

Plaquette 5 – Simuler et résoudre des problèmes

Algorithmique et Python — Première STMG

Corrections détaillées

Boîte à outils

Simuler une épreuve de Bernoulli de paramètre p . *random()* renvoie un nombre au hasard dans $[0; 1[$; la condition *random()* < p est vraie avec la probabilité p .

```
from random import random, randint
```

```
def bernoulli(p):
    if random() < p:
        return 1
    return 0
```

Fréquence dans un échantillon de taille n .

$$f = \frac{\text{nombre de succès}}{n}$$

```
def frequence(n, p):
    s = 0
    for k in range(n):
        s = s + bernoulli(p)
    return s / n
```

Fluctuation. Les fréquences de N échantillons se dispersent autour de p ; leur écart-type s est proportionnel à $\frac{1}{\sqrt{n}}$, et environ 95 % d'entre elles sont dans $[p - 2s; p + 2s]$.

Dé. *randint(1, 6)* renvoie un entier au hasard entre 1 et 6, bornes comprises.

Loi des grands nombres (constat). Sur un grand nombre de simulations, la fréquence d'un événement est proche de sa probabilité.

Correction de l'exercice 1 – Simuler une épreuve de Bernoulli

a) *random()* renvoie un nombre au hasard entre 0 et 1, chaque intervalle étant atteint avec une probabilité égale à sa longueur. La condition *random()* < 0,3 est vraie quand ce nombre tombe dans $[0; 0,3[$, intervalle de longueur 0,3 :

$$P(\text{random}() < 0,3) = 0,3$$

La fonction renvoie alors 1 (succès), sinon 0 (échec) : elle simule une loi de Bernoulli de paramètre 0,3.

b) On décide que 1 signifie « le client achète » et 0 « il n'achète pas » : chaque appel *bernoulli(0.3)* simule un nouveau client.

Réponse. a) *random()* < 0,3 est vrai avec la probabilité 0,3 — b) 1 = achat, 0 = pas d'achat.

Correction de l'exercice 2 – Fréquence dans un échantillon

a) La boucle simule $n = 100$ épreuves de Bernoulli de paramètre 0,3 ; l'accumulateur s compte les succès (chaque succès ajoute 1). La fonction renvoie la **fréquence** des succès :

$$f = \frac{\text{nombre de succès}}{n}$$

b) Les fréquences varient d'un échantillon à l'autre, autour de 0,3 : c'est la **fluctuation d'échantillonnage**. Elles sont ici toutes un peu inférieures à 0,3, ce qui peut arriver : chaque échantillon de 100 clients est différent, et la fréquence observée n'est jamais exactement la probabilité.

Réponse. a) la fréquence des succès sur 100 épreuves — b) fluctuation autour de 0,3.

Correction de l'exercice 3 - Simuler 500 échantillons

a) F est une liste de 500 fréquences, chacune obtenue en simulant un échantillon de 100 épreuves. La compréhension appelle 500 fois la fonction *frequence*.

b) On divise chaque effectif par 500 :

$$\frac{335}{500} = 67 \% \quad \frac{478}{500} = 95,6 \% \quad \frac{500}{500} = 100 \%$$

La moyenne des fréquences, 0,3018, est très proche de $p = 0,3$. Environ deux tiers des échantillons sont à moins d'un écart-type de p , environ 95 % à moins de deux écarts-types, et tous à moins de trois. Une fréquence à plus de $2s$ de p est donc **rare** dans ce modèle.

Réponse. b) 67 %, 95,6 %, 100 %.

Correction de l'exercice 4 - Compter dans un intervalle

Une fréquence f est dans l'intervalle $[p - ks; p + ks]$ lorsque son écart à p ne dépasse pas ks :

$$p - ks \leq f \leq p + ks \Leftrightarrow |f - p| \leq ks$$

On complète avec la valeur absolue :

```
if abs(f - p) <= k * s:
```

ou, sans valeur absolue, *if* $p - k * s \leq f \leq p + k * s$:. La fonction est un **compteur** : *compte(F, 0.3, 0.0467, 2)* renverrait 478 pour la simulation de l'exercice précédent.

Réponse. $\text{abs}(f - p) \leq k * s$.

Correction de l'exercice 5 - Taille de l'échantillon et dispersion

Quand n est multiplié par 4 (de 25 à 100, puis de 100 à 400), l'écart-type est à peu près **divisé par 2** : $0,0903 \approx 2 \times 0,048$ et $0,048 \approx 2 \times 0,0231$.

