Chaînes de Markov et génome

Etienne Pardoux

Leçon 4

Nous allons présenter de façon très sommaire l'algorithme SEM, qui est le plus utile dans les situations que nous décrirons plus loin. Pour chaque valeur du paramètre inconnu θ, on considère la loi conditonnelle des états cachés, sachant la suite des nucléotides, notée

$$\mathbb{P}_\theta \left(Y_1 = y_1, \ldots, Y_N = y_N | X_1 = x_1, \ldots, X_N = x_N \right),$$

ou plutôt

$$\mathbb{P}_\theta \left(Y_1^N = y_1^N | X_1^N = x_1^N \right).$$

L'algorithme SEM est un algorithme itératif, que l'on initie avec une valeur θ_0. L'itération qui remplace θ_n par θ_{n+1} se décompose en deux étapes comme suit :

- *Simulation* On tire au hasard une réalisation de la suite aléatoire Y_1^N, suivant la loi $\mathbb{P}_{\theta_n}(Y_1^N = \cdot | X_1^N = x_1^N)$. Notons $y_1^N(n)$ la suite obtenue ainsi.
- *Ré–estimation* On choisit

$$\theta_{n+1} = \text{argmax}_\theta \mathbb{P}_\theta \left(Y_1^N = y_1^N(n), X_1^N = x_1^N \right).$$

Dans l'algorithme EM, l'étape de simulation est remplacée par lme calcul de $\mathbb{E}_{\theta_n}(Y_1^N | X_1^N = x_1^N)$.

12 Modèle semi–markovien caché

12.1 Les limites du modèle de Markov caché

On a vu ci–dessus à la section 7.1 que les temps de séjour d'une chaîne de Markov dans chacun des états visité suivent des lois géométriques. Le modèle de la section 11 implique donc que les longueurs des plages codantes et non codantes d'un génome prokaryote suivent des lois géométriques. Or cette hypothèse ne cadre pas avec les données dont on dispose. Il y

a là un premier argument pour envisager un modèle plus général, mais on va voir maintenant un argument encore plus convainquant pour abandonner le modèle de Markov caché.

Examinons plus précisément notre problème, en nous limitant à nouveau pour simplifier au génome prokaryote. Il est bien sûr essentiel de prendre en compte l'information contenue dans les codons START et STOP. Si l'on renonce à un modèle phasé, on est obligé d'introduire 3 états START, 3 états codants et 3 états STOP, chacun correspondant à une des trois phases de lecture, le tout doit être multiplié par deux pour tenir compte du brin complémentaire. On ajoute un état non codant. Cela fait en tout 19 états. Certes, la plupart des termes de la matrice de transition sont nuls, mais cela fait quand même beaucoup d'états, et dans le cas eukaryote la situation est bien pire. On peut réduire ce nombre avec un modèle phasé, mais on récupère la même complexité en multipliant par trois le nombre de matrices de transition à estimer. Enfin on pourrait penser travailler sur la suite des codons plutôt que sur celle des nucléotides, mais ceci ne serait pas valable pour les parties non codantes.

On va voir ci–dessous que le modèle semi–markovien permet de réduire le nombre d'états à trois dans le cas prokaryote, en outre qu'il permet de choisir une loi plus réaliste que la loi géométrique pour la longueur des plages codantes.

12.2 Qu'est–ce qu'une chaîne semi–markovienne?

La réponse dépend des auteurs. Je vais donner ma définition. Comme son nom l'indique, une chaîne semi–markovienne est "un peu moins markovienne" (i.e. oublie un peu moins son passé) qu'une chaîne de Markov. Étant donnée une suite aléatoire $(X_1,\ldots,X_N)$, et $1 < n < N$, on définit pour chaque $1 < n < N$ la v. a. η_n de la façon suivante

$$\eta_n = \sup\{k \geq 0, X_{n-k} = X_{n-k+1} = \cdots = X_n\}.$$

Dans l'application qui nous intéresse, c'est le nombre de sites à gauche du site n, qui sont dans la même plage que celui–ci. Bien entendu, si l'on connaît la réalisation de la suite $(X_1,\ldots,X_n)$, on connaît la valeur de η_n. On notera $\varphi_n(x_1,\ldots,x_n)$ la valeur de η_n quand $(X_1,\ldots,X_n) = (x_1,\ldots,x_n)$.

