Projet STA211 - Sujet 2 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).

Ajustement d'une loi de Weibull sur des données de durée de vie d'un composant industriel, avec censures à droite

On cherche à estimer la distribution $\mathcal{P}$ de la durée de vie T d'un composant industriel (batterie de portable ou de voiture, turbine d'une centrale à énergie renouvelable, ...). On dispose pour cela d'un jeu de données de n temps de fonctionnement observés $t_1, \ldots, t_n$, où :

— pour $i = 1, \ldots, p$, t_i est un temps à défaillance, c'est-à-dire que la durée de vie T_i du i-ème composant est exactement t_i : $T_i = t_i$;

— pour $i = p+1, \ldots, n$, t_i est une censure à droite, c'est-à-dire que la durée de vie T_i du i-ème composant est au moins $T_i \geq t_i$;

On suppose de plus que les durées de vie suivent la loi de Weibull $\mathcal{W}_{\alpha,\kappa}$, de fonction de répartition :

$$\mathcal{P}[T_i \leq t|\alpha,\kappa] = F(t|\alpha,\kappa) = 1 - \exp\left(-\alpha\ t^{\kappa}\right).$$

Le but de cet exercice est donc de proposer plusieurs méthodes, fréquentistes et bayésiennes, pour estimer les paramètres α, κ à l'aide du jeu de données $t_{1:n}$ et de comparer leurs résultats.

Les données sont fournies dans le fichier `donnees_Weibull_censuree`

Simulation

1 Calculer la fonction quantile $F^{-1}(p|\alpha,\kappa)$ de la loi de Weibull, et en déduire un algorithme de simulation de cette loi basée sur l'inversion générique.

2 Ecrire un programme `R` qui simule `n` observations de la loi de Weibull de paramètres `alpha` et `kappa` donnés, censurée au-dessus d'un niveau `t0` donné (concrètement, on censure au-dessus de t_0 en remplaçant chaque valeur simulée T par $\min(T, t_0)$). Le programme renvoie le vecteur de

données simulées, ordonné de telle sorte que les p premières observations ne soient pas censurées, ainsi que le nombre p.

Calcul de la vraisemblance

3 Montrer que la log-vraisemblance $\ell(t_{1:n}|\alpha,\kappa,p)$, dans le modèle de Weibull dont les $n-p$ dernières données sont censurées à droites, s'écrit :

$$\ell(t_{1:n}|\alpha,\kappa,p) = p(\log\alpha + \log\kappa) + (\kappa-1)\sum_{i=1}^{p}\log t_i - \alpha\sum_{i=1}^{n} t_i^{\kappa}$$

4 Donner l'expression exacte du gradient, et de la matrice hessienne de ℓ

Approche fréquentiste

5 Montrer que l'estimateur du maximum de vraisemblance $\widehat{\alpha}_{MLE}$ se déduit de l'estimateur du maximum de vraisemblance $\widehat{\kappa}_{MLE}$ à l'aide d'une formule qu'on explicitera, puis montrer que pour calculer ce dernier il faut résoudre un problème d'optimisation en 1D.

6 À l'aide de la fonction `optimize`, écrire un programme `R` qui prend en entrée le vecteur `t` des durées de fonctionnement observées ; le nombre `p` de données non censurées, et qui calcule les estimateurs du maximum de vraisemblance de (α,κ).

7 Estimer (α,κ) par maximum de vraisemblance 100 fois, à partir de 100 jeux de données simulés à l'aide de la fonction écrite en première partie, pour les choix suivants de paramètres :

- $(\alpha,\kappa) = (5,2)$;
- $t_0 = F^{-1}(0.6|5,2)$ le quantile à 60% de la loi de Weibull simulée. On fera attention au fait que le nombre p de données non censurées varie à chaque simulation
- Estimer empiriquement par Monte-Carlo :
  - le biais et la variance de chaque estimateur
  - le coefficient de variation de chaque estimateur

  Discuter des résultats.

Approche bayésienne

8 Montrer que, si la loi *a priori* sur α est la loi Gamma $\mathcal{G}(a,b)$, alors la loi *a posteriori* conditionnelle de α sachant κ, notée $\pi(\alpha|\kappa,t_{1:n},p)$, est encore une loi Gamma, dont on précisera les hyperparamètres.

Dans la suite, on prendra de même une loi *a priori* de type Gamma pour κ, de paramètres c et d, et on utilisera le choix "faiblement informatif" suivant : $a=b=c=d=10^{-3}$.

