Online EM for functional data

Alexandre Philbert, Raphaël Romero

ENS Paris-Saclay

09/01/2018

Introduction

- Online Em
- Comparaison de online Em et Em classique sur le cas des mixtures de poisson
- Présentation du modéle hierarchique
- Présentation de l'algorithme
- Application numérique : Implémentation pour le dataset de growth velocity curves

Online EM

- Algorithme itératif dérivé de l'EM
- Mise à jour séquentielle des paramètres
- Utilise une approximation stochastique à l'étape E
- Modèles exponentiels:

$$logL_\theta(X, Y) = \langle S(X, Y), \psi(\theta) \rangle - A(\theta)$$

Online EM

- EM Classique :
- E:

$$S_n^{k+1} = \frac{1}{n}\sum_{t=1}^{n} \mathbb{E}[s(Y_t, Z_t) | Y_t, \theta_n^k]$$

- M:

$$\theta_n^{k+1} = \bar{\theta}(S_n^{k+1})$$

où $\bar{\theta} : S \mapsto argmax_\theta(\langle S, \psi(\theta)\rangle - A(\theta))$

Online EM

En utilisant ces notations, on remarque que par le théorème central limite, lorsque le nombre d'observations devients grand les étapes deviennent :

- Quand $n \to \infty$:
- E:
$$S_n^{k+1} = \mathbb{E}_{\pi_Y} \mathbb{E}[s(Y_1, Z_1) | Y_1, \theta_n^k]$$
- M :
$$\theta_n^{k+1} = \bar{\theta}(S_n^k)$$

Online EM

Dans ce contexte l'online EM vise à trouver les solutions de l'équation

$$S = \mathbb{E}_{\pi_Y}\mathbb{E}[s(Y_1, Z_1)|Y_1, \bar{\theta}(S)]$$

D'où l'idée d'utiliser une approximation stochastique (Robbins-Monro)

Online EM

- Etape SA-E : Soit (γ_n) une suite décroissante dont la série diverge et dont la série des carrés converge.

$$S_{n+1} = (1 - \gamma_n)S_n + \gamma_n \mathbb{E}[s(X_n, Y_n)|Y_n]$$

- Etape M: $\theta_{n+1} = argmax_\theta \langle S_{n+1}, \psi(\theta) \rangle - A(\theta)$

Online EM en pratique

- Il peut arriver que l'on n'ait pas accès à

$$\mathbb{E}[s(X_n, Y_n)|Y_n]$$

- Solution: Calcul approché par MCMC

$$\mathbb{E}[s(X_n, Y_n)|Y_n] \approx \frac{1}{m_n} \sum_{k=1}^{m_n} s(X_n[k], Yn)$$

Mixture de poisson

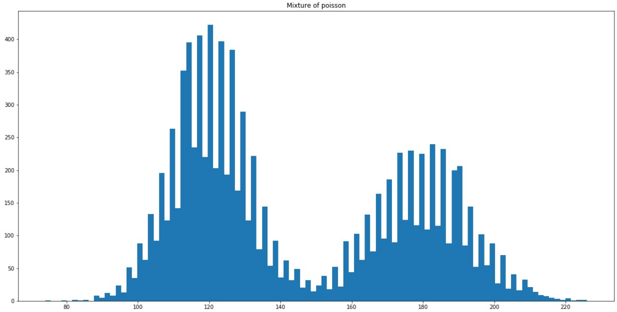
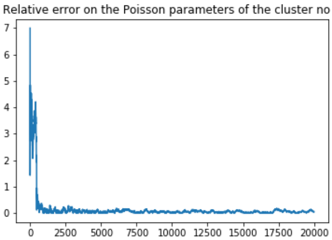


Figure 1: Histogramme de 1000 variables générées selon deux clusters

Mixture de poisson



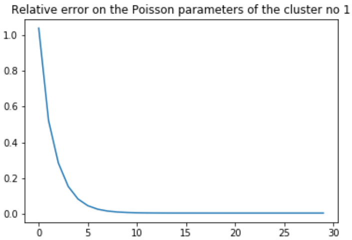


Figure 2: Erreur relative sur le paramètre de poisson associé au premier cluster (Online à gauche, batch à droite)

