Projet modèles à espace d'état et méthodes de Monte Carlo séquentielles
Algorithme PMCMC pour estimer l'évolution de populations animales

Samuel Ritchie, Sholom Schechtman, Nicolas Schreuder

FIGURE 1 – Un myocastor coypus (plus communément dénommé "ragondin")

1 Introduction

Pour le projet relatif au cours "Modèles à espaces d'état de méthodes de Monte Carlo séquentielles" nous avons travaillé sur l'article [PHH10].

Cet article s'intéresse à l'utilisation d'un algorithme de type PMCMC (Particle Markov Chain Monte Carlo), développé dans [ADH10], pour obtenir des échantillons tirés selon les lois à posteriori de modèles à espace d'état non linéaires.

Nous débutons ce rapport par une brève description des modèles de l'article avant de présenter les résultats numériques que nous avons obtenus sur des données synthétiques et réelles.

L'intégralité des codes écrits pour ce projet sont disponibles sur le dépôt Github accessible via le lien suivant `https://github.com/NicolasSchreuder/AdPMCMC_project`.

2 Cadre et modèles

Dans cette partie nous décrivons le cadre considéré et nous présentons succinctement les modèles d'évolution de population considérés dans l'article.

Les auteurs considèrent un modèle à espace d'état classique, il est constitué d'un processus d'observations y_t et d'un processus de Markov latent n_t. La relation entre ces deux processus est caractérisée par l'équation d'observation [1] :

$$y_t = n_t + w_t, w_t \sim \mathcal{N}(0, \sigma_w^2) \tag{1}$$

Afin de définir pleinement l'évolution de la population animale, il reste à choisir le modèle régissant le processus latent n_t. Cinq modèles sont évoqués dans l'article. Nous les décrivons ici, du plus simple au plus sophistiqué.

1. L'article évoque une équation plus générale $y_t = g(n_t) + w_t$, nous supposons ici que g est la fonction identité, ce que l'article suggère implicitement

2.1 Modèle M_0 (Exponentiel)

Le modèle M_0 est le modèle le plus simple considéré dans l'article, c'est le modèle de croissance exponentielle, décrit par l'équation suivante :

$$\log N_{t+1} = \log N_t + b_0 + \epsilon_t \tag{2}$$

Le paramètre b_0 représente le taux de croissance individuel maximum, c'est la différence entre le taux de naissance et de mortalité. Dans ce type de modèle (linéaire en les paramètres et en le processus latent), le taux de croissance ne dépend pas de la taille de la population, il est constant égal à b_0.

2.2 Modèle M_1 (Ricker)

Afin de rendre dépendant le taux de croissance à la taille de la population, on introduit un terme exponentiel en N_t pour obtenir une équation theta-logistique :

$$\log N_{t+1} = \log N_t + b_0 + b_1 N_t + \epsilon_t \tag{3}$$

En réécrivant cette équation sous la forme $N_{t+1} = N_t e^{b_0(1-\frac{N_t}{K})+\epsilon_t}$, on met ainsi en évidence une capacité maximale (dite *'carrying capacity'*) $K = -\frac{b_0}{b_1}$ qui existe lorsque $b_0 < 0$ (pour la plupart des populations la dépendance en la densité b_1 est négative).

2.3 Modèle M_2 (Theta-logistic)

Le modèle de Ricker suppose une dépendance linéaire du taux de croissance en la densité, ce qui n'est pas forcément le cas dans la réalité. Le modèle theta-logistique, plus général, corrige cela en rajoutant un paramètre b_3 de dépendance en la densité :

$$\log N_{t+1} = \log N_t + b_0 + b_2 (N_t)^{b_3} + \epsilon_t \tag{4}$$

de telle manière que $N_{t+1} = N_t e^{b_0(1-(\frac{N_t}{K})^{b_3})+\epsilon_t}$, où la *'carrying capacity'* est désormais définie par $K = (-\frac{b_0}{b_2})^{\frac{1}{b_3}}$, b_0 et b_2 devant être de signe opposé pour qu'elle existe.

