Feuille de travaux pratiques - Python #4

Emeline LUIRARD

Dans ce TP, nous allons voir comment simuler une chaîne de Markov avec Python et comment illustrer certaines de leurs propriétés. Par chaîne de Markov, on entendra ici chaîne de Markov discrête, homogène en temps, à espace d'états au plus dénombrable. Dans la suite (X_n) sera une chaîne de Markov à espace d'états E et de matrice de transtion P.

1 Simulation de chaînes de Markov

1.1 A partir de sa matrice de transition

En Python, il n'y a pas (à ma connaissance) de fonction toute faite pour générer une chaîne de Markov à partir de sa matrice de transition. Mais ce n'est pas bien difficile et vous pouvez retrouver la méthode dans [Vig18][p. 98] :
On commence par créer la matrice de transition P dans un `array` (cf TP 1). On rappelle que

$$\mathbb{P}(X_{n+1} = y | X_n = x) = P(x, y).$$

Ensuite on va utiliser la fonction `np.random.choice(a,p)` qui renvoie une variable aléatoire à valeurs dans a. p contient les probabilités associées. Ici, l'espace d'états E est en bijection avec $[\![0, Card(E) - 1]\!]$.

```
def markov_avec_P(n,P,x0):
"simule les n premiers pas d'une chaine de Markov "
"à partir de sa matrice de transition"
        X=np.zeros(n,dtype=np.int) #on indique que X contiendra seulement des entiers.
        X[0]=x0
        for k in range(n-1):
                X[k+1]=np.random.choice(a=range(len(P)), p=P[X[k],:])
                # Les états sont numérotés de 0 à len(P)-1
        return X
```

Considérons par exemple le cas où l'espace d'états $E = \{\text{oui}, \text{non}, \text{ne se prononce pas}\}$ est de cardinal 3 et où la matrice de transition est donnée par

$$P = \begin{pmatrix} 7/10 & 2/10 & 1/10 \\ 1/10 & 8/10 & 1/10 \\ 1/10 & 1/10 & 8/10 \end{pmatrix}.$$

Dans cet exemple, on a alors $E' := \{0, 1, 2\}$ avec les correspondances naturelles entre états $0 =$ oui, $1 =$ non et $2 =$ ne se prononce pas. Dans ce cadre, la réponse `ans = 0. 0. 2. 1.` à la commande `markov_avec_P(4,P,1)` correspond à la trajectoire (oui, oui, ne se prononce pas, non) dans E qui a pour probabilité

$$\mathbb{P}_{\text{oui}}(X_1 = \text{oui}, X_2 = \text{ne se prononce pas}, X_3 = \text{non}) = P_{11}P_{13}P_{32} = \frac{7}{10} \times \frac{1}{10} \times \frac{1}{10} = \frac{7}{1000}.$$

N.B. *Cette méthode est à utiliser lorsque l'espace d'états E est petit et la matrice de transition P facilement codable en Python. Ainsi, pour simuler une marche aléatoire sur $\mathbb{Z}$, il va falloir procéder autrement.*

1.2 Avec la relation de récurrence

Une chaîne de Markov peut toujours s'écrire sous la forme $X_{n+1} = f(X_n, U_{n+1})$, pour $n \geq 0$, où f est une fonction mesurable à valeurs dans E et (U_n) est une suite v.a. i.i.d. et indépendante de X_0 (donc des $(X_k)_{k<n}$).

N.B. *Lorsque la fonction f dépend du temps n, on a devant nous une chaîne de Markov non-homogène en temps. La construction de proche en proche à l'aide de la relation de récurrence s'appliquera donc aussi à ces chaînes de Markov.*

Prenons l'exemple d'une marche aléatoire symétrique sur $E = \mathbb{Z}$. Sa matrice de transition est, certes, simple mais infinie. Cependant, nous avons la relation

$$X_{n+1} = X_n + U_{n+1},$$

où les variables (U_n) sont indépendantes de loi $\mathbb{P}(U_n = 1) = \mathbb{P}(U_n = -1) = 1/2$.