C'est la propriété annoncée : s est proportionnel à $\frac{1}{\sqrt{n}}$. Or

$$\sqrt{4n} = 2\sqrt{n}$$

donc multiplier n par 4 divise $\frac{1}{\sqrt{n}}$ par 2. Pour diviser la dispersion par k , il faut multiplier la taille de l'échantillon par k^2 .

En pratique, un sondage auprès de 400 personnes est deux fois plus précis qu'un sondage auprès de 100, mais coûte quatre fois plus cher.

Réponse. $n \times 4$ divise s par 2 : s proportionnel à $\frac{1}{\sqrt{n}}$.

Correction de l'exercice 6 – Simuler un dé

`randint(1, 6)` simule un lancer de dé équilibré ; la liste `de` contient 6000 lancers, et `de.count(6)` compte les 6 obtenus.

La fréquence du 6 est

$$\frac{1009}{6000} \approx 0,168$$

La probabilité d'obtenir 6 vaut $\frac{1}{6} \approx 0,167$. La fréquence observée en est très proche : sur un grand nombre de lancers, la fréquence d'un événement se rapproche de sa probabilité. Avec seulement 12 lancers, l'écart aurait pu être bien plus grand.

Réponse. Fréquence environ 0,168, proche de $\frac{1}{6} \approx 0,167$.

Correction de l'exercice 7 – Estimer une probabilité par simulation

a) La fonction lance au plus quatre fois un dé : dès qu'un 6 sort, elle renvoie 1 (le `return` arrête la fonction) ; si aucun 6 ne sort en quatre lancers, elle renvoie 0. Elle simule l'épreuve « obtenir au moins un 6 en quatre lancers ».

b) Le programme affiche la fréquence de cet événement sur 10 000 simulations. La probabilité exacte se calcule par le contraire :

$$1 - \left(\frac{5}{6}\right)^4 \approx 0,518$$

La fréquence simulée, 0,5243, en est proche (écart d'environ 0,007). La simulation confirme que l'événement a un peu plus d'une chance sur deux de se produire.

Réponse. b) 0,5243 contre 0,518 : simulation cohérente.

Correction de l'exercice 8 – Simuler trois clients

`acheteurs()` simule trois clients qui achètent chacun avec la probabilité 0,2 et renvoie le nombre d'acheteurs ; le programme affiche la fréquence de 0, 1, 2 et 3 acheteurs sur 10 000 simulations.

La loi exacte se calcule avec l'arbre (1, 3, 3, 1 chemins) :

$$0,8^3 = 0,512 \quad 3 \times 0,2 \times 0,8^2 = 0,384$$

puis $3 \times 0,2^2 \times 0,8 = 0,096$ et $0,2^3 = 0,008$.

Les fréquences simulées (0,5148 ; 0,3802 ; 0,0961 ; 0,0089) sont toutes à moins de 0,005 des probabilités : la simulation confirme le calcul.

Réponse. Loi exacte 0,512 ; 0,384 ; 0,096 ; 0,008, très proche des fréquences.

Correction de l'exercice 9 – Repérer une erreur de simulation

L'erreur. La condition `random() > 0.1` est vraie quand le nombre tiré est dans $]0,1 ; 1[$, intervalle de longueur 0,9 :

$$P(\text{random}() > 0,1) = 0,9$$

L'élève simule donc un retard avec la probabilité 0,9, d'où ses 90 % de retards.

La correction. Pour un événement de probabilité 0,1, on écrit `if random() < 0.1` : On obtient alors environ 10 % de retards sur un grand nombre de simulations.

On contrôle toujours une simulation en vérifiant que la fréquence obtenue est proche de la probabilité voulue.

Réponse. Il faut $random() < 0.1$.

Correction de l'exercice 10 - Une fonction de seuil réutilisable

On multiplie par q tant que le seuil n'est pas dépassé, en comptant les étapes :

```
def seuil(u0, q, S):
    u = u0
    n = 0
    while u <= S:
        u = u * q
        n = n + 1
    return n
```

$seuil(1200, 1.08, 2000)$ renvoie 7 (un nombre d'abonnés qui augmente de 8 % par mois dépasse 2000 au bout de 7 mois).

