MÉTHODE DE MONTE-CARLO
EXERCICES DU 9 MARS 2015

par Rémi Peyre

EXERCICE 1 — Processus stable

Un processus stable *(normalisé) d'exposant $\alpha \in (0, 2)$ est un processus à valeurs dans $\mathbb{R}$, markovien, évoluant par sauts, défini de la façon suivante : pendant l'intervalle de temps dt, la probabilité que survienne un saut d'amplitude dans $[x, x + dx]$ vaut (indépendamment de tout ce qui s'est passé avant) $|x|^{-1-\alpha} dxdt$, pour tout $x \in \mathbb{R}$* [*]*. (On suppose par ailleurs le processus issu de 0 au temps 0). Ici nous allons considérer une approximation de ce processus, où on ne retient que les sauts d'amplitude (en valeur absolue) supérieure à ε, où ε est une certaine valeur de troncature. Dans cet exercice, nous prendrons $\alpha = 1{,}5$ et $\varepsilon = 10^{-2}$ — dans les codes qu'on écrira, il est demandé de traiter ces paramètres comme des macros* [†]*, afin de pouvoir les modifier facilement si on le voulait.*

1. (✿) Justifier que les instants auxquels se produisent les sauts (indépendamment de leur intensité) forment un « processus ponctuel de Poisson » d'intensité constante, càd. que la probabilité qu'un saut se produise dans un intervalle de temps $[t, t + dt]$ vaut (indépendamment de tout ce qui s'est passé avant) λdt, où λ, appelée l'« intensité » du processus ponctuel de Poisson, vaut ici

$$\lambda := 1\ 333{,}333.$$

(On calculera l'expression littérale de λ dans le cas général).

2. Écrire une fonction `Poisson_ponct` qui représente les instants des sauts. Cette fonction prendra en argument un temps T, ne renverra rien, et affichera l'ensemble des instants de saut, en traçant un point sur l'axe des abscisses à chaque instant correspondant. (On testera la fonction pour $T = 0{,}05$ p. ex.).

3. (✿) Justifier que, conditionnellement au fait qu'il se produise un saut à un instant donné, la loi de l'amplitude de ce saut est (indépendamment de tout se qui s'est passé avant cet instant) une loi à densité sur $\mathbb{R}$ notée μ, définie par :

$$d\mu(x) := \lambda^{-1} \mathbf{1}_{\{|x| \geqslant \varepsilon\}} |x|^{-1-\alpha} dx.$$

4. Écrire un programme `mu` qui simule la loi μ. Ce programme ne prendra aucun argument et retournera la valeur simulée.

5. Écrire une fonction `proc_stable` qui trace la trajectoire du processus stable. Cette fonction prendra en argument un temps T, ne renverra rien, et affichera la trajectoire du processus sur $[0, T]$. Je vous demande de tracer cette trajectoire, non pas en traçant la courbe exacte tenant compte de *chaque* saut, mais en calculant les valeurs de la courbe pour les temps qT avec $q \in \{0, 1/1\ 024, \ldots, 1\ 023/1\ 024, 1\}$.

6. Évaluer, par la méthode de Monte-Carlo, la probabilité que le processus stable ne dépasse pas (en valeur absolue) le seuil $y_1 := 1$ avant l'instant $T_1 := 4$.

---

[*]. En toute rigueur, pour un tel processus dont la densité de sauts est infinie, cette définition laisserait encore deux paramètres implicites non fixés, mais nous ne nous en préoccuperons pas ici, dans la mesure où la troncature évite ce problème.

[†]. Sous Matlab, cela se fera par le biais de variables globales définies une unique fois au début du code.

On considère dorénavant un autre processus (qu'on différenciera du précédent à l'aide du symbole '~'), qui évolue également poisonniennement par sauts, la densité des sauts valant également λ*, mais où la loi des sauts est maintenant la loi* $\tilde{\mu}$ *définie par :*

$$d\tilde{\mu}(x) := Z^{-1}\mathbf{1}_{\{|x|\geqslant\varepsilon\}}e^{-(x/y_1)^2}|x|^{-1-\alpha},$$

où Z *est la constante qui fait de* $\tilde{\mu}$ *une mesure de probabilité. Nous admettrons qu'il est possible de calculer numériquement la valeur de* Z *avec une grande précision, et qu'on alors celle-ci égale (pour les paramètres décrits ci-dessus) à*

$$Z = 1\ 328{,}899.$$

7. Écrire un programme `mutilde` simulant $\tilde{\mu}$.
Indication : Penser à la méthode de rejet, avec pour mesure de référence μ.

