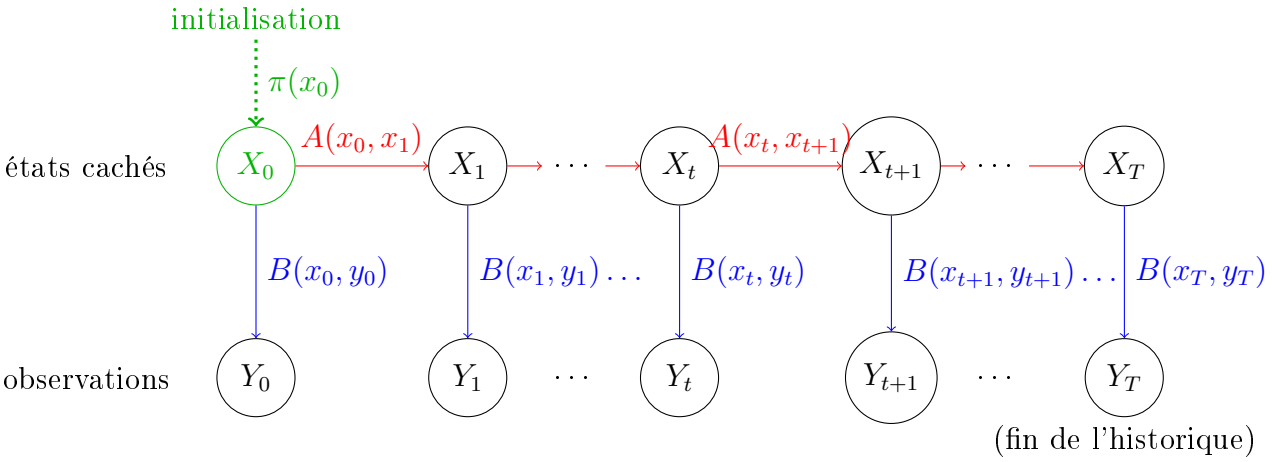
Retrouvons l'algorithme de Baum–Welch avec le formalisme EM

Jean-Baptiste Masson

avril 2019

1 HMM classique : notations

Schéma de génération des données :



Ensembles prédéfinis :

$\mathscr{X}$ = ensemble fini des états possibles (par exemple $\{1; 2; \ldots; n\}$).

$\mathscr{Y}$ = ensemble des observations possibles (fini ou $\mathbb{R}$ ou $\ldots$).

Le modèle λ est constitué de trois objets :

$\pi =$ loi de l'état initial :
$\pi : \mathscr{X} \to [0; 1]$ est une loi de probabilité sur $\mathscr{X}$.

$A =$ matrice des transitions :
$A : \mathscr{X} \times \mathscr{X} \to [0; 1]$ telle que $\forall x,\ [x' \mapsto A(x, x')]$ est une loi de probabilité sur $\mathscr{X}$.
Pour i et j deux états on note plus simplement $A(i, j) = a_{ij}$.

$B =$ description des émissions :
$B : \mathscr{X} \times \mathscr{Y} \to [0; 1]$ telle que $\forall x,\ [y \mapsto B(x, y)]$ est une loi de probabilité sur $\mathscr{Y}$.
Si $\mathscr{Y}$ est fini B est la matrice des b_{ik} avec $i \in \mathscr{X}$ et $k \in \mathscr{Y}$;
si $\mathscr{Y}$ est continu les $[y \mapsto B(x, y)]$ sont des densités $[y \mapsto f_\theta(y)]$,
etc.

Pour X et Y, on distingue les majuscules (variables aléatoires) des minuscules (valeurs fixées par l'observation ou par une hypothèse).

Notations simplifiées pour les intervalles de temps : quand $0 \leqslant t_1 \leqslant t_2 \leqslant T$,
$\mathbf{X}_{t_1:t_2}$ désigne le vecteur aléatoire $(\, X_{t_1} \,;\, X_{(t_1+1)} \,;\, \ldots \,;\, X_{t_2} \,)$, et $\mathbf{X} = \mathbf{X}_{0:T}$;
$\mathbf{y}_{t_1:t_2}$ désigne le vecteur fixe $(\, y_{t_1} \,;\, y_{(t_1+1)} \,;\, \ldots \,;\, y_{t_2} \,)$, et $\mathbf{y} = \mathbf{y}_{0:T}$;
etc.

