DS10

Correction du devoir surveillé

Exercice 1

1. On a $A = \begin{pmatrix} 9/10 & 7/10 & 8/10 \\ 1/20 & 0 & 0 \\ 1/20 & 3/10 & 2/10 \end{pmatrix} = \frac{1}{20}\begin{pmatrix} 18 & 14 & 18 \\ 1 & 0 & 0 \\ 1 & 6 & 4 \end{pmatrix}$.

2. Analysons ce programme :

   - On démarre à `i = 1`. À chaque passage dans la boucle `for k = 1:10000`, le vecteur `a` contient une simulation des états par lesquels passe un Doudou initialement au repos pendant 100 minutes, et `a(n)` contient l'activité de ce Doudou à la 100-ème minute. On teste : si `a(n) == 1`, c'est-à-dire si Doudou dort à $n = 100$, alors on augmente `c(1)` de 1. À la fin de cette boucle, `c(1)` contiendra donc le nombre de Doudou dormant à la 100-ème minute.
   - De même, pour `i = 2`, `c(2)` contient le nombre de Doudou qui mange à la 100-ème minute, et pour `i = 3`, `c(3)` contient le nombre de Doudou courant à la 100-ème minute.
   - On renvoie `c/10000`, c'est-à-dire le vecteur contenant les `c(i)/10000` : il s'agit de la fréquence (effectif des Doudou à l'état i divisé par l'effectif total) des Doudou pour chaque activité (état) à la 100-ème minute.

   En l'exécutant, on obtient d'après l'énoncé :

   ```
   0.8813    0.045    0.0737
   ```

   On a donc une fréquence de `0.8813` des Doudou qui dorment à la minute 100, une fréquence de `0.045` de Doudou qui mangent, et une fréquence de `0.0737` de Doudou qui courent. Étant donné qu'on a reproduit un très grand nombre de fois cette expérience (10000 fois), on a donc une approximation de la loi de X_{100} par la loi faible des grands nombres :

   $$P(X_{100} = 1) \approx 0.8813, \quad P(X_{100} = 2) \approx 0.045, \quad P(X_{100} = 3) \approx 0.0737.$$

   Notons cependant que ce ne sont que des estimations ponctuelles, et qu'on pourrait avoir la malchance (même si la probabilité est faible étant donné la taille 10000 des échantillons observés) que ces estimations soient très éloignées des valeurs théoriques.

3. La première colonne de P contient un vecteur propre de A pour la valeur propre 1. À renormalisation près pour que la somme de ses coefficients fasse 1, ce qui d'ailleurs été fait en ligne 2 du script, on obtient la loi stationnaire U de la chaîne de Markov. Rappelons que U satisfait $U = AU$, et que (X_n) converge en loi vers une variable aléatoire X dont la loi est donnée par ce vecteur U :

   $$P(X = 1) = 0.8839779, \quad P(X = 2) = 0.0441989, \quad P(X = 3) = 0.0718232.$$

   On observe d'ailleurs cette convergence sur la loi de X_{100} qui est proche, déjà pour $n = 100$, de la loi stationnaire U.

Exercice 2

1. On peut utiliser le code suivant :

```
function x = exponentielle(lbd)
      u = rand()
      x = -(1/lbd)*log(1-u)
endfunction
```

2. (a) On résout : $p = 1 - e^{-\lambda} \Leftrightarrow \lambda = -\ln(1-p)$. Avec le résultat admis, et en se souvenant que `floor` est la fonction partie entière sur `Scilab`, on propose la fonction suivante :

```
function x = geom(p)
      lbd = -log(1-p)
      x = exponentielle(lbd)
      y = floor(x)+1
endfunction
```

(b) On procède ainsi :

```
x = zeros(1,10000) \\ vecteur de taille 10000 avec que des 0
for k = 1:10000
     x(k) = geom(0.2) \\ remplace la k-ème composante par une simulation
end
```

(c) La commande `mean(x)` calcule la moyenne des composantes du vecteur `x`. Par la loi faible des grands nombres, elle devrait être proche de l'espérance $\frac{1}{0.2} = 5$. C'est bien ce qu'on obtient ici.

De même, `stdev(x)` calcule l'écart-type des composantes du vecteur `x`. Or l'écart-type empirique est un estimateur ponctuel (sans biais et convergent, on l'avait vu en TD) de l'écart-type théorique. En composant par la fonction continue $x \mapsto x^2$, on obtient un estimateur de la variance théorique. La valeur renvoyée devrait donc être proche de la variance théorique $\frac{q}{p^2} = \frac{0.8}{0.04} = 20$. C'est bien là aussi ce qu'on observe.

(d) Rappelons que `[0:29]` est le vecteur de taille 30 contenant les entiers $0, 1, \dots, 29$. Quand on effectue l'instruction `(1-p)^[0:29]`, on obtient le vecteur $[(1-p)^0, (1-p)^1, \dots, (1-p)^{29}]$. Ainsi, `v` contient le vecteur :

$$[p(1-p)^0, p(1-p)^1, \dots, p(1-p)^{29}].$$

Il s'agit des probabilités $P(X = i)$ pour $X \hookrightarrow \mathscr{G}(0.2)$ et $i = 0, \dots, 29$.