$seuil(5000, 1.02, 6000)$ renvoie 10 : un capital de 5000 € placé à 2 % par an dépasse 6000 € au bout de 10 ans, puisque

$$5000 \times 1,02^9 \approx 5975,46 \quad 5000 \times 1,02^{10} \approx 6094,97$$

Réponse. 7 et 10 (le capital dépasse 6000 € au bout de 10 ans).

Correction de l'exercice 11 - Chercher un optimum par balayage

On parcourt les quantités et on garde la meilleure rencontrée :

```
def B(q):
    return -q**2 + 80*q - 1200

meilleur = 0
for q in range(61):
    if B(q) > B(meilleur):
        meilleur = q
print(meilleur, B(meilleur))
```

Le programme affiche **40 400** : le bénéfice est maximal pour 40 articles et vaut 400 €.

On retrouve le résultat du chapitre sur le second degré : $B(q) = -(q - 20)(q - 60)$, et le sommet est au milieu des racines,

$$q = \frac{20 + 60}{2} = 40$$

Réponse. **40 400** : 40 articles, bénéfice de 400 €.

Correction de l'exercice 12 - Estimer l'espérance d'un jeu

Le gain G vaut 8 avec la probabilité $\frac{1}{6}$, 1 avec la probabilité $\frac{2}{6}$ (faces 4 et 5) et -3 avec la probabilité $\frac{3}{6}$ (faces 1, 2, 3) :

$$E(G) = 8 \times \frac{1}{6} + 1 \times \frac{2}{6} - 3 \times \frac{3}{6} = \frac{8 + 2 - 9}{6} = \frac{1}{6} \approx 0,167$$

La moyenne simulée, 0,1309, est proche de l'espérance : sur un grand nombre de parties, le gain moyen se rapproche de $E(G)$. Le jeu est légèrement favorable au joueur, qui gagne en moyenne environ 17 centimes par partie ; l'organisateur doit revoir ses gains s'il veut récolter de l'argent.

Réponse. $E(G) = \frac{1}{6} \approx 0,167$ €, proche de la moyenne simulée 0,1309.

Correction de l'exercice 13 - Découper un programme en fonctions

On découpe la tâche en trois fonctions simples :

```
def retard():
    return random() < 0.1

def commande_en_retard():
    for k in range(4):
        if retard():
            return 1
    return 0

def estimation(N):
    return sum(commande_en_retard() for k in range(N)) / N
```

retard simule un colis, *commande_en_retard* une commande de quatre colis, *estimation(N)* la fréquence sur N commandes.

La probabilité exacte, par le contraire, vaut

$$1 - 0,9^4 = 1 - 0,6561 = 0,3439$$

estimation(10000) doit donc donner une valeur proche de 0,34.

Réponse. Trois fonctions ; probabilité exacte 0,3439.

Correction de l'exercice 14 - Attendre le premier succès

a) La boucle « passe un nouvel appel » tant que l'appel échoue (*random()* $\geq p$, probabilité 0,7), en comptant les appels dans n . Elle s'arrête au premier succès : la fonction renvoie le **numéro de l'appel** qui donne le premier rendez-vous.

b) Le premier rendez-vous au troisième appel correspond à : échec, échec, succès. C'est un seul chemin de l'arbre :

$$0,7 \times 0,7 \times 0,3 = 0,147$$

La fréquence simulée, 0,1419, en est proche. De même, la simulation donne environ 0,30 pour un rendez-vous dès le premier appel et 0,21 au deuxième, conformément aux probabilités 0,3 et $0,7 \times 0,3 = 0,21$.

Réponse. a) le numéro du premier appel réussi — b) 0,147, proche de 0,1419.

Correction de l'exercice 15 - Synthèse : faut-il faire une promotion ?

a) On simule chaque client par une épreuve de Bernoulli et on cumule la marge :

```
def marge_jour(p, m):
    total = 0
    for k in range(100):
        if random() < p:
            total = total + m
    return total
```

b) En moyenne, le nombre d'acheteurs est $100 \times p$, d'où une marge moyenne de

$$100 \times 0,25 \times 12 = 300 \quad 100 \times 0,35 \times 9 = 315$$

c) Les moyennes simulées (301,75 € et 313,05 €) sont proches des espérances (300 € et 315 €). La promotion fait gagner en moyenne environ 15 € de marge par jour : elle est rentable, à condition que la hausse du taux d'achat se confirme sur le terrain.

Réponse. b) 300 € sans promotion, 315 € avec — c) la promotion est rentable.