Définition 12.1 *Une suite aléatoire $(X_1,\ldots,X_N)$ à valeurs dans E est une chaîne semi–markovienne ssi pour tout $1 < n \leq N$, pour tout $(x_1,\ldots,x_{n-1},x,y) \in E^{n+1}$,*

$$\begin{aligned}&\mathbb{P}(X_{n+1} = y | X_1 = x_1,\ldots,X_{n-1} = x_{n-1},X_n = x)\\ &\quad = \mathbb{P}(X_{n+1} = y | X_n = x,\eta_n = \varphi_n(x_1,\ldots,x_{n-1},x)).\end{aligned}$$

Le fait que l'état suivant d'une chaîne semi–markovienne dépende non seulement de l'état courant, mais de la longueur de la "visite" dans cet état jusqu'au site considéré fait que les lois des temps de séjour dans les divers états sont complètement arbitraires.

Plus précisément, une façon "générique" de préciser la loi d'une chaîne semi–markovienne (et aussi d'indiquer comment la simuler) est la suivante.

– D'une part on associe à chaque point $x \in E$ une loi de probabilité $(d_x(n), n \in \mathbb{N}\backslash\{0\})$ sur les entiers privés de 0, qui précise les lois des longueurs des "plages" sur lesquelles la chaîne est constante.

– D'autre part on se donne une matrice de transition P sur $E \times E$ d'une chaîne de Markov, dont tous les termes diagonaux sont nuls. Cette matrice décrit suivant quelle loi la chaîne change d'état, quand elle en change.

Voyons comment simuler une chaîne semi–markovienne dont la loi est caractérisée par les données : pour tout $x \in E$, d_x désigne la loi du temps de séjour à l'état x, et $P_{x\cdot}$ la loi du prochain état visité après l'état x. Si x est le point de départ ($X_1 = x$), on tire une variable alétoire T_1 à valeurs dans $\mathbb{N}^*$ de loi d_x. Notons n la valeur simulée. Alors $X_1 = X_2 = X_3 = \cdot = X_n = x$. On tire une v.a. Z_1 de loi $P_{x\cdot}$ sur $E\backslash\{x\}$. Supposons que le résultat du tirage soit $Z_1 = y$. Alors $X_{n+1} = y$, et on recommence en tirant une v. a. T_2 de loi d_y, et une v.a. Z_2 de loi $P_{y\cdot}$, et ainsi de suite. Tous les tirages successifs sont bien entendu indépendants les uns des autres.

12.3 Le modèle semi–markovien caché

Encore une fois, limitons–nous pour simplifier au cas prokaryote. On considère 3 états cachés, l'état 0 pour *non codant*, l'état 1 pour *codant à l'endroit*, l'état 2 pour *codant à l'envers*. L'état 0 est un état *markovien* (on verra ci–dessous la raison de cette restriction), ce qui veut dire que la loi des longueurs des plages non codantes est une loi géométrique de paramètre q (à estimer). Les états 1 et 2 sont dits *semi–markoviens*. On choisira comme loi des longueurs des plages codantes à l'endroit et à l'envers l'image par l'application $x \to 3x$ d'une loi binomiale négative de paramètres $m \in \mathbb{N}^*$ et $0 < p < 1$ (i. e. cette loi décrit le nombre de codons plutôt que le nombre de nucléotides).

Définition 12.2 *On dit que la v.a. T suit la loi binomiale négative de paramètres m et p si T est le nombre de jets de pile ou face nécessaires pour obtenir exactement m piles, p désignant la probabilité d'obtenir pile à chaque coup. Soit*

$$\mathbb{P}(T = k) = C_{m-1}^{k-1}(1-p)^{k-m}p^k,$$

qui vaut la probabilité d'obtenir $m-1$ piles au cours des $k-1$ premiers coups, multipliée par la probabilité d'obtenir pile au k–ième coup.