(e) On a trié dans `y` l'échantillon par modalités : `y(:,1)` contient donc l'ensemble des modalités de l'échantillon, qui sont des entiers ≥ 1 puisqu'il s'agit de réalisations de la loi $\mathscr{G}(0.2)$, et `y(:,2)` contient l'effectif de chaque modalité. Si on divise par l'effectif de l'échantillon, on obtient donc la fréquence de chaque modalité. C'est ce que donne la commande `y(:,1),y(:,2)/length(x)`.

Le script trace deux diagrammes en bâtons :

- Le premier est obtenu à partir de l'échantillon, avec en abscisse les modalités et en ordonnée les fréquences observées : il s'agit de la loi empirique (obtenue à partir de l'échantillon, et donc de la fonction `exponentielle`) de $\mathscr{G}(0.2)$.
- Le deuxième donne en abscisse le vecteur de 1 à 30 et en ordonnée les probabilités théoriques.

En comparant ces deux diagrammes, on compare donc les fréquences observées avec les probabilités théoriques. Par la loi faible des grands nombres, elles devraient être proches si la simulation respecte bien la distribution d'une loi géométrique. On observe que c'est effectivement bien le cas.

Exercice 3

1. Rappelons que la variable A contient la matrice $\begin{pmatrix} 1 & 4 & 2 \\ 3 & 2 & 8 \end{pmatrix}$. Scilab renvoie un vecteur de taille le nombre de colonne, contenant le minimum sur chacune des colonnes de A.

   Remarque. `'r'` signifie ici row, car le minimum est pris par colonne, suivant les lignes. On a la commande analogue `min(A,'c')` et l'équivalent pour `max`

2. (a) `X` contient une matrice à 7 lignes et 10000 colonnes dont chaque coefficient est une réalisation d'une variable suivant une loi $\mathscr{E}(1)$.
   D'après la question précédente, `I` est donc un vecteur de taille 10000 tel que pour tout $1 \leq i \leq 10000$, `Y(i)` contient le minimum de la i-ème colonne, c'est à dire de 7 réalisations indépendantes de variables qui suivent une loi $\mathscr{E}(1)$. Il s'agit donc d'une réalisation de I_n.
   On dispose donc de 10000 réalisations de I_n, on trace alors l'histogramme des fréquences associé à ces données statistiques, réparties en 100 classes de mêmes amplitudes.
   On trace ensuite sur le même diagramme la densité de la loi $\mathscr{E}(7)$ sur $[-0.1, 1]$.

(b) On remarque que l'aire de chaque rectangle de l'histogramme correspond à l'air sous la densité. Ainsi par comparaison entre l'histogramme des fréquences et la densité de $\mathscr{E}(7)$, on peut conjecturer que I_n suit une loi $\mathscr{E}(7)$.

Exercice 4 (Extrait Problème Ecricome 2016)

1. On a une probabilité de $\frac{x}{x+y}$ de tirer une boule rouge. Donc le programme doit retourner 0 avec une probabilité $\frac{x}{x+y}$. Il faut donc que la condition soit satisfaite avec cette probabilité. On va donc prendre comme condition $r < \dfrac{x}{x+y}$ qui est satisfaite avec une probabilité de $P(R < \frac{x}{x+y}) = P(R \leq \frac{x}{x+y}) = \frac{x}{x+y}$ où $R \hookrightarrow \mathscr{U}([0,1])$.

```
function res = tirage(x,y)
    r = rand()
    if r< x/(x+y) then
       res = 0
    else
       res = 1
    end
endfunction
```

2. Si $r = 0$ alors on ajoute une rouge sinon on ajoute une blanche, d'où `x = x+1` ou `y = y+1` ; le nombre de rouges ajoutées est donc la différence entre le nombre de boules rouges x au final et le nombre de boules rouges a au départ, soit `Xn = x-a`. Ainsi on a :

```
function Xn = experience(a,b,n)
    x = a
    y = b
    for k=1:n
        r = tirage(x,y) \\ r = 0 si rouge au tirage k, r = 1 si blanche
        if r == 0 then
           x = x+1
        else
           y = y+1
        end
    end
    Xn = x-a
endfunction
```

3. Attention, la numérotation des éléments du tableau doivent aller de 1 à $n+1$ alors que les valeurs prises par X_n vont de 0 à n.

```
function loi = simulation(a,b,n,m)
    loi = zeros(1,n+1) \\pour stocker les fréquences
    for k=1:m
        r = experience(a,b,n) \\ valeur de Xn à la k-eme réalisation
        loi(r+1) = loi(r+1)+1 \\ on augmente l'effectif de Xn = r de 1
    end
    loi = loi/m \\on divise par l'effectif total pour obtenir la fréquence
endfunction
```

4. La distribution des fréquence semble équiprobable, donc on peut conjecturer que $X_n \hookrightarrow \mathscr{U}(\llbracket 1, n \rrbracket)$.

Remarque. Ce résultat était ensuite démontré dans la suite du sujet d'Ecricome.