8. **(★)** Quelle est la densité de μ par rapport à $\tilde{\mu}$? En déduire comment calculer la densité de probabilité, pour une trajectoire donnée, de la loi du processus non tildé par rapport à la loi du processus tildé.
Indication : On calculera la densité relative comme un produit qu'on maintiendra au cours de la boucle de calcul de chaque simulation ; où à chaque saut d'intensité x, on multipliera la densité relative par $d\mu/d\tilde{\mu}(x)$.

9. Appliquer une méthode d'échantillonnage préférentiel pour améliorer la précision de l'estimation précédente.

EXERCICE 2 — Intermittences

Une usine produit des boulons tout au cours de la journée, de 08 h 00 à 18 h 00. En fonctionnement normal, elle produit 100 000 boulons par heure. Toutefois, il faut exactement 1 h *de préchauffage du système avant que la production puisse démarrer (y compris le matin à 08 h 00). En outre, l'approvisionnement électrique de l'usine est vétuste : l'électricité tombe en panne (en moyenne) une fois toutes les* 3 h*, avec des pannes qui durent de l'ordre de* 1 h*. Plus précisément, on modélise les pannes électriques ainsi :*

- *À un instant donné (donc en particulier le matin à 08 h 00), l'électricité a* 1/4 *risque d'être en panne*[‡].
- *Quand l'électricité est en panne, il faut attendre une loi exponentielle de moyenne* 1 h *avant qu'elle soit réparée.*
- *Quand l'électricité fonctionne, elle tombe en panne au bout d'une loi exponentielle de moyenne* 3 h.

1. Simuler l'évolution de l'état du réseau électrique au cours de la journée. On présentera le résultat sous la forme d'un graphique, où « en panne » sera représenté par -2 et « en service » par -1. (On présentera la réponse à cette question sous la forme d'un programme `reseau_graph`, qui ne prendra aucun argument et ne reverra aucune valeur, et affichera le graphique souhaité). Vérifier la cohérence des résultats obtenus.

2. Modifier le programme précédent (en un programme `reseau_usine_graph`) pour faire en sorte de tracer aussi sur la graphique l'évolution de l'état de l'usine au

[‡]. Cela vient de ce qu'on pourrait démontrer que, lorsque cela fait suffisamment longtemps que le processus « réseau électrique » a démarré, la probabilité que l'électricité soit en panne à un instant donné converge vers 1/4.

cours de la journée. (On représentera l'état « à l'arrêt » par 0, l'état « en préchauffage » par 1/2, et « en fonctionnement » par 1). Vérifier la cohérence des résultats obtenus.

3. Justifier que le processus « état du réseau électrique » est markovien, mais que le processus « état de l'usine » n'est *pas* markovien. Justifier qu'en revanche, si l'état de l'usine au cours du préchauffage n'est plus représenté par une constante, mais par le temps écoulé depuis le début du préchauffage [§], alors le processus « (état du réseau électrique, état de l'usine) » est markovien, et même le processus « état de l'usine » tout seul.

4. Modifier la programme précédent (en un programme `reseau_usine_graph_prod`) pour qu'il ait maintenant une valeur de sortie, correspondant au nombre de boulons total produit sur la journée [¶]. Vérifier la cohérence des résultats obtenus.

5. Évaluer par la méthode de Monte-Carlo la production journalière moyenne de l'usine. Le programme, appelé `MC_prod`, prendra en argument le nombre de simulations demandées, ne renverra rien, et affichera une estimation de la production journalière moyenne, assortie d'un intervalle de confiance.

6. Démontrer que, à 17 h 00, le nombre moyen de boulons que l'usine continuera de produire d'ici la fin de la journée vaut :

– 0 si l'usine est à l'arrêt ;

– 85 040,607 si l'usine est en fonctionnement ;

– 214 959,393 $\times (e^{t/3} - 1)$ si cela fait un temps t que le préchauffage a commencé.

