CPE Lyon 3ETI 2020-2021

Probabilités (M3) : Travaux pratiques Matlab

TP1

Exercice préliminaire

La commande `[h,xout]=hist(X,k)` permet d'ordonner les éléments d'un vecteur `X` en `k` classes.

- `xout` est un vecteur contenant `k` valeurs appelées *classes*, c'est un découpage des valeurs de `X`
- `h` est un vecteur contenant k valeurs, chacune de ces valeurs représentent le nombre de valeurs de X comprises dans chacune des k classes : h est aussi appelé *histogramme* de `X`

L'objectif de cet exercice est de se familiariser avec la commande `hist`. Pour cela on vous demande de :

- Télécharger sur CPe-campus les fichiers Matlab `ExPrelim1.m` et `ExPrelim2.m`
- Exécuter ces programmes
- Commenter et analyser dans le détail les résultats observés

Exercice 1

Est-il plus ou moins avantageux, lorsqu'on joue aux dés, de parier :

- A : sur l'apparition d'au moins un 6 quand on lance 4 fois un dé
- B : sur l'apparition d'au moins un double-six, quand on lance 24 fois deux dés

Le chevalier de Méré (Antoine Gombaud : 1607-1684), qui était un grand joueur, avait remarqué que le premier mode de pari avait une probabilité supérieure à 1/2. Se laissant abuser par un soi-disant argument de proportionnalité, le chevalier considérait que le deuxième mode de pari avait la même probabilité de réussite que (A) en raisonnant ainsi : en lançant un dé, il y a 6 issues; en lançant deux 2 dés, il y en a 36, soit 6 fois plus. Puisqu'il est avantageux de parier sur l'apparition d'au moins un 6 en lançant le dé 4 fois de suite, il doit être avantageux de miser sur l'apparition d'au moins un double-six en lançant deux dés 24 fois de suite. Malheureusement pour le chevalier, les règles des probabilités sont plus complexes, et c'est Blaise Pascal (1623-1662) qui calcula la vraie probabilité de (B), très légèrement inférieure à 1/2 : le deuxième jeu n'est pas avantageux !

1) Ecrire une fonction simulant les résultats de *n* lancers d'un dé non pipé à six faces en utilisant les commandes Matlab `rand` et `floor` (partie entière) ou `ceil` (partie entière +1), sous la forme :

   ```
   function y=LancerDeSixFaces(n)
   ```

   où `n` est le nombre de lancers et `y` est le tableau contenant les résultats des `n` lancers.

   Vérifier la validité de cette fonction (on pourra utiliser les commandes `hist` et `bar` de Matlab et représenter sur un graphique les fréquences d'obtention des différents numéros).

2) Ecrire un programme simulant le jeu (A) et le jeu (B) et vérifier que le premier jeu est plus avantageux que le second, contrairement à l'intuition du chevalier de Méré… On comparera avec les résultats théoriques, et on observera comment les probabilités empiriques évoluent lorsqu'on fait varier le nombre de simulations.

Exercice 2 : méthode de Monte-Carlo

1) Jeu de fléchettes

On lance au hasard des fléchettes sur une cible carrée de côté 1 sur laquelle est représenté un quart de disque de rayon 1 et de centre l'origine. On suppose que l'origine se situe sur le coin en bas à gauche de la cible.

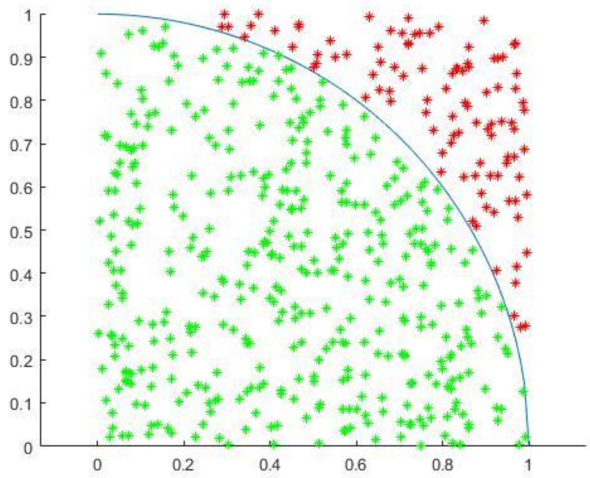
On considère l'évènement A : « la fléchette arrive dans le quart de disque ».

On note $p = P(A)$ la probabilité de l'évènement A et n le nombre de fléchettes lancées.

Concevoir un programme Matlab qui simule le lancer de fléchettes sur la cible et en déduire une approximation de π.