Exercice 1. Modèle de Wright-Fisher
Soient k et N deux entiers tels que $0 < k < N$. On considère la chaîne de Markov $(X_n^N)_{n\geq 0}$ à valeurs dans $E := \{0, \ldots, N\}$, issue de $X_0^N := k$ et dont la matrice de transition est donnée par :

$$P_{ij} = \mathbb{P}(X_{n+1}^N = j | X_n^N = i) = \binom{N}{j} \left(\frac{i}{N}\right)^j \left(1 - \frac{i}{N}\right)^{N-j}.$$

1. *Écrire un programme qui étant donné un entier N, renvoie la matrice de transition P de la chaîne ci-dessus.*
2. *En utilisant la première méthode, écrire un programme qui prend en entrée les entiers k, N, n et qui génère et trace une trajectoire $(X_m^N)_{m=0\ldots n}$.*
3. *En utilisant la deuxième méthode, écrire un programme qui prend en entrée les entiers k, N, n et qui génère et trace une trajectoire $(X_m^N)_{m=0\ldots n}$.*
4. *Pour des trajectoires de longueur $n = 100$, à l'aide de la fonction `time.clock()` du module `time` (à importer), comparer le temps de calcul des deux programmes précédents, pour différentes valeurs de N.*

2 Mise en évidence des propriétés

Là où l'algèbre linéaire et les probabilités se rencontrent...

2.1 Etats recurrents et transients

Pour mettre en évidence, à l'aide de simulations, les états récurrents et transitoires d'une chaîne à espace d'états fini, on peut par exemple simuler une longue trajectoire de la chaîne et représenter la proportion de temps passé dans chaque état. Cette proportion sera négligeable pour les états transitoires, contrairement aux états récurrents.

2.2 Mesure invariante

On rappelle qu'une mesure π est invariante pour la chaîne de Markov $(X_n)_n$ de matrice de transition P si

$$\forall y \in E, \sum_x \pi(x)P(x,y) = \pi(y),$$

ou encore $\pi P = \pi$ en prenant π sous la forme d'un vecteur ligne. π est donc un vecteur propre à droite de P associé à la valeur propre 1, c'est à dire aussi, un vecteur propre à gauche de ${}^tP$ pour la valeur propre 1. La commande `eigVal,eigVec=np.linalg.eig(P.T)` vous sera donc très utile afin de trouver une mesure invariante. Et la commande `np.real` vous permettra d'avoir le résultat sous forme de réels.

La probabilité invariante d'une chaîne de Markov irréductible se retrouve également à l'aide du théorème suivant. En effet, rappelons que sur un espace d'états fini, lorsque qu'une chaîne de Markov est irréductible, elle est automatiquement récurrente positive et admet ainsi une unique probabilité invariante.

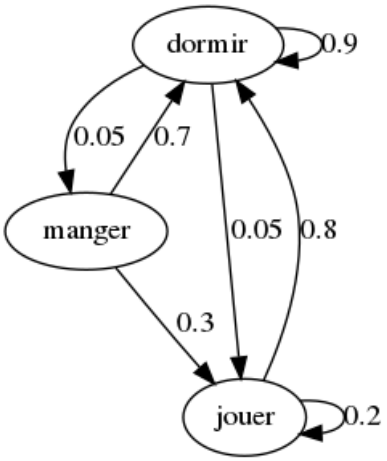
Théorème 2.1. *On note X^x la chaîne qui part du point x, π son unique probabilité invariante et on introduit*

$$T_n^x(y) = \frac{1}{n}\sum_{k=0}^{n-1} \mathbb{1}_{\{X_k^x = y\}}.$$

Alors $(T_n^x(y))_n$ converge quand n tend vers l'infini vers $\pi(y)$.

Lorsque la chaîne est, en plus, apériodique, on sait qu'elle converge en loi vers la mesure invariante.