Mixture de poisson

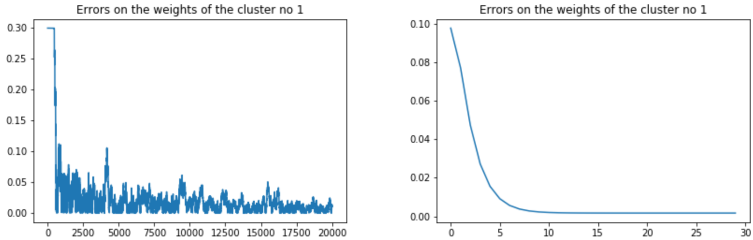


Figure 3: Erreur relative sur le poids associé au premier cluster (Online à gauche, batch à droite)

Mixture de poisson

- A chaque itération on n'a besoin que d'une observation
- L'Online EM assez robuste combiné avec une moyenne de Polyak-Ruppert
- Il n'est pas nécessaire de parcourir toutes les données à chaque itérations pour obtenir de bons résultats

Modèle déformable

- Espace des observation (en théorie) : espace de fonctions $f : \mathbb{U} \to \mathbb{R}$
- Les "observations" sont $Y(u) = \lambda f(D(u, \beta)) + \sigma W(u)$
- Ces fonctions sont décomposées sur un dictionnaire $\{\phi_l\}_{l=1,\dots,m}$:

$$f = \sum_{l=1}^{m} \alpha_l \phi_l$$

Modèle déformable

- En réalité on n'observe que $Y(u_i)$, $u_i \in \Omega = \{u_1, ..., u_{|\Omega|}\}$
- Forme matricielle du modèle:

$$Y = \lambda \Phi_\beta \alpha + \sigma W$$

où
$(\Phi_\beta)_{ij} = \phi_j(u_i), \alpha = (\alpha_1, ..., \alpha_m)^T, W = W(u_1), ..., W(u_\Omega)$

Mixture de modèles déformables

- Modèle hiérarchique utile pour le clustering de données fonctionnelles
- Plusieurs classes $(\mathcal{C}_j)_{j=1,\ldots,C}$ auxquelles sont associées les composantes $(\alpha_j)_{j=1,\ldots,C}$

$$Y \in \mathcal{C}_j \Rightarrow Y = \lambda \Phi_\beta \alpha_j + \sigma W$$

$$[\Phi_\beta]_{s,l} = \phi_l \circ D(u_s, \beta_j)$$

ou les $(u_s)_s$ sont les points d'observations et $(\phi_l)_l$ une base de fonctions, typiquement des noyaux gaussiens. Typiquement β est une variables aléatoire dont la distribution dépend de la classe j.

Formulation du problème statistique

- Observations : Y

$$Y|I = j, \lambda, \beta \sim \mathcal{N}(\lambda \Phi_\beta \alpha_j, \sigma^2 I_d)$$

- Paramètres à estimer: $\theta = ((\Gamma_j, \alpha_j, \omega_j)_j, \sigma)$
- Variables latentes : (X, I) avec $X = (\lambda, \beta)$

$$I \sim \mathcal{M}(1, \omega)$$

$$\lambda \sim \mathcal{G}amma(a, 1/a)$$

$$\beta|I = j \sim \mathcal{N}(0, \Gamma_j)$$

- Hyperparamètres : a

Vraisemblance

- $L_\theta(I, Y, X) = g_\theta(Y|I, X) p_\theta(X|I) w_I$

$$g_\theta(Y|I, X) \propto exp(-\frac{1}{2\sigma^2}||Y - \lambda\phi_\beta\alpha_I||^2)$$

$$p_\theta(X|I) \propto exp(-\frac{1}{2}\beta^T\Gamma_I^{-1}\beta)\lambda^{a-1}exp(-a\lambda)$$

$$p_\theta(I = j) = \omega_j$$

Forme exponentielle

- $log L_\theta(I, X, Y) = t(\theta) + \langle r(\theta), S(I, X, Y)\rangle$
- log-partition:

$$t(\theta) = log\frac{a^a}{\mathcal{G}(a)} - \frac{|\Omega|}{2} log 2\pi\sigma^2 - d_\beta log 2\pi$$

