

Dans ce chapitre, nous avons présenté les principaux algorithmes d'inférence bayésienne permettant d'estimer les états inobservés dans le contexte de chaînes et d'arbres de Markov cachés. Ces algorithmes permettent le calcul rapide des lois marginales a posteriori, ce qui rend possible la mise en place de l'estimateur du MPM. Nous avons également présenté deux algorithmes généraux d'estimation des paramètres, qui sont EM et ICE. Dans la suite, nous utiliserons sauf mention contraire l'algorithme ICE. En effet, dans certains modèles, la loi des observations conditionnellement aux états cachés est complexe et l'étape M de

l'algorithme EM peut être délicate. Par ailleurs, le principe de ICE nous permettra d'en proposer une extension dans le cas des bruits à mémoire longue étudiés dans le chapitre 5. De plus, nous montrerons au travers de simulations présentées au cours de cette thèse le bon comportement de l'algorithme ICE, qui s'avère capable d'estimer correctement les paramètres en présence de bruits importants. Joint aux estimateurs du MAP ou du MPM, la méthode ICE permet ainsi, une fois le modèle fixé, de proposer des algorithmes de recherche des états cachés de manière automatique, ce qui revêt une importance primordiale dans de nombreuses applications. Concernant la convergence de l'algorithme ICE, des premiers résultats ont été démontrés dans [101]. Une autre approche abordée dans la thèse de N. Brunel [21] est celle des fonctions "estimantes". La théorie des fonctions estimantes développée dans [57] permet de réunir dans le même formalisme de nombreuses méthodes d'estimation.