Projet STA211 - Sujet 1 au choix

"Méthodes de simulation numérique statistique"

Sophie Ancelet et Merlin Keller

Mai 2021

Ce projet peut être réalisé seul ou en binôme. Sa réalisation nécessite un ordinateur. Vous rédigerez :

— soit un fichier RMarkdown intégrant simultanément un rappel des questions, vos réponses écrites à ces questions et vos codes R

— soit un document word/pdf intégrant un rappel des questions et vos réponses écrites à ces questions ainsi qu'un fichier R contenant vos codes.

Attention ! Vos réponses doivent être systématiquement justifiées et les fichiers de code transmis doivent être **directement exécutables** sous R. Vos fichiers seront à envoyer **au plus tard le mercredi 19 mai 2021** aux deux adresses suivantes : **sophie.ancelet@irsn.fr et merlin.keller@edf.fr** avec pour objet DMSTA211 suivi de votre nom (ou de vos deux noms si vous travaillez en binôme).

Estimation d'une taille de population à partir de données de capture-marquage-recapture

Les méthodes de capture-marquage-recapture sont des méthodes astucieuses d'échantillonnage non destructif pour évaluer le nombre (inconnu) d'individus N dans une population. Dans le domaine de la gestion halieutique, par exemple, elles consistent à effectuer un certain nombre de pêches successives, avec remise, à l'aide d'un dispositif de capture (le plus souvent pêche électrique) d'efficacité π. L'efficacité π est la probabilité de capture d'un poisson : elle dépend du milieu et de l'effort de pêche, et peut dépendre de la taille du poisson d'intérêt. Les poissons capturés à chaque pêche sont marqués puis remis à l'eau. Utilisées principalement en écologie, les méthodes de capture/marquage/recapture trouvent aussi des applications de portée bien plus large pour des recensements menés dans différents domaines.

Dans cet exercice, nous considérons le cas de 2 expériences successives de pêche (avec marquage et remise) réalisées afin d'estimer le nombre (inconnu) de poissons N dans un lac. On appelle C_1 et C_2 le nombre total de poissons capturés et marqués lors des pêches 1 et 2 respectivement. On appelle C_{20} le nombre de poissons non marqués capturés lors de la deuxième pêche et C_{21} le nombre de poissons marqués capturés lors de la deuxième pêche. On a donc : $C_2 = C_{20} + C_{21}$.

Les données disponibles proviennent d'une expérience réelle "miniature" de capture-marquage-recapture réalisée par des étudiants à l'aide d'un saladier ("le lac") rempli de riz ("l'eau du lac") et de haricots blancs ("les poissons"). Les données observées par les étudiants sont les suivantes : $C_1 = 125$**,** $C_{20} = 134$ **et** $C_{21} = 21$.

On considère le modèle probabiliste $\mathcal{M}$ suivant :

$$
\begin{aligned}
C_1 &\sim Binomial(N, \pi) \\
C_{20}|C_1 &\sim Binomial(N - C_1, \pi) \\
C_{21}|C_1 &\sim Binomial(C_1, \pi)
\end{aligned}
$$

Pour commencer...

1 Montrer que la log-vraisemblance du modèle probabiliste $\mathcal{M}$ s'écrit : $log([C_1 = c_1, C_{20} = c_{20}, C_{21} = c_{21}|\pi, N]) = log\left(C_N^{c_1} C_{n-c1}^{c_{20}} C_{c_1}^{c_{21}}\right) + (c_1 + c_2) log(\pi) + (2N - c_1 - c_2) log(1 - \pi)$

2 Pour tout $u \in [0, 1]$, écrire l'expression de l'inverse généralisée de la fonction de répartition de la variable aléatoire C_1 de loi $Binomial(N, \pi)$. En déduire une fonction `R` qui génère par inversion générique `n` tirages indépendants de la variable aléatoire C_1. Utiliser cette fonction pour tirer un échantillon de $n = 10000$ réalisations de loi $Binomial(125, 0.15)$. Comparer les fréquences obtenues avec les fréquences théoriques.

3 Utiliser la fonction `R` précédente pour définir une seconde fonction `R` permettant de générer des réalisations possibles de capture-marquage-recapture (i.e. des variables aléatoires C_1, C_{20} et C_{21}) selon le modèle $\mathcal{M}$.

Supposons N connu

Supposons tout d'abord N=950 (connu) et estimons l'efficacité π.

4 Calculer l'estimateur du maximum de vraisemblance $\hat{\pi}_{MLE}$ du paramètre π. Sachant les données observées, en déduire une estimation de π.