,

- Statistique exhaustive: $S(I, X, Y) = (S_j(I, X, Y))_{j=1,\dots,C}$ où

$$S_j(I, X, Y) = \delta_{I,j}(1, \lambda\phi_\beta^T Y, \lambda^2\phi_\beta^T\phi_\beta, \beta\beta^T, ||Y||^2, \lambda, log\lambda)$$

- Paramètre canonique :

$$r(\theta) = (r_1(\theta), \dots, r_C(\theta))$$

$$r_j(\theta) = \frac{1}{2}(2log(w_j) - logdet(\Gamma_j), \frac{2\alpha_j}{\sigma^2}, \frac{-\alpha_j\alpha_j^T}{\sigma^2}, (-\Gamma_j^{-1})^T,$$

$$-\frac{1}{\sigma^2}, -2a, -2(a-1))$$

Algorithme (1)

- A chaque fois qu'une nouvelle observation Y_n est disponible, on fait une nouvelle étape d'approximation stochastique:

$$s_{n,j} = s_{n-1,j} + \rho_n(\mathbb{E}_{\theta_{n-1}}[S_j(I, X, Y_n)|Y_n] - s_{n-1,j})$$

- M step :

$$\theta_n = argmax_\theta\{t(\theta) + \sum_j < r_j(\theta), s_{n,j} >\}$$

- Dans la pratique : on ne met pas à jour θ_n pour chaque nouvelle observation

Algorithme (2)

- Problème : Pas de closed form pour esperance selon $\pi_{\theta_n}(\cdot|Y)$
- Solution : $\mathbb{E}_{\theta_n}[S_j(X, I, Y)|Y_n] \approx \frac{1}{m_n}\sum_{k=1}^{m_n} S_j(I[k], X[k], Yn)$
- Problème : On ne sait pas simuler selon $\pi_{\theta_n}(\cdot|Y)$
- Solution : Utiliser une méthode de simulation MCMC type Gibbs
- Problème : $\pi_{\theta_n}(\cdot|Y)$ n'est pas définie sur un espace produit
- Solution : On introduit des variables auxiliaires

Algorithme (3)

- On simule avec la méthode de Gibbs

$$\tilde{\pi_\theta}(I, \tilde{X}_1, ..., \tilde{X}_C) = \pi_\theta(I, X_I | Y) \prod_{j \neq I} \kappa_{\theta,j}(\tilde{X}_j)$$

- Pour $j = I$, $\tilde{\pi_\theta}(\tilde{X}_j | I, \tilde{X_{-j}}, Y) = \pi_\theta(\tilde{X}_I | I, Y)$ nécessite un Random Walk Metropolis-Hastings
- On ne retourne que les (X_I, I) simulés

Algorithme (4)

En conclusion lorsque une Y_n arrive

- On simule un échantillion $\{I_n[k], X_n[k]\}_k$
- On met à jour la statistique suffisante dans l'étape SA en remplaçant l'esperance par la moyenne des echantillions
- On met à jour θ lors de la M step (pas forcément à chaque fois)

Algorithme (5)

Pour la M step on a une formule close

- $$\hat{w}_j = \frac{\tilde{s}_{n,j}^{(1)}}{\sum_{j'=1}^{C} \tilde{s}_{n,j'}^{(1)}}$$

  $$\hat{\alpha}_j = (\tilde{s}_{n,j}^{(3)}))^{-1} \tilde{s}_{n,j}^{(2)}$$

  $$\hat{\sigma}^2 = \frac{1}{|\Omega|} \sum_{j=1}^{C} (-2\hat{\alpha}_j^T \tilde{s}_{n,j}^{(2)} + tr(\hat{\alpha}_j \hat{\alpha}_j^T \tilde{s}_{n,j}^{(3)}) + \tilde{s}_{n,j}^{(5)})$$