2 Algorithme de Baum–Welch

L'algorithme classique de Baum–Welch part d'un modèle λ^0 fourni, et itère des « mises à jour » de λ tant que nécessaire.

Début itération :
Les valeurs actuelles du modèle sont dans 3 matrices : $\lambda = (\pi; A; B)$.

Étape forward :
On calcule les $\alpha(i,t)$ par récurrence, pour tous $i \in \mathscr{X}$ et $t = 0, \ldots, T$.
$\mathbf{t = 0}$ **:** $\forall i \in \mathscr{X},\ \alpha(i,0) = \pi(i) \cdot B(i, y_0)$.
$\mathbf{t \text{ -> } t{+}1}$ **:** $\forall i \in \mathscr{X},\ \alpha(i,t+1) = \left[\sum_{j \in \mathscr{X}} \alpha(j,t) \cdot a_{ji} \right] \cdot B(i, y_{t+1})$.

Étape backward :
On calcule les $\beta(i,t)$ par récurrence, pour tous $i \in \mathscr{X}$ et $t = T, \ldots, 0$.
$\mathbf{t = T}$ **:** $\forall i \in \mathscr{X},\ \beta(i,T) = 1$.
$\mathbf{t{+}1 \text{ -> } t}$ **:** $\forall i \in \mathscr{X},\ \beta(i,t) = \sum_{j \in \mathscr{X}} \underline{\beta(j,t+1) \cdot a_{ij} \cdot B(j, y_{t+1})}$.

Calculs intermédiaires :
On calcule les $\gamma(i,t)$ par $\tilde{\gamma}(i,t) = \alpha(i,t) \cdot \beta(i,t)$
qu'on normalise ensuite pour tout $t = 0, \ldots, T$: $\gamma(i,t) = \tilde{\gamma}(i,t) \Big/ \sum_{i' \in \mathscr{X}} \tilde{\gamma}(i',t)$;
ainsi que les $\xi(i,j,t)$ par $\tilde{\xi}(i,j,t) = \alpha(i,t) \cdot \underline{a_{ij} \cdot \beta(j,t+1) \cdot B(j, y_{t+1})}$
qu'on normalise pour tout $t = 0, \ldots, T{-}1$: $\xi(i,j,t) = \tilde{\xi}(i,j,t) \Big/ \sum_{i' \in \mathscr{X}} \sum_{j' \in \mathscr{X}} \tilde{\xi}(i',j',t)$.

Mise à jour du modèle λ :
$\forall i \in \mathscr{X},\quad \pi(i) \leftarrow \gamma(\, i \,;\, t = 0\,)$.
$\forall i \in \mathscr{X},\ \forall j \in \mathscr{X},\quad a_{ij} \leftarrow \sum_{t=0}^{T-1} \xi(i,j,t) \Big/ \sum_{t=0}^{T-1} \gamma(i,t)$.
$\forall i \in \mathscr{X},\ \forall k \in \mathscr{Y},\quad b_{ik} \leftarrow \sum_{t=0,\ldots,T \,/\, y_t = k} \gamma(i,t) \Big/ \sum_{t=0}^{T} \gamma(i,t)$.

Test d'arrêt :
Si la vraisemblance $\mathscr{L}(\lambda|\mathbf{y}) = \sum_{i \in \mathscr{X}} \alpha(i,T)$ a peu augmenté, arrêter et retourner λ.

Pour le reprogrammer en **R**, utiliser de grands tableaux `A`, `alpha`, etc. et effectuer des opérations sur les lignes et colonnes (`rowSums`, etc.) ou plus généralement « le long de certaines dimensions » avec la fonction `apply`.