2.4 Modèles M_3 et M_4 (Allee effect)

Pour certaines populations (et notamment pour de petits effectifs), il a été observé un phénomène de dépendance *positive* entre densité d'une population et taux de croissance [2] : ce phénomène est appelé *l'effet Allee*. Il en existe deux types : fort lorsqu'en dessous d'une densité de population critique, le taux de croissance par individu est négatif ; faible si le taux de croissance est toujours positif mais plus faible pour des faibles densités.
Les modèles M_3 et M_4 décrits dans l'article permettent de rendre compte de cet effet, respectivement dans sa version forte et faible.

3 Algorithme PMCMC

La distribution d'intérêt est la loi à posteriori de la trajectoire sous-jacente et des paramètres connaissant les observations. Les paramètres, que l'on rassemble dans ce qui suit dans le vecteur $\boldsymbol{\theta}$, correspondent aux paramètres b_i des équations régissant le processus latent n_t, de l'état initial n_0 et des variances des erreurs de l'équation du processus latent et de l'équation d'observation σ_ϵ et σ_w.

2. Ce qui est donc contraire aux modèles décrits jusqu'ici

Le théorème de Bayes nous permet alors d'écrire la densité de cette distribution de la façon suivante :

$$p(n_{1:t}, \boldsymbol{\theta}|y_{1:t}) \propto p(\boldsymbol{\theta}) \prod_{t=1}^{T} p(y_t|n_t, \boldsymbol{\theta}) p(n_t|n_{t-1}, \boldsymbol{\theta}) \tag{5}$$

L'inférence bayésienne, à travers les méthodes de Monte Carlo séquentielles, est une approche particulièrement intéressante pour approximer les modèles précédemment décrits.
L'algorithme PMCMC ([ADH10]) a été développé pour ce type de modèle où il faut estimer à la fois un processus de Markov inconnu et les paramètres du modèle.

Nous décrivons une itération de l'algorithme :

— Des paramètres candidats $\boldsymbol{\theta}^*$ sont générés à partir d'une loi à priori pour l'initialisation, à partir d'un noyau de transition ensuite.
— Grâce à un filtre particulaire, des trajectoires vraisemblables $n^*_{1:T}$ par rapport aux observations et aux paramètres candidats sont générées. Une trajectoire $n^*_{1:T}$ est sélectionnée de façon aléatoire parmi ces trajectoires, en fonction du poids qui lui est associé.
— Le couple $[\boldsymbol{\theta}^*, n^*_{1:T}]$ est ensuite accepté avec un certaine probabilité α qui dépend exclusivement de la vraisemblance du modèle (calculée dans le filtre), des lois à priori et des noyaux.

L'algorithme PMMH revient donc à intégrer un filtre particulaire à un algorithme Metropolis-Hastings. Le détail de l'algorithme se trouve dans [PHH10] et [ADH10].

La probabilité d'accepter le candidat $[\boldsymbol{\theta}^*, n^*_{1:T}]$ par rapport au candidat précédemment accepté $[\boldsymbol{\theta}, n_{1:T}]$ s'exprime facilement :

$$\alpha([\boldsymbol{\theta}, n_{1:T}], [\boldsymbol{\theta}^*, n^*_{1:T}]) = \min\left(1, \frac{p(y_{1:T}|\boldsymbol{\theta}^*)p(\boldsymbol{\theta}^*)q(\boldsymbol{\theta}^*, \boldsymbol{\theta})}{p(y_{1:T}|\boldsymbol{\theta})p(\boldsymbol{\theta})q(\boldsymbol{\theta}, \boldsymbol{\theta}^*)}\right) \tag{6}$$

Par ailleurs, les auteurs de l'article proposent une amélioration de l'algorithme PMCMC classique, qu'ils appellent *Adaptive PMCMC*, en ayant recours à noyau adaptatif pour générer les nouveaux paramètres candidats. Au lieu de tirer les nouveaux paramètres candidats selon une gaussienne centrée en le dernier paramètre et à matrice de covariance fixée, on met à jour la matrice de covariance à l'aide de l'ensemble des paramètres tirés lors des itérations précédentes de l'algorithme, améliorant ainsi la probabilité d'accepter à l'étape suivante.

4 Résultats numériques

Nous présentons ici quelques résultats numériques obtenus sur données synthétiques et réelles à l'aide de cet algorithme PMCMC. Nous ne présenterons à chaque fois les résultats que pour l'un des modèles (M_0 pour les données synthétiques, M_2 pour les réelles), cependant l'ensemble des résultats pour les modèles sont disponibles sur le dépôt Github du projet.