Le programme doit afficher :

- la figure suivante :



- les informations suivantes dans la « command window »:

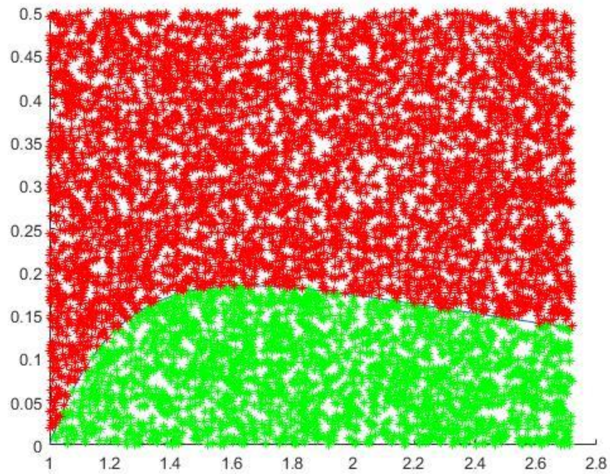
```
APPROXIMATION DE PI (METHODE DE MONTE-CARLO)
Nombre total de points : 500
Nombre de points dans le quadrant : 399
Valeur approximative de pi : 3.192000
```

2) La méthode précédente, basée sur un rapport d'aires, permet de calculer des intégrales. Par exemple, en remplaçant l'arc de cercle par la représentation graphique de la fonction $x \longmapsto \frac{\ln x}{x^2}$, appliquer ce principe au calcul suivant (attention il faudra aussi changer le carré unité par un rectangle judicieusement choisi) :

$$J = \int_1^e \frac{\ln x}{x^2} dx$$

Le programme doit afficher :

- la figure suivante :



- les informations suivantes dans la « command window » :

```
VALEUR APPROXIMATIVE D'UNE INTEGRALE
Nombre total de points : 7000
Nombre de points sous la courbe : 2114
Valeur approximative de l'intégrale : 0.259461
Valeur "exacte" de l'intégrale : ????????
```

3) Faire une estimation de l'intégrale :

$$I = \iint_{\mathcal{D}} \frac{1}{(x+y)^3} dxdy$$

avec :

$$\mathcal{D} = \{(x,y) \in \mathbb{R}^2 \mid 1 \leq x \leq 3, \quad y \geq 2, \quad x+y \leq 5\}$$

a. Calculer « à la main » la valeur exacte de I (on dessinera dans un premier temps le domaine $\mathcal{D}$ puis le volume que représente l'intégrale I).

b. En utilisant la méthode de Monte-Carlo, donner une estimation de I.

<u>Indications</u> :

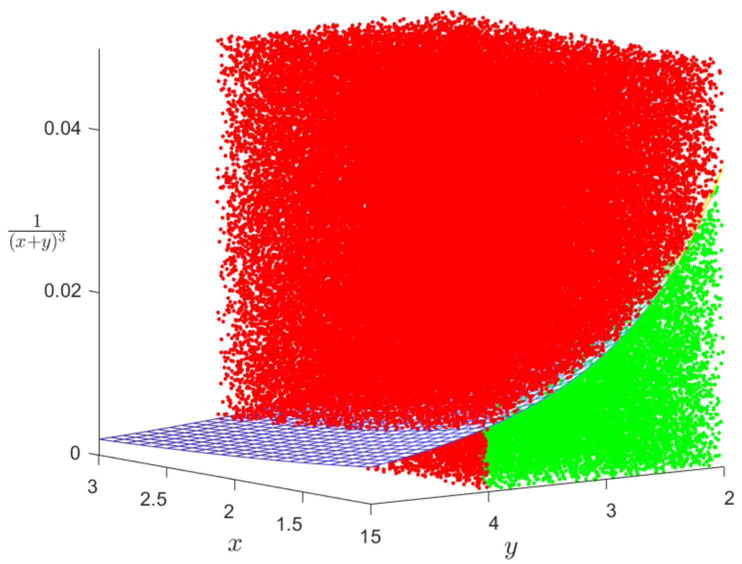
- On prendra $N = 10000$ points
- A l'aide des commandes `meshgrid` et `mesh` on représentera la surface d'équation :

$$(x,y) \mapsto \frac{1}{(x+y)^3}$$

et, comme dans les 2 questions précédentes, on représentera en vert les points en-dessous de la surface et en rouge les points au-dessus de la surface.

Le programme doit afficher :

- la figure suivante :



- les informations suivantes dans la « command window »

```
VALEUR APPROXIMATIVE D'UNE INTEGRALE DOUBLE
Nombre total de points : 100000
Nombre de points dans le quadrant : 13355
Valeur approximative de l'intégrale : 0.026710
Valeur "exacte" de l'intégrale : ?????????
```