  $$\hat{\Gamma}_j = \frac{\tilde{s}_{n,j}^{(4)}}{\tilde{s}_{n,j}^{(1)}}$$

Implementation : Growth velocity curves

- $(u_s)_s$ irréguilièrement éspacé dans (2,18)
- $(\phi_l)_l$ noyaux guassiens régulièrement espacés dans (2,18)
- $D(u,\beta) = u_i + (u_i - u_f) * H(u,\beta)$
- $H(u,\beta) = \frac{\int_{u'_i}^{u} exp(\sum_{k=1}^{d_\beta} \beta_k \Psi_k(\nu))}{\int_{u'_i}^{u'_f} exp(\sum_{k=1}^{d_\beta} \beta_k \Psi_k(\nu))}$

Données utilisées

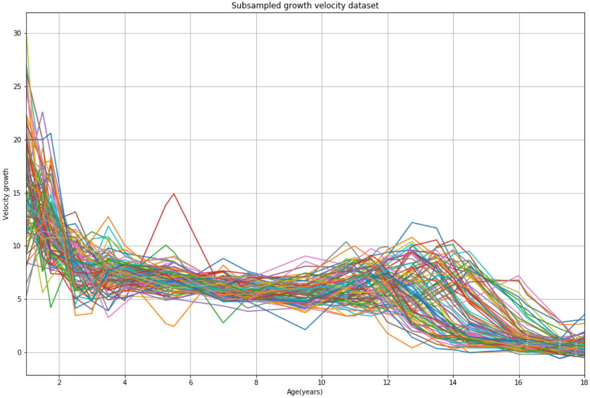


Figure 4: Données: Berkeley Growth Velocity Curves

Implementation

- Pour déterminer la densité des pseudo priors il faut trouver

$$argmax_X \pi_\theta(X|j, Y)$$

- coûteux en temps
- $D(., \beta) = C_0 + C_1 D^{-1}\{exp(D^{-1}w_\beta)\}$
- dans l'article w_β est décomposé dans une base de d_β P-splines
- Nous l'avons décomposé dans une base de d_β B-spline d'ordre 1
- le temps de calcul est divisé par 100

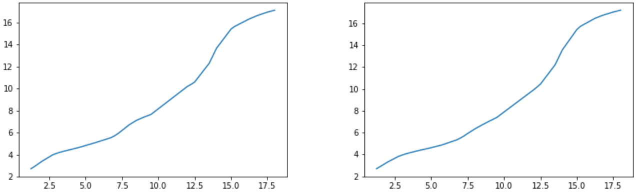


Figure 5: $D(u, \beta)$ pour β aléatoire, gauche : le modèle de l'article, droite : notre modèle

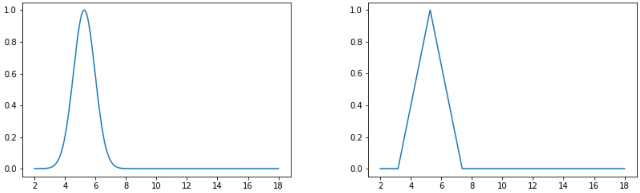


Figure 6: $D(u, \beta)$ pour β aléatoire, gauche : le modèle de l'article, droite : notre modèle

Implementation

- Nous n'avons pas déterminé une variance qui nous donne un taux d'acceptation moyen raisonable pour le RWMH : adaptive RW ?
- Nous n'avons pas réussit à reproduire les résultats de l'article.

- Florian Maire, Eric Moulines, and Sidonie Lefebvre. 2017. Online EM for functional data. Comput. Stat. Data Anal. 111, C (July 2017), 27-47. DOI: https://doi.org/10.1016/j.csda.2017.01.006
- Online EM Algorithm for Latent Data Models Olivier Cappé (LTCI), Eric Moulines (LTCI)
- Ramsay, J. O. and Li, X. (1998), Curve registration. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 60: 351–363. doi:10.1111/1467-9868.00129