Exemple :
```
A <- array( 1:24, dim=c(4,3,2) )
apply( X=A, MARGIN=c(1,2), FUN=sum )
```
calcule les sommes de `A` le long de sa 3^{e} dimension :
le résultat est une matrice 4×3 contenant des sommes de 2 valeurs.
```
apply( X=A, MARGIN=3, FUN=mean )
```
calcule les moyennes de `A` le long de ses 1^{e} et 2^{e} dimensions :
le résultat est un tableau de taille 2 contenant des moyennes de 12 valeurs.

Malheureusement les étapes *forward* et *backward* utilisent une récurrence : il n'est donc pas possible de les traiter autrement qu'avec une boucle, ce qui en langage interprété est lent…

Relations importantes :
$\alpha(i,t) = \mathbb{P}_\lambda(\, X_t = i \,\cap\, \mathbf{Y}_{0:t} = \mathbf{y}_{0:t} \,)$
$\beta(i,t) = \mathbb{P}_\lambda(\, \mathbf{Y}_{(t+1):T} = \mathbf{y}_{(t+1):T} \mid X_t = i \,)$
$\tilde{\gamma}(i,t) = \mathbb{P}_\lambda(\, X_t = i \,\cap\, \mathbf{Y} = \mathbf{y} \,) = \alpha(i,t) \cdot \beta(i,t)$
$\gamma(i,t) = \mathbb{P}_\lambda(\, X_t = i \mid \mathbf{Y} = \mathbf{y} \,)$ avec $\sum_i \gamma(i,t) = 1$
$\tilde{\xi}(i,j,t) = \mathbb{P}_\lambda(\, X_t = i \,\cap X_{t+1} = j \,\cap\, \mathbf{Y} = \mathbf{y} \,)$ a un ingrédient commun avec β (souligné)
$\xi(i,j,t) = \mathbb{P}_\lambda(\, X_t = i \,\cap X_{t+1} = j \mid \mathbf{Y} = \mathbf{y} \,)$ avec $\sum_i \sum_j \xi(i,j,t) = 1$
$\sum_j \tilde{\xi}(i,j,t) = \tilde{\gamma}(i,t)$ et $\sum_j \xi(i,j,t) = \gamma(i,t)$: en fait les γ peuvent se déduire des ξ
$\forall t,\ \sum_i \tilde{\gamma}(i,t) = \mathscr{L}(\lambda|\mathbf{y}) = \sum_i \alpha(i,T)$

3 Formalisme EM en général

La vraisemblance : $\mathscr{L}(\,\lambda \,|\, \mathbf{y}\,) = \mathbb{P}_\lambda(\mathbf{Y} = \mathbf{y})$
a généralement une expression difficile à optimiser (produit de sommes) car il faut tenir compte de toutes les possibilités (ici trajectoires) pour tous les X_i.

La vraisemblance « complétée » :
$\mathscr{L}_c(\,\lambda \,|\, \mathbf{x}\,;\, \mathbf{y}\,) = \mathbb{P}_\lambda(\,\mathbf{X} = \mathbf{x} \cap \mathbf{Y} = \mathbf{y}\,)$
a une forme plus simple (produit) car elle s'exprime avec des x_i particuliers.

Dempster, Laird et Rubin (1977) définissent alors une fonction auxiliaire :
$$Q(\lambda^+|\lambda^-) = \mathbb{E}\left[\, \ln \mathscr{L}_c\left(\lambda^+ \middle|\, \underbrace{\mathbf{X}^-}_{\text{loi issue de } \lambda^-}\ ;\ \mathbf{Y}\right) \middle|\, \underbrace{\mathbf{Y} = \mathbf{y}}_{\text{obs. fixes}}\right].$$

Leur (méta-)algorithme part d'un modèle λ^0 fourni, et itère des étapes E et M (tant que nécessaire. Il garantit que la vraisemblance augmente à chaque itération.

L'étape E *(expectation)* consiste à calculer les coefficients permettant d'expliciter l'espérance Q comme une fonction de λ^+, quand λ^- est fourni par l'itération précédente.

