TP bonus : chaines de Markov

Nicolas Jouvin

```r
library(dplyr)
library(ggplot2)
library(knitr)
```

Chaînes de Markov à temps discret : Exercice 50 du polycopié de S

Question 1

Faire l'exercice 50 du polycopié téléchargeable ici

Question 2

Créez la matrice de transition A suivante en R. Vérifiez que c'est bien une matrice stochastique, *i.e* que $\sum_j A_{ij} = 1$.

```
0.1 0.4 0.3 0.2
0.1 0.4 0.3 0.2
0   0.1 0.4 0.5
0   0   0.1 0.9
```

```r
A = matrix(c(0.1, 0.4, 0.3, 0.2,
             0.1, 0.4, 0.3, 0.2,
             0., 0.1, 0.4, 0.5,
             0., 0., 0.1, 0.9) ,ncol = 4, byrow = TRUE )
A
```

```
     [,1] [,2] [,3] [,4]
[1,]  0.1  0.4  0.3  0.2
[2,]  0.1  0.4  0.3  0.2
[3,]  0.0  0.1  0.4  0.5
[4,]  0.0  0.0  0.1  0.9
```

```r
rowSums(A) # sanity check
```

```
[1] 1 1 1 1
```

Question 2: Trouver la loi stationaire

On sait que la loi stationaire π vérifie $\pi^\top = \pi^\top A$ et $\sum_i \pi_i = 1$. On va l'estimer par plusieurs méthodes.

2.a) Méthode brute-force

> **Attention**: Ce n'est pas une méthode efficace !

On sait que A^n va converger vers une matrice constante par ligne, avec π en vecteur ligne. Coder une estimation de π en posant $n = 100$.

2.b) Via une décomposition en valeur propre et la fonction eigen()

π est le vecteur propre de A^T associé à la v.p. 1, et de norme l_1 unitaire. On va chercher P, D (diagonale), et P^{-1} telles que $A^T = PDP^{-1}$.

A l'aide de la fonction `eigen()` du package **MASS**,

1. retrouver la matrice P (vecteurs propres à droite) et la matrice D (les valeurs propres).
2. vérifier que 1 est bien valeur propre de A^T
3. trouver π

Comparer le π trouvé à celui de la méthode "brute-force" en norme l_1.

2.c) efficacité des 2 méthodes

La seconde méthode est plus général que la première car, avec une décomposition en valeur propre on peut calculer A^n pour tout n

$$(A^T)^n = PD^nP^{-1} \iff A^n = (PD^nP^{-1})^T$$

Coder A^n pour $n = 10^5$ en $\mathbf{R}$ de manière efficace.

Question 3: Système linéaire

Retrouver π en resolvant le système linéaire sur-contraint comme vu dans les slides.

Question 4: simulation de la chaîne

On se donne un $X_0 \in \{0, \dots, 3\}$ et un temps d'arrêt n. On souhaite simuler une chaine $(X_0, \dots, X_n)$

Coder la fonction `sample_chain(n, x_0)` en $\mathbf{R}$.

Question 5: Ergodicité

Avec n assez grand on s'attend à ce que la chaîne "oublie son passé" et visite les états conformément à la distribution stationnaire.

Proposez une petite simulation pour vérifier cela. Essayer différentes valeur de x_0 en point de départ.