4.1 Données synthétiques

Nous avons commencé par appliquer l'algorithme AdPMCMC à des données synthétiques simulées selon les modèles M_0 à M_4. Un exemple des trajectoire obtenues par filtre particulaire sont présentées ci-après (selon M_0 à gauche, M_2 à droite). La variance de la log-vraisemblance obtenue par SMC est de l'ordre de l'unité, indiquant une bonne adaptation aux données.
Les résultats d'estimation des paramètres présentés ci-après ont été obtenus par AdPMCMC sur le modèle M_0 avec $L = 150$ particules, sur 50.000 itérations avec un taux d'acceptation global d'environ 13%. Le seuil adaptatif (probabilité que l'on simule avec un noyau adaptatif) a été fixé à 50%.
L'initialisation a été effectuée selon les valeurs données dans l'article.

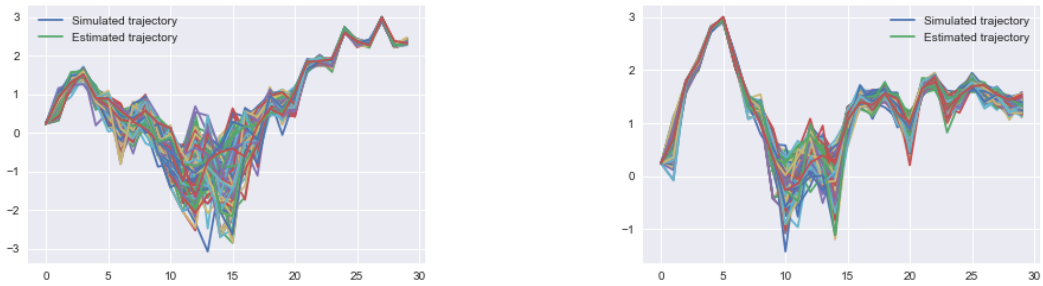

FIGURE 2 – Trajectoires simulées par SMC pour M_0 (gauche) et M_2 (droite)

Les distributions a posteriori ainsi que des nuages de point des paramètres du modèle M_0 sont représentés dans la Figure 3, ainsi que l'illustration de la période de burn-in concernant le paramètre σ_w.[3].

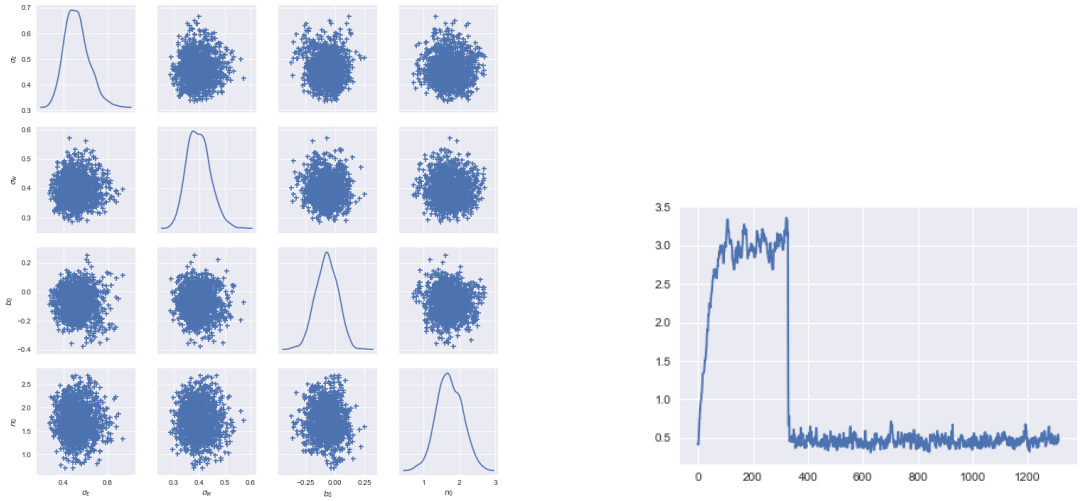

FIGURE 3 – Distributions à posteriori et nuages de point des paramètres pour le modèle M_0 (a gauche), convergence de la chaîne de Markov simulant la loi à posteriori de σ_w (à droite)

4.2 Données réelles

Nous avons par la suite appliqué ces modèles aux données Nutria qui représentent l'évolution d'une population de ragondins sur un dizaine d'années. Les données sont représentées dans la figure 4.

