
TP1 : Calcul de probabilités au Yam's

Objectifs

1. Reprendre en mains le python, en découvrant ses fonctions spécifiquement dédiées aux probabilités.
2. Calcul **théorique** de certaines probabilités (dénombrements, etc).
3. Estimation directe de probabilités par la **simulation numérique**.



À récupérer sur l'ENT

- Vous devez déposer le compte rendu du TP sur l'ENT. Pas de dépôt par mail!

Présentation du TP

Au Yam's (aussi appelé Yahtzee), on lance 5 dés simultanément avec l'objectif de réaliser des figures. Voici une liste des figures réalisables :

Nom	Composition
Brelan	Trois dés identiques
Carré	Quatre dés identiques
Yam	Cinq dés identiques
Full	Trois dés identiques + deux dés identiques
Petite suite	1,2,3,4,5
Grande suite	2,3,4,5,6

1. Dans un premier temps, nous considérerons la probabilité d'obtenir ces figures en **un seul lancer**. Les probabilités sont alors assez simples pour être calculées de façon exacte, avec des dénombrements.
2. Cependant, dans le vrai jeu du Yam's, on a droit à **trois lancers** pour réaliser les figures. Dans ce cas, les calculs théoriques deviennent pénibles. On préférera donc une autre approche : **simuler plein de parties avec un ordinateur**, pour estimer les probabilités de manière purement numérique.

Exercice 1 (Prise en main).

Importer la classe `random` de Numpy avec la commande `import numpy.random as npr`.

1. Exécuter la commande `npr.rand()`
2. Exécuter la commande `npr.seed(seed=1)`
3. Exécuter la commande `npr.randint(3,5)`
4. En utilisant l'aide, trouver comment python génère les résultats des commandes précédentes.
5. Générer un tableau binaire de taille 10 (qui ne contient que 0 ou 1)
6. A l'aide de la commande `Counter` de la classe `collections`, calculer les occurrences de 1 et de 0.
7. Écrire une fonction `fct_occ` qui prend un tableau d'entiers en entrée, calcule les occurrences de chacun de ses éléments et rend un tableau qui contient ces occurrences.
8. Générer un tableau aléatoire `tab` de taille 5 qui ne contient que des entiers entre 1 et 6.
9. À l'aide de la fonction `fct_occ` et de `tab`, comment détecteriez-vous
 - Un carré ?
 - Un full ?

- Une petite suite ?

(Une phrase rapide pour chaque question est suffisante – l’implémentation effective se fait à l’exercice suivant.)

Exercice 2 (Probabilités de différentes figures).

Pour répondre à cet exercice, outre votre fonction `fct_occ`, les fonctions numpy `where` et `equal`, voire `sort`, pourront s’avérer pratiques.

1. a) Simulez un grand nombre de tirages à 5 dés (par exemple, 10 000), et comptez le nombre de fois où on obtient un **carré**. Déduisez-en une estimation numérique pour la probabilité de réaliser un carré.
b) Quelle est la probabilité *théorique* de réaliser un carré ? Vérifiez que vous retrouvez (presque) le même nombre.
2. Mêmes questions (estimation numérique + valeur théorique) pour le **full**.
3. Mêmes questions (estimation numérique + valeur théorique) pour le **brelan**.
4. Mêmes questions (estimation numérique + valeur théorique) pour le **yam**.
5. Mêmes questions (estimation numérique + valeur théorique) pour la **petite suite**.

Exercice 3 (Le maximum de 6 en trois lancers).

Dans le vrai jeu, on dispose de 3 lancers pour réaliser les figures de son choix. À chaque nouveau lancer, on peut garder les dés qui nous intéressent du lancer précédent, et relancer uniquement les autres dés.

1. Ecrivez une fonction qui simule un joueur « jouant les 6 » :

```
1 def joue_les_6() :
```

- Après chaque lancer, on met de côté les dés ayant fait 6, et on relance les autres.
- On s’arrête au bout de trois lancers, et on renvoie le nombre final de 6 obtenus, dans la variable `nombre`.

2. Appelez 10 000 fois la fonction `joue_les_6`, et stockez les résultats dans un tableau de taille 10 000. Visualisez l’histogramme de vos simulations (voir la fonction `hist` dans `matplotlib` ou `histplot` dans `seaborn`, par exemple).
3. a) Combien de 6 obtiendrez-vous avec la plus forte probabilité ?
b) Avec quelle probabilité parviendrez-vous à obtenir le Yam de 6 ?

Exercice supplémentaire (facultatif)

Exercice 4 (Le Yam en trois lancers).

On souhaite réaliser un Yam (en trois lancers, et indifféremment du chiffre utilisé). Au premier lancer, on a obtenu une paire de 4, et trois autres dés dépareillés. Au second lancer, vaut-il mieux :

- Garder les deux 4, et relancer les trois autres dés ?
- Entièrement relancer les cinq dés ?

Vous répondrez à cette question par la méthode de votre choix (arguments théoriques et/ou simulations).