Préparation Oral HEC

Séance 3

Corrigé

Exercice à préparer (2023)

Une urne contient $n \geq 2$ boules distinctes $B_1, B_2, \dots, B_n$, que l'on tire successivement et avec remise. Pour tout $r \in \llbracket 1, n \rrbracket$, on note Y_r la variable aléatoire qui donne le rang du premier tirage au bout duquel les boules $B_1, B_2, \dots, B_r$ ont été tirées au moins une fois.

1. Cours : quel est le nombre de bijections d'un ensemble à n éléments dans un ensemble de n éléments ?

Il y a $n!$ bijections d'un ensemble à n éléments dans un ensemble à n éléments.
(pour le redémontrer dans le cas de l'ensemble $\llbracket 1, n \rrbracket$ par exemple : pour construire une telle bijection φ , on a n choix pour $\varphi(1)$, $n-1$ choix pour $\varphi(2)$, ..., et plus le choix pour $\varphi(n)$ si toutes les images précédentes ont été construites.)

2. Déterminer la loi, l'espérance et la variance de Y_1 .

D'après la définition Y_1 est le temps d'attente de l'événement « tirer la boule B_1 ». Les tirages étant indépendants (il y a remise) et le tirage de B_1 ayant lieu avec une proba $\frac{1}{n}$ (n boules), on a le schéma d'une loi géométrique : $Y_1 \hookrightarrow \mathcal{G}(1/n)$; d'où $E(Y_1) = n$ et $V(Y_1) = n^2(1 - \frac{1}{n}) = n(n-1)$.

3. Écrire une fonction en Python prenant en argument deux entiers n et r (on supposera $1 \leq r \leq n$, sans le vérifier) et qui simule une réalisation de la variable Y_r .

On peut proposer la fonction suivante, qui maintient une liste à r éléments valant 0 si la boule i n'a pas été tirée et 1 si elle l'a été. Pour plus de commodité on renumérote les boules de 0 à $n-1$, ce qui ne change rien à l'affaire.

```
def Y(n,r):
    boules_tirees = r*[0]
    nb_tirages = 0
    while np.sum(boules_tirees) < r:
        boule = rd.randint(n)
        if boule < r:
            boules_tirees[boule] = 1
            nb_tirages = nb_tirages + 1
    return nb_tirages
```

4. Préciser l'univers-image de Y_r . Calculer $\mathbb{P}(Y_r = r)$.

Assez clairement $Y_r(\Omega) = \llbracket r, +\infty \llbracket$ (il faut au moins r tirages pour tirer $B_1, \dots, B_r$).

On a $(Y_r = r)$ ssi on a tiré $B_1, B_2, \dots, B_r$ lors des r premiers tirages, dans n'importe quel ordre.

On raisonne par équiprobabilité. Il y a n^r résultats possibles pour les r premiers tirages ; parmi ceux-ci $r!$ tirages mènent à $(Y_r = r)$ (les manières d'ordonner $B_1, \dots, B_r$).

Ainsi $\mathbb{P}(Y_r = r) = \frac{r!}{n^r}$.

5. Montrer que l'événement A_r : « on tire les boules $B_1, \dots, B_{r-1}$ lors des r premiers tirages sans jamais tirer les boules $B_{r+1}, \dots, B_n$, et on tire la boule B_r au $(r+1)$ -ème tirage » a pour probabilité :

$$\mathbb{P}(A_r) = \frac{\binom{r}{2}(r-1)!}{n^{r+1}}$$

Là ça se corse !

On raisonne encore par équiprobabilité : sur les $r+1$ premiers tirages, on a n^{r+1} issues possibles. il s'agit de voir combien de ces issues mènent à A_r .

D'après la description de l'événement, il faut dénombrer les séquences de $r+1$ boules où B_r apparaît en $(r+1)$ -ème position, et les r premiers tirages sont constitués de $B_1, \dots, B_{r-1}$.

Parmi ces dernières boules, une et une seule apparaît 2 fois : il y a $(r-1)$ choix pour cette boule, et $\binom{r}{2}$ choix pour les 2 positions où elle apparaît. Pour compléter la séquence il suffit ensuite de prendre les $(r-2)$ boules non encore utilisées et de les distribuer dans les $(r-2)$ cases disponibles. Ceci donne $(r-2)!$ possibilités (les différents ordonnements).