Exercice 2. *Le chat Gribouille ne réalise que 3 actions dans sa journée : dormir, manger et jouer. la probabilité de changer d'activité à chaque unité de temps ne dépend que de l'activité précédente et pas du reste du déroulement de sa journée ni de l'heure. Les probabilités de transition sont données sur le graphe de transition suivant :*



1. *Est-ce une chaîne de Markov ? Est-elle homogène ? Quelles sont les classes d'équivalence, les états récurrents, transitoires ? Est-elle irréductible, récurrente, apériodique ?*
2. *On suppose que Gribouille commence par dormir. Au bout de 7 unités de temps, quelle est la probabilité qu il joue ? S'il commence par manger,quelle est la probabiltié qu'il dorme au*

bout de 42 unités de temps ?
Attention : `P**3` ne calcule pas la puissance matricielle.

3. *On suppose que Gribouille commence par manger. Quelles sont les diverses probabilités après n unités de temps, pour $n = 10, 25, 50, 100$? Y-a-t-il convergence en loi ?*
4. *Trouver la probabilité invariante de la chaîne de 3 manières différentes.*

Exercice 3. 1. *Écrire une fonction qui à partir de P, détermine la probabilité invariante π.*

2. *Inventer une matrice de transition P d'une chaîne de Markov irréductible, récurrente positive sur un espace d'états à 5 éléments.*
3. *Écrire une fonction qui mesure la distance d_n entre la matrice P^n et la matrice dont les lignes sont égales à π. Aide : Utiliser `np.linalg.norm`. Tracer $\log(d_n)/n$ en fonction de n. Faire le lien avec la décomposition spectrale de P.*
4. *À partir de la simulation d'une longue trajectoire de la chaîne, illustrer le théorème ergodique qui assure que presque-sûrement, pour toute fonction intégrable f*

$$\lim_{n\to+\infty} \frac{1}{n}\sum_{k=1}^{n} f(X_k) = \int_E f(x)\pi(dx).$$

Quel est l'ordre des fluctuations dans la limite ci-dessous ? Illustrer.

5. *En utilisant la méthode de Monte-Carlo, mettre en évidence la convergence en loi*

$$\lim_{n\to+\infty} \mathbb{P}(X_n = x) = \pi(x), \quad \forall x \in E.$$

On pourra par exemple tracer sur le même graphique l'histogramme empirique et celui associé à la loi π. Illustrer de même la convergence en variation totale :

$$\lim_{n\to+\infty} \sum_{x\in E} |\mathbb{P}(X_n = x) - \pi(x)| = 0.$$

6. *En utilisant la méthode de Monte-Carlo, estimer l'espérance du temps de retour en $x \in E$. Comparer à l'inverse de $\pi(x)$.*

Exercice 4. *Pierre joue pour la première fois au casino. Il possède 3 euros (pourquoi investir plus si on perd ?). Il ne veut s'arrêter que lorsqu'il aura gagner 7 euros ou qu'il n'aura plus de sous. Il reste tout le temps sur la même machine. Chaque tour lui coûte un euro. S'il gagne, il récupère deux euros. Cela se produit avec probabilité 0.4.*

1. *Modéliser le problème.*
2. *Quelles sont les classes d'équivalence, les états récurrents, transitoires ? Est-elle irréductible, récurrente, apériodique ?*
3. *Donner la loi de la fortune de Pierre au bout de 3 tours, 10 tours, 100 tours. Calculer une estimation de la probabilité que Pierre réussisse à gagner 7 euros et estimer la.*
4. *Quelle est la probabilité que Pierre s'arrête de jouer ? Estimer numériquement le nombre de fois que Pierre va jouer avant de s'arrêter. Quelle est la valeur théorique ?*
5. *Dans le cas où il gagne 7 euros, estimer numériquement l'espérance du nombre de fois qu'il va jouer avant de les gagner. Quelle est la valeur théorique ?*

3 Pour aller plus loin...

Exercice 5. *Ecrire une fonction qui permet de trouver les classes d'équivalences d'une chaîne de Markov à partir de sa matrice de transition.*
Aide : On pourra commencer par écrire une fonction qui trouve la plus grande chaîne possible qui va d'un état x à un état y sans passer deux fois par le même état.

Références

[Vig18] Vincent Vigon. *python proba stat.* Independently published, October 2018.