Indication : En fait, j'ai mis les valeurs numériques juste que vous puissiez sauter la question sans danger ; la question consiste en substance à retrouver (en la justifiant) l'expression littérale des formules ci-dessus.

7. En déduire une méthode de conditionnement pour le calcul du nombre de boulons moyens produit, implémentée dans un programme `MC_prod_cond`. Vérifier la cohérence des résultats obtenus. L'amélioration de l'efficacité est-elle intéressante ? Pourvait-on s'en douter ?

Exaspérée par les pannes à répétition de l'électricité, sur lesquelles elle n'a pas de pouvoir, la directrice de l'usine envisage de changer la chaine de production : le rendement en fonctionnement serait le même, mais le préchauffage du système ne durerait plus qu'une demi-heure. Comme cette nouvelle chaine serait onéreuse à mettre en place, la directrice cherche à savoir quelle amélioration en résulterait sur la production journalière moyenne.

8. Estimer l'amélioration apportée, en utilisant une méthode de couplage. (Programme `MC_prod_coupl`). Vérifier la cohérence des résultats obtenus.

9. **(★)** Employer le conditionnement *en plus* du couplage pour améliorer l'efficacité du résultat. Vérifier la cohérence des résultats obtenus.

EXERCICE 3 — File d'attente

Le but de cet exercice est de simuler un phénomène de file d'attente*, par exemple à un péage sur l'autoroute, sous une hypothèse d'arrivée poissonnienne des voitures et de traitement poissonnien de leurs requêtes* [‖].

[§]. Mais n'utilisez pas cette représentation du préchauffage dans vos réponses.

[¶]. On ne se préoccupera pas ici du fait que les formules donnent des nombres de boulons non entiers.

[‖]. Cette seconde hypothèse est évidemment peu réaliste concernant le cas d'un péage, mais ce n'est qu'un exercice...

Ce qu'on va simuler est l'évolution au cours du temps (noté t, compté en minutes) du nombre V_t de voitures attendant au péage. Ce nombre de voitures évolue markoviennement, avec les règles suivantes. Au cours d'un intervalle de temps dt :

- S'il y a au moins une voiture qui attend*, la probabilité qu'une voiture passe (et donc que V_t diminue d'une unité) est αdt, avec $\alpha = 4\ \mathrm{min}^{-1}$;*
- *La probabilité qu'une nouvelle voiture arrive (et donc que V_t augmente d'une unité) est βdt, où β est la densité du trafic.*

1. Sachant la valeur de V_t à un instant donné, quelle est la loi du temps d'attente avant que V_t évolue, et la loi du saut que V_t fera à cet instant ?

2. Écrire un programme simulant $(V_t)_{0 \leqslant t \leqslant T}$ (en partant de $V_0 = 0$), les arguments du programme étant T et β. Lancer les simulations pour $T = 720$ et respectivement $\beta = 0{,}5\alpha$, $\beta = 0{,}95\alpha$ et $\beta = 1{,}05\alpha$. Commenter.

3. Pour *une* simulation donnée, exprimer le temps d'attente moyen (sur *cette* simulation) τ des automobilistes en fonction de $(V_t)_t$. (Si $V_T > 0$, on fera comme si les voitures attendant encore au temps T étaient alors traitées instantanément). Modifier le programme pour qu'il retourne également cette valeur.
Indication : La réponse est que le temps d'attente moyen est l'intégrale de V, divisée par le nombre total de sauts vers le haut. Voyez-vous pourquoi ?...

4. (★) Expliquer pourquoi est-ce que, lorsqu'on réalise plusieurs simulations, le temps moyen d'attente des automobilistes sur l'ensemble des simulations n'est *pas* la moyenne des temps d'attente moyens sur chaque simulation.

5. (✿) Montrer que le nombre moyen d'automobilistes arrivant au cours d'une journée de durée T vaut exactement αT.

6. (★) Déterminer le temps d'attente moyen d'un automobiliste par la méthode de Monte-Carlo.
Indication : Le temps d'attente moyen des automobilistes n'est pas l'espérance du temps d'attente moyen par journée, mais on peut toutefois le déterminer à partir de deux autres espérances... Lesquelles ?