3. Notons que l'on retrouve exactement le même comportement que celui dans l'article avec une chute brutale vers une valeur autour de laquelle les échantillons se concentrent

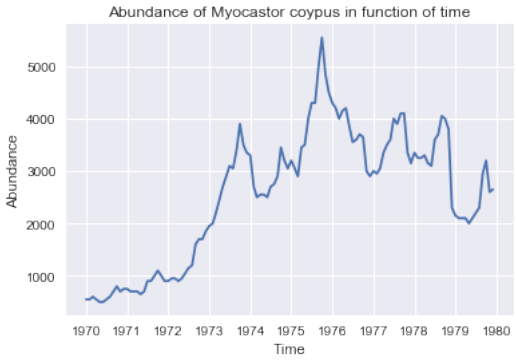

FIGURE 4 – Evolution de la population de myocastor coypus

Les résultats présentés sur la Figure 5 ont été obtenus en appliquant le modèle M_2 (theta-logistic) aux données. L'algorithme AdPMCMC a été utilisé avec 50.000 itérations et 150 particules, le seuil adaptatif étant une nouvelle fois fixé à 50%. Sur les 50.000 itérations, 13.000 ont été acceptés soit un très fort taux d'acceptation de 26%. Un burn-in initial de 5000 itérations a été utilisé pour les graphiques qui suivent. Nous retrouvons des graphiques similaires à ceux de l'article.

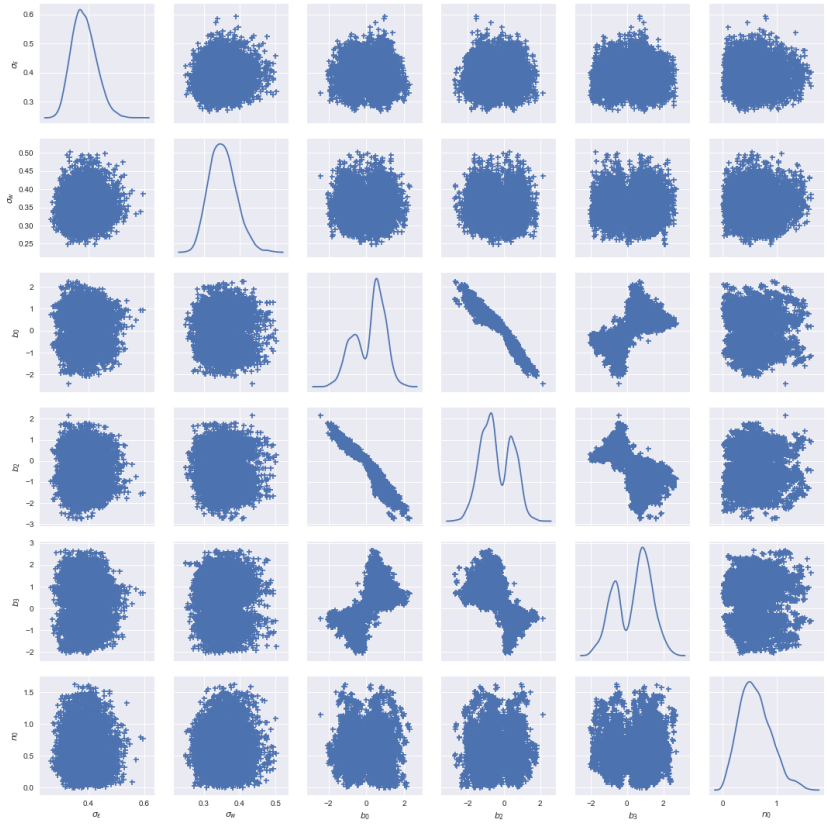

FIGURE 5 – Densités à posteriori (estimées) et nuages de points des paramètres estimés par AdPMCMC pour le modèle M_2 (de droite à gauche : $\sigma_\epsilon, \sigma_w, b_0, b_2, b_3, n_0$)

Références

[ADH10] Christophe Andrieu, Arnaud Doucet, and Roman Holenstein. Particle markov chain monte carlo methods. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 72(3) :269–342, 2010.

[PHH10] Gareth W. Peters, Geoff R. Hosack, and Keith R. Hayes. Ecological non-linear state space model selection via adaptive particle markov chain monte carlo (adpmcmc). 2010.