Ainsi $\text{Card}(A_r) = (r-1)\binom{r}{2}(r-2)! = \binom{r}{2}(r-1)!$; puis $\mathbb{P}(A_r) = \frac{\text{Card}(A_r)}{n^{r+1}}$ ce qui est la formule attendue.

6. En déduire que :

$$\mathbb{P}(Y_r = r+1) = \frac{r(n-r)r! + \binom{r}{2}r!}{n^{r+1}}$$

Les issues traitées à la question précédente font partie de $(Y_r = r+1)$: c'est le cas où on ne tire pas d'autre boule que $B_1, \dots, B_r$. Mais attention on ne finit pas forcément par B_r ! Il faut compter ces séquences r fois, selon le choix de la boule qui achève les tirages.

La seconde possibilité est que, sur les $(r+1)$ tirages, il apparaisse une fois une boule B_i où $i > r$. Dénombrons ces séquences. Une séquence de $(r+1)$ tirages menant à $(Y_r = r+1)$ est telle que :

- on choisit la boule B_i ($i > r$) qui apparaîtra dans le tirage : $(n-r)$ choix
- on choisit sa position dans les $r+1$ tirages : attention on ne peut pas la mettre en dernier ! donc r choix.
- On dispose $B_1, \dots, B_r$ dans les tirages restants : $r!$ possibilités.

Avec tout ça on a $r(n-r)r!$ tirages.

D'où, en ajoutant les deux contributions :

$$\text{Card}(Y_r = r+1) = r\text{Card}(A_r) + r(n-r)r! = \binom{r}{2}r! + r(n-r)r!$$

puis

$$\mathbb{P}(Y_r = r + 1) = \frac{\text{Card}(Y_r = r + 1)}{n^{r+1}}$$

ce qui donne la formule attendue.

7. On fixe r . Pour tout $i \in \{1, \dots, r\}$, on note W_i la variable aléatoire représentant le nombre de tirages nécessaires pour que, pour la première fois, i boules distinctes parmi les boules $B_1, B_2, \dots, B_r$ soient sorties. On pose $X_1 = W_1$ et, pour tout $i \geq 2$, $X_i = W_i - W_{i-1}$.

- (a) Déterminer la loi de X_i ainsi que son espérance.

X_i est le temps d'attente de la i -ème boule parmi les $B_1, \dots, B_r$: on attend le tirage d'une boule parmi les $r - (i - 1)$ recherchées (le tout en effectuant des tirages équiprobables sur n boules).

Comme argumenté précédemment, $X_i \hookrightarrow \mathcal{G}\left(\frac{r - i + 1}{n}\right)$; et directement $\mathbb{E}(X_i) = \left(\frac{n}{r - i + 1}\right)$.

- (b) En déduire l'espérance de Y_n .

$Y = n$ est le temps d'attente de la collection complète ! Donc

$$Y_n = \sum_{i=1}^n X_i$$

Par linéarité de l'espérance

$$\mathbb{E}(Y_n) = \sum_{i=1}^n \mathbb{E}(X_i) = n \sum_{i=1}^n \frac{1}{n - i + 1} = n \sum_{k=1}^n \frac{1}{k}$$

en posant $k = n - i + 1$ (rappel : ici $r = n$).

- (c) Trouver un équivalent de $\mathbb{E}(Y_n)$ quand n tend vers $+\infty$.

Il faut savoir que $\mathbb{E}(Y_n) = n \ln(n)$ par équivalent de la somme harmonique. Je vous laisse aller voir en TD comment on montre cela (comparaison série-intégrale).

Exercice sans préparation (2023)

Un graphe est dit régulier s'il est simple et si tous ses sommets ont le même degré. Ce degré commun est alors appelé le degré du graphe régulier.

1. Écrire le code Python d'une fonction `regulier(M)` qui prend en entrée la matrice d'adjacence d'un graphe simple et qui renvoie un booléen égal à `True` si le graphe est régulier, à `False` sinon.
2. Déterminer les graphes réguliers de degré 0, 1 et 2.
3. Soit G un graphe régulier de degré 3. Que dire du nombre de sommets de G ?

Voir corrigé sur le rapport 2023 (planche 5).