9 Montrer que la densité marginale *a posteriori* de κ est proportionnelle à :

$$\pi(\kappa|t_{1:n},p) \propto \pi(\kappa)\kappa^n \prod_{i=1}^{n} t_i^{\kappa-1} \left(b+\sum_{i=1}^{n} t_i^{\kappa}\right)^{-(a+n)}$$

On cherche à simuler cette densité marginale *a posteriori* $\pi(\kappa|t_{1:n},p)$ de κ par une méthode d'acceptation-rejet, en utilisant une loi instrumentale $g(\kappa)$ de type Gamma.

(a) Montrer en supposant que $t_{\max} = \max_i t_i > 1$ les équivalents suivants :

$$\pi(\kappa|t_{1:n},p) \overset{\kappa\to 0^+}{\sim} \pi(\kappa)\kappa^p$$
$$\pi(\kappa|t_{1:n},p) \overset{\kappa\to\infty}{\sim} \pi(\kappa)t_{\max}^{\kappa(a+n)}.$$

En déduire quelles contraintes doivent respecter les paramètres (e,f) de la loi instrumentale $g(\kappa) = \mathcal{G}(\kappa|e,f)$ pour que le rapport $\pi(\kappa|t_{1:n},p)/g(\kappa)$ reste borné.

(b) Calculer $\widehat{\kappa}_{MAP} = \arg\max_\kappa \pi(\kappa|t_{1:n},p)$ à l'aide de la fonction `optimize` et l'approximation de Laplace de la variance *a posteriori* de κ, donnée par :

$$\widehat{\sigma}^2_{\kappa_{MAP}} = -\left(\frac{\partial^2 \log \pi(\kappa|t_{1:n},p)}{\partial\kappa^2}\right)^{-1}\Big|_{\kappa=\widehat{\kappa}_{MAP}}.$$

(on pourra utiliser la fonction `hessian` du package `numDeriv` pour calculer la dérivée seconde).

(c) Choisir pour loi instrumentale g la loi Gamma d'espérance égale à $\widehat{\kappa}_{MAP}$ et de variance égale à $\sigma^2_{\kappa_{MAP}}$, en vérifiant qu'elle respecte les conditions en 1 pour que le rapport $\pi(\kappa|t_{1:n},p)/g(\kappa)$ soit borné.

(d) Toujours en utilisant la fonction `optimize`, calculer le sup du quotient $M = \sup_\kappa \frac{\pi(\kappa|t_{1:n},p)}{g(\kappa)}$

10 Ecrire un programme `R` qui génère - par une méthode d'acceptation-rejet - un échantillon $\kappa_1,\ldots,\kappa_G$ de taille $G = 10\,000$ selon la densité marginale *a posteriori* $\pi(\kappa|t_{1:n},p)$. Puis, pour $g = 1,\ldots,G$, simuler α_g dans la loi conditionnelle *a posteriori* $\pi(\alpha|\kappa,t_{1:n},p)$ pour $\kappa = \kappa_g$. Représenter les densités *a priori* et *a posteriori* pour chaque paramètre, ainsi que le graphe de corrélation *a posteriori* du couple $(\kappa,\alpha^{-1/\kappa})$.

11 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 (α,κ) sachant les données $t_{1:n}$ et p en mettant à jour :

— le paramètre α avec un échantillonneur de Gibbs

— le paramètre κ avec un échantillonneur de Metropolis-Hastings (MH) basé sur une loi de proposition Normale centrée en la valeur courante κ_c et d'écart-type δ :

$$g(\kappa|\kappa_c) = \mathcal{N}(\kappa|\kappa_c,\delta^2)$$

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

13 Lancer à présent 3 chaînes de Markov à partir de positions initiales différentes en fixant δ à la valeur précedemment choisie afin de générer un échantillon $((\alpha^{(1)}, \kappa^{(1)}), \ldots, (\alpha^{(G)}, \kappa^{(G)}))$ de taille $G = 10\,000$. 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 ?

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 α et κ.

15 Représenter les densités *a priori* et *a posteriori* pour chaque paramètre, ainsi que le graphe de corrélation *a posteriori* du couple $(\kappa, \alpha^{-1/\kappa})$. Comparer les résultats issus des deux algorithmes d'inférence bayésienne. Lequel préférez-vous ?