L'étape M *(maximization)* consiste à trouver λ^+ qui maximise $Q(\cdot|\lambda^-)$. Généralement cela revient aux formules de maximum de vraisemblance « habituelles », avec des « pondérations » par les coefficients de l'étape E. On met alors à jour $\lambda^- \leftarrow \lambda^+$, et on passe à l'itération suivante.

4 Application du formalisme EM aux HMM

4.1 Fonction Q

Pour les HMM avec $\lambda = (\pi, A, B)$, d'après le schéma page 1 la vraisemblance complétée s'écrit :

$$\mathscr{L}_c(\,\lambda\,|\,\mathbf{x}\,;\,\mathbf{y}\,) \;=\; \pi(x_0)\cdot B(x_0,y_0)\cdot\prod_{t=1}^{T} A(x_{t-1},x_t)\cdot B(x_t,y_t).$$

On doit donc étudier l'espérance, quand les X_i suivent la loi décrite par λ^-, et conditionnellement à la connaissance de tous les y_t, de la quantité suivante :

$$\ln\mathscr{L}_c(\,\lambda^+\,|\,\mathbf{X}\,;\,\mathbf{Y}\,) \;=\; \underbrace{\ln\pi^+(X_0)}_{F} + \underbrace{\ln B^+(X_0,Y_0)}_{G} + \left[\sum_{t=1}^{T}\underbrace{\ln A^+(X_{t-1},X_t)}_{H_t} + \underbrace{\ln B^+(X_t,Y_t)}_{J_t}\right].$$

Espérance $[\dots]$ de F :

$$\mathbb{E}_{(\lambda^-)}[F|\mathbf{y}] \;=\; \sum_{i\in\mathscr{X}} \underbrace{\mathbb{P}_{(\lambda^-)}(X_0=i|\mathbf{Y}=\mathbf{y})}_{\gamma^-(i,0)\ issu\ de\ \lambda^-}\cdot\ln\underbrace{\pi^+(i)}_{variable}.$$

Espérance $[\dots]$ de G :

$$\mathbb{E}_{(\lambda^-)}[G|\mathbf{y}] \;=\; \sum_{i\in\mathscr{X}} \underbrace{\mathbb{P}_{(\lambda^-)}(X_0=i|\mathbf{Y}=\mathbf{y})}_{\gamma^-(i,0)\ issu\ de\ \lambda^-}\cdot\ln\underbrace{B^+(i,y_0)}_{variable}$$

Espérance $[\dots]$ de H_t :

$$\mathbb{E}_{(\lambda^-)}[H_t|\mathbf{y}] \;=\; \sum_{i\in\mathscr{X}}\sum_{j\in\mathscr{X}} \underbrace{\mathbb{P}_{(\lambda^-)}(X_{t-1}=i\cap X_t=j|\mathbf{Y}=\mathbf{y})}_{\xi^-(i,j,t-1)\ issu\ de\ \lambda^-}\cdot\ln\underbrace{A^+(i,j)}_{variable}.$$

Espérance $[\dots]$ de J_t :

$$\mathbb{E}_{(\lambda^-)}[J_t|\mathbf{y}] \;=\; \sum_{i\in\mathscr{X}} \underbrace{\mathbb{P}_{(\lambda^-)}(X_t=i|\mathbf{Y}=\mathbf{y})}_{\gamma^-(i,t)\ issu\ de\ \lambda^-}\cdot\ln\underbrace{B^+(i,y_t)}_{variable}.$$

On voit apparaître « naturellement » les quantités γ et ξ définies dans l'algorithme de Baum–Welch. Sans surprise, la formule pour G est celle de J_t étendue à l'indice $(t=0)$.