La logique voudrait que l'on choisisse comme valeur du paramètre m le plus petit nombre d'acides aminés que contient un gène, plus deux (pour les codons START + STOP). Malheureusement, ce nombre minimal peut être extrêmement réduit dans des cas tout à fait exceptionnels, alors qu'il est de l'ordre de la dizaine hormis ces cas tout à fait exceptionnels. Un choix raisonnable semble être $m = 10$, mais ce choix doit être critiqué–validé par le Biologiste (et/ou confronté à la séquence étudiée).

Quant au paramètre p, il doit être estimé.

Quant à la loi de probabilité qui régit les changements d'état, elle est définie comme suit. On admet que tout plage codante (à l'endroit comme à l'envers) est suivie d'une plage non codante. Donc $P_{10} = P_{20} = 1$. En outre, $P_{01} + P_{02} = 1$, et on peut soit supposer que $P_{01} = 1/2$, soit chercher à estimer cette quantité.

Discutons maintenant de la loi des nucléotides, sachant l'état caché. On peut admettre que dans les plages non codantes, les nucléotides sont i. i. d., la loi commune étant à estimer. Dans

une plage codante, on suppose par exemple que les codons sont mutuellement indépendants, le premier prenant ses valeurs dans l'ensemble des codons START possibles, le dernier dans l'ensemble des codons STOP possibles, les autres codons étant i. i. d., à valeurs dans les codons possibles qui codent pour un acide aminé (en particulier ces codons ne peuvent pas prendre comme valeur un des codons STOP). La description de la loi des codons d'une plage codante à l'envers se déduit aisément de celle que nous venons de donner pour les plages codantes à l'endroit.

12.4 Algorithme de Viterbi dans le cas semi–markovien caché

Nous allons maintenant décrire comment l'algorithme de Viterbi s'écrit dans notre nouvelle situation (qui est en fait une situation mixte markov caché – semi–markov caché).

Comme pour les chaînes de Markov cachées, l'idée est de calculer des quantités $\delta_y(n)$, pour tous les états cachés y, et $1 \leq n \leq N$. Pour $y = 0$, on pose comme précédemment

$$\delta_y(n) = \max_{y_1,y_2,\ldots,y_{n-1}} \mathbb{P}_\theta(Y_1 = y_1, \ldots, Y_{n-1} = y_{n-1}, Y_n = y, X_1 = x_1, \ldots, X_n = x_n).$$

Pour $y = 1$ (et de même pour $y = 2$), on pose

$$\delta_y(n) = \max_{y_1,\ldots,y_{n-1}} P_\theta(Y_1 = y_1,\ldots,Y_{n-1} = y_{n-1}, Y_n = y, Y_{n+1} \neq y, X_1 = x_1,\ldots,X_n = x_n)$$

La formule de récurrence est la suivante :

$$\delta_y(n) = \max \Big\{ P_\theta(X_1^n = x_1^n | Y_1^n = y, Y_{n+1} \neq y) d_y(n) \mu_y,$$
$$\max_{1 \leq k \leq n-1} \big[P_\theta(X_{n-k+1}^n = x_{n-k+1}^n | Y_{n-k} \neq y, Y_{n-k+1}^n = y, Y_{n+1} \neq y) \max_{z \neq y} [\delta_z(n-k) P_{zy}] d_y(k) \big] \Big\}$$

Nous n'allons pas détailler plus avant les calculs. Remarquons que le fait qu'à chaque étape on ait à calculer un max sur $1 \leq k \leq n$ rend l'algorithme a priori quadratique en N, ce qui est une mauvaise nouvelle. Cependant on peut limiter la portée de ce max, en arguant du fait qu'une région codante se termine au premier codon STOP rencontré. Cette remarque est encore vraie pour un exon dans le cas eukaryote : celui–ci se termine au plus loin lors de la rencontre du premier codon STOP (soit que ce codon marque effectivement la fin du gène, soit qu'il soit situé dans un intron). La même remarque ne s'applique pas aux régions non codantes, d'où le choix de prendre l'état 0 markovien.