

PROBLÈME DU DUC DE TOSCANE

Simulation de lancers de dés avec XCAS

Compétences mathématiques :

- Simulations de lancers de 3 dés. Introduction à la notion de probabilités.

Compétences informatiques :

- Tirages aléatoires avec hasard, ranm. Calculs de moyenne. Boîte à moustache.

Prérequis :

- Boucle « POUR ».

Partie 1: UN PEU DE (PETITE) HISTOIRE...

Cosme II de Médicis (Florence 1590-1621), Duc de Toscane, fut le protecteur de l'illustre Galilée (né à Pise le 15 février 1564 et mort à Florence le 8 janvier 1642) son ancien précepteur. Profitant d'un moment de répit du savant entre l'écriture d'un théorème sur la chute des corps et la création de la lunette astronomique, le Grand Duc lui soumet le problème suivant : il a observé qu'en lançant trois dés cubiques et en faisant la somme des numéros des faces, on obtient plus souvent 10 que 9, alors qu'il y a autant de façons d'obtenir 9 que 10, à savoir six.

Après quelques réflexions, Galilée rédigea un petit mémoire sur les jeux de hasard en 1620 expliquant le phénomène.

Partie 2: SIMULATION DE L'EXPÉRIENCE

Le problème

N'ayant pas la même expérience du jeu que le bon Cosme, nous allons utiliser notre logiciel favori, XCAS, pour simuler un grand nombre de parties et étudier statistiquement les résultats obtenus.

1. Nous utiliserons en premier lieu `hasard(n)` qui renvoie un entier appartenant à l'intervalle $[0 ; n[$. Comment utiliser cette commande pour obtenir le résultat du lancer d'un dé ? De trois dés ?
2. Plutôt que d'appuyer 100 000 fois sur la touche , nous allons utiliser une commande plus pratique, `ranm(1, nombre d'expériences, 'expérience')` qui renvoie une liste de nombre d'expériences résultats de l'expérience.
Par exemple `ranm(1,10000, 'hasard(3)')` renverra 10 000 nombres entiers égaux à 0, 1 ou 2.
Construisez alors une liste T de 100 000 tirages de notre problème ducal.
3. On utilisera également `count_eq(n, liste)` qui compte le nombre d'apparitions de n dans la liste liste.
Comptez maintenant le nombre d'apparitions de 9 et 10 dans T puis renouvelez plusieurs fois l'expérience.

Partie 3: SIMULATION À GRANDE ÉCHELLE

1. Observation

Observez et commentez ce programme :

```
toscane(tailedeliste,nbdelistes):={
local neuf,dix,T,n,d,s,k,mn,md;
neuf=NULL; dix=NULL; //on crée 2 listes vides au départ
pour k de 1 jusque nbdelistes faire //on va créer nbdelistes... listes
  T:=ranm(1,tailedeliste,'hasard(6)+hasard(6)+hasard(6)+3'); // on construit une
  liste de
"tailedeliste" lancers de 3 dés en affichant les résultats
  neuf:=neuf,count_eq(9,T); // on compte le nombre de 9
  dix:=dix,count_eq(10,T); // on compte le nombre de 10
fpour
n:=evalf(moyenne([neuf])/tailedeliste)*100; //on calcule la fréquence des sorties
d:=evalf(moyenne([dix])/tailedeliste)*100;
s:=tailedeliste*nbdelistes;
mn:=couleur(moustache([neuf]),jaune+rempli+line_width_3);
md:=couleur(moustache([dix]),bleu+rempli+line_width_3);
afficher("Sur "+s+" essais, la fréquence de sortie de 9 est de "+n+"%, et celle de 10 est
de "+d+"%");
```

```
mn,md;  
};;
```

On teste :

```
toscano(10000,100)
```

REMARQUE : si vous effectuez plusieurs essais avec différentes valeurs, il peut être utile de régler à nouveau la fenêtre automatiquement en cliquant sur :

Menu->Voir->autoscale

2. Résolution

Comment expliquez ce phénomène ? Comparez par exemple les issues (6,2,2) et (6,3,1) : y a-t-il autant de manières des les obtenir ?