Finalement, dans la somme sur t on regroupe les y_t égaux entre eux sous forme de $\sum_{k\in\mathscr{Y}}\sum_{t/y_t=k}$ et on obtient une expression de Q où les variables sont bien séparées :

$$\begin{aligned} Q(\lambda^+|\lambda^-) \;=\;& \sum_{i\in\mathscr{X}}\gamma^-(i,0)\cdot\ln\pi^+(i) + \sum_{i\in\mathscr{X}}\sum_{j\in\mathscr{X}}\left[\sum_{t=0}^{T-1}\xi^-(i,j,t)\right]\cdot\ln A^+(i,j) \\ &+ \sum_{i\in\mathscr{X}}\sum_{k\in\mathscr{Y}}\left[\sum_{t=0,\dots,T\,/\,y_t=k}\gamma^-(i,t)\right]\cdot\ln B^+(i,k). \end{aligned}$$

4.2 Étape E

L'étape E consiste ici à calculer les γ et les ξ issus du modèle précédent λ^-. Elle correspond donc exactement aux étapes *forward*, *backward*, et *calculs intermédiaires* de Baum–Welch.

4.3 Étape M

L'étape M consiste ici à calculer les tableaux π^+, A^+ et B^+ qui maximisent Q sous plusieurs contraintes (les sommes de certains coefficients doivent être égales à 1).

On peut utiliser la propriété suivante *(qui se prouve avec des multiplicateurs de Lagrange)* :

> Le maximum de $f : (x_1, \dots, x_n) \mapsto \sum a_i \cdot \ln x_i$, sous contrainte $\sum x_i = 1$,
> avec $(a_1, \dots, a_n)$ constantes positives, est atteint lorsque $\forall i,\ x_i = a_i / \sum_k a_k$.

séparément avec chaque composante de λ^+ ayant une contrainte de somme égale à 1 :

- le tableau $\pi^+(\dots)$ indexé par i ;
- pour chaque i, le tableau $A^+(i, \dots)$ indexé par j ;
- pour chaque i, le tableau $B^+(i, \dots)$ indexé par k.

On arrive à montrer ainsi (et grâce à certaines relations de la page 3) que l'étape M correspond exactement à l'étape de *mise à jour* dans l'algorithme de Baum–Welch.

5 Variante avec des observations continues

5.1 Émissions gaussiennes

Lorsque l'espace $\mathscr{Y}$ est de nature continue, par exemple $\mathbb{R}$, il faut remplacer la matrice B par une famille de fonctions de densité…

Supposons ici que tous ces densités sont gaussiennes : $B(i, y) = \dfrac{1}{\sigma_i \sqrt{2\pi}} \cdot exp\left(\dfrac{-(y - \mu_i)^2}{2\,\sigma_i^2} \right)$;
alors la matrice des paramètres b_{ik} est remplacée par

- un tableau de moyennes $\mathbf{M} = [\mu_i]_{i \in \mathscr{X}}$;
- un tableau d'écarts-types $\mathbf{S} = [\sigma_i]_{i \in \mathscr{X}}$ ou de variances $\mathbf{V} = [v_i]_{i \in \mathscr{X}}$ avec $v_i = \sigma_i^2$.

On insère le $B(i, y)$ ci-dessus tel quel dans les formules qui l'utilisent (pour les $\alpha, \beta, \gamma, \xi$).

D'autre part dans l'expression de la fonction Q on insère

$$\ln B(i, y) \;=\; \underbrace{\frac{-1}{2} \ln(2\pi)}_{\textit{constante}} \;-\; \ln(\sigma_i) \;-\; \frac{1}{2} \left(\frac{y - \mu_i}{\sigma_i} \right)^2 \;=\; \frac{-1}{2} \ln(2\pi) \;-\; \frac{1}{2} \ln(v_i) \;-\; \frac{(y - \mu_i)^2}{2\,v_i}.$$

La mise à jour de la matrice $[b_{ik}]$ doit être remplacée par la mise à jour de $\mathbf{M}$ et $\mathbf{V}$ (ou $\mathbf{S}$).
On aboutit classiquement à des formules de moyennes et variances pondérées par les $\gamma^-(i, t)$:

nouveaux « effectifs flous » : $n_i^+ = \displaystyle\sum_{t=0}^{T} \gamma^-(i, t)$;

(Attention : tester si n_i^+ n'est pas trop proche de 0,
sinon état i trop peu souvent visité et modèle à rejeter ou adapter.)

nouvelles moyennes : $\mu_i^+ = \dfrac{1}{n_i^+} \displaystyle\sum_{t=0}^{T} \left[\gamma^-(i, t) \cdot y_t \right]$;

nouvelles variances : $v_i^+ = \dfrac{1}{n_i^+} \displaystyle\sum_{t=0}^{T} \left[\gamma^-(i, t) \cdot (y_t - \mu_i^+)^2 \right] = \dfrac{1}{n_i^+} \displaystyle\sum_{t=0}^{T} \left[\gamma^-(i, t) \cdot y_t^2 \right] - (\mu_i^+)^2$.