5 Assignons une loi *a priori* beta(α, β) sur le paramètre π. Montrer que la loi *a posteriori* de π sachant N notée $[\pi|N, C_1, C_{20}, C_{21}]$ est alors une loi beta de paramètres $C_1 + C_2 + \alpha$ et $2N - C_1 - C_2 + \beta$. Donner l'expression de l'espérance de cette loi beta. Que remarquez-vous ?

6 Posons α=1 et β=3. Représenter sur un même graphe les densités *a priori* et *a posteriori* de π ainsi que l'estimateur du maximum de vraisemblance de $\hat{\pi}_{MLE}$. Commenter les résultats.

Supposons N et π inconnus

Approche fréquentiste

Pour évaluer le nombre d'individus N dans une population d'intérêt à partir de deux expériences de pêche de type capture-marquage-recapture, un estimateur fréquentiste naif est l'estimateur de "Petersen" défini par :

$$\hat{N} = \frac{C_1 C_2}{C_{21}}$$

7 Appliquer cet estimateur au jeu de données réelles observées afin d'estimer le nombre de "poissons" N dans "le lac".

8 Supposons ici que les "vraies" valeurs des paramètres soient $N_{true} = 923$ et $\pi_{true} = 0.15$. Simuler 100 jeux de données à l'aide de la fonction implémentée à la question 3, en déduire 100 estimations du paramètre N puis estimer empiriquement - par Monte-Carlo - le biais relatif de $\hat{N}$. Recommencer en faisant varier N_{true} de 100 à 1000 par pas de 10 puis représenter l'évolution du biais relatif en fonction de N_{true}. Discuter des résultats.

Approche bayésienne

Considérons une loi *a priori* beta($\alpha = 1, \beta = 3$) pour π et une loi uniforme sur l'ensemble fini d'entiers $\{1, \ldots, 2000\}$ pour N.

9 La question 5. a montré que la loi conditionnelle complète de π est : $\pi | N, y \sim Beta(C_1 + C_2 + \alpha, 2N - C_1 - C_2 + \beta)$. Donner l'expression de la loi conditionnelle complète de N (à une constante multiplicative près). Reconnaissez-vous une forme analytique connue ?

10 Implémenter un algorithme MCMC sous la forme d'une fonction `R` nommée `MCMC` qui va permettre d'échantillonner dans la loi jointe *a posteriori* du couple (N, π) sachant les données $y = (c_1, c_{20}, c_{21})$ en mettant à jour :

- le paramètre π avec un échantillonneur de Gibbs
- le paramètre N avec un échantillonneur de Metropolis-Hastings (MH), en utilisant comme loi de proposition une loi uniforme (discrète) sur $\{N^{curr} - k, N^{curr} + k\}$ où N^{curr} désigne la valeur courante du paramètre N à une itération donnée et k est un paramètre de saut.

11 **Choix du saut k :** Utiliser la fonction MCMC précédemment implémentée pour calculer puis tracer l'évolution du taux d'acceptation associé à la mise à jour de N en fonction de différentes valeurs du paramètre k (par exemple, allant de 1 à 301 par pas de 10). Pour chaque valeur de k, on pourra faire tourner l'algorithme MCMC pendant 10 000 itérations et qu'avec une seule chaîne de Markov pour cette étape de calibration. Quelle valeur de k vous semble la meilleure (rappel : viser un taux d'acceptation d'environ 40%) ? Vous conserverez cette valeur pour la suite.

12 Lancer à présent 3 chaînes de Markov à partir de positions initiales différentes en fixant k à la valeur précedemment choisie afin de générer 3 échantillons $((N^{(1)}, \pi^{(1)}), \ldots, (N^{(G)}, \pi^{(G)}))$ de taille $G = 20\,000$. Faites un examen visuel des chaînes de Markov obtenues et calculer la statistique de Gelman-Rubin. Identifiez-vous un problème de convergence de l'algorithme MCMC implémenté vers sa loi stationnaire ? Si oui, comment proposez-vous d'y remédier ? Combien d'itérations `X` vous semblent a minima nécessaires pour espérer avoir atteint l'état stationnaire ?

13 Analyser les autocorrélations intra-chaînes. Qu'en pensez-vous ?

14 Supprimer les `X` premières itérations correspondant à votre temps-de-chauffe "estimé" de l'algorithme afin de constituer votre échantillon *a posteriori*. Calculer la taille d'échantillon effective (ESS) de l'échantillon *a posteriori* constitué. Qu'en pensez-vous ? Si l'ESS vous semble trop petit, refaites tourner l'algorithme en augmentant le nombre d'itérations G jusqu'à obtenir un ESS "satisfaisant" pour bien estimer N et π.

15 Donner les statistiques *a posteriori* et représenter les lois *a posteriori* approchées pour les paramètres inconnus de votre modèle. Comparer les résultats obtenus à ceux obtenus avec une approche fréquentiste.