5.2 Observations continues et multivariées

Pour plusieurs variables observées simultanément : $\vec{y} = (y^{(1)}; y^{(2)}; \dots) = (\text{bruit}, \text{CO}_2, \dots)$,
on peut définir une gaussienne par état et par variable : $B(i, y^{(\ell)}) =$ densité de $\mathcal{N}(\mu_{i\ell}; \sigma_{i\ell})$
telles que pour chaque état i, $B(i, \vec{y}) = \prod_{\ell} B(i, y^{(\ell)})$.

Les tableaux $\mathbf{M}$ et $\mathbf{V}$ deviennent des matrices indexées par (i, ℓ).

Cela revient à supposer les variables *indépendantes conditionnellement* à la connaissance des états. Les formules de mise à jour sont alors les mêmes que ci-dessus, séparément variable par variable (les n_i^+ ne dépendent que de i et pas de ℓ : ne les calculer qu'une fois).

6 Astuce numérique : échelle logarithmique

On est parfois amené à calculer des produits de nombreuses probabilités : il y a un risque d'*underflow*. Une astuce consiste à stocker non pas les valeurs du modèle λ mais leurs logarithmes : les produits de probabilités sont remplacés par une somme de valeurs stockées.

Par contre dans ce cadre, effectuer des sommes de probabilités n'est plus une opération « basique » : on peut avoir recours à l'algorithme suivant.

entrées : $\ln P_1,\ \ln P_2,\ \dots,\ \ln P_n$ *// logarithmes de probabilités : dans* $[-\infty\,;\,0]$

sortie : $\ln S$ *// telle que* $S = \sum P_i$

étapes de calcul :

$D \leftarrow \dots$ *// choix d'un décalage**

$\forall i,\ Q_i \leftarrow (\ln P_i) + D$ *// cela vaut aussi* $\ln(P_i \times e^D)$

$\forall i,\ R_i \leftarrow \exp(Q_i)$ *// calcul stabilisé de* $P_i \times e^D$

$T \leftarrow \sum R_i$ *// calcul stabilisé de* $\sum P_i \times e^D = S \times e^D$

$\ln S \leftarrow (\ln(T)) - D$ *// calcul stabilisé de* $\ln(S \times e^D) - D = \ln S + D - D = \ln S$

* Le décalage peut être quelconque mais il « stabilisera » les calculs uniquement si les R_i sont dans une gamme de flottants éloignée des *underflows*/*overflows* susceptibles de se produire à cause de la fonction exponentielle.

Pour le format double précision, les logarithmes « sans problèmes » sont situés entre -708 et $+709$ environ : il faut choisir D pour que les Q_i et $\ln(T)$ soient dans cette gamme si possible (sauf éventuellement les plus petits $Q_i \dots$).

Une stratégie possible : $D = -\max\{\ln P_i \ /\ i = 1, \dots, n\}$;
ici on garantit $\max Q_i = 0$ donc $\max R_i = 1$ et enfin $T \leqslant n$.

(Attention : si $D = +\infty$ *renvoyer* $\ln S = -\infty$*, car tous les* P_i *sont nuls mais les* Q_i *sont NaN.)*

On peut s'inspirer de cet algorithme pour définir une fonction de « normalisation » :
étant donné $(\ln c_1, \dots, \ln c_n)$, calculer $(\ln c'_1, \dots, \ln c'_n)$ tels que $[c']$ proportionnel à $[c]$ et $\sum c'_i = 1$
(utile notamment pour le calcul des γ *et des* ξ*).*

Le document de Mark Stamp expose une stratégie similaire, plus spécifique aux HMM (section « *HMM scaling* »).