Capture Marquage Recapture

Partie 1 : Fluctuation d'échantillonnage

On s'intéresse à une très grande collection de billes. On décide d'en marquer 30% avec de la peinture avant de les mélanger avec les autres.
On souhaite maintenant effectuer des prélèvements de taille variable dans notre stock de billes et, dans chacun de ces échantillons, compter la part d'entre elles qui sont marquées avec de la peinture.
Cette part sera-t-elle proche des 30% effectivement marquées dans la population initiale ?

On peut réaliser cette expérience avec de vraies billes. On peut aussi réaliser l'expérience à l'échelle de la classe, avec des feuilles, des stylos, ou le matériel à disposition : essayez vous-mêmes !
Cependant, pour pouvoir réaliser nos expériences sur des échantillons en nombre et en taille variable (et surtout pour économiser du temps et des litres de peinture !), nous allons réaliser une simulation en Python.

Simulation d'un échantillon

On considère un échantillon de taille t, c'est-à-dire composés de t individus. Parmi eux, 30% sont marqués. La proportion p d'individus marqués dans cette population est égale à 0.3.

Pour représenter cette expérience avec une simulation Python, on génère t nombres aléatoires compris entre $[0;1[$ à l'aide de la commande `random`. Tous les nombres inférieurs à 0.3 symbolisent des individus marqués, les autres non. On compte ensuite dans l'échantillon le nombre d'individus marqués, c'est-à-dire le nombre de valeurs inférieures à 0.3.

1. Télécharger le script [1] puis compléter la fonction `echantillon(t)`. On donne ci-dessous la première partie du script :

```
from math import *
from random import random
from matplotlib.pyplot import scatter,show
#L'import de ce module est nécessaire pour la suite du code
def echantillon(t):
  marque= ...
  for i in range(t):
```

1. https ://my.numworks.com/python/elodie-gamot/fluctuation

```
    individu=random()
    if individu< ... :
      marque+=1
  return ...
```

```
from math import *
from random import random
from matplotlib.pyplot import scatter,show
#L'import de ce module est nécessaire pour la suite du code
def echantillon(t):
  marque=0
  for i in range(t):
    individu=random()
    if individu<0.3 :
      marque+=1
  return marque
```

2. Exécuter sur la calculatrice `echantillon(100)` trois fois de suite. Quelles sont les valeurs obtenues ? Quelle interprétation peut-on en faire dans le contexte de notre énoncé ?

   Si le programme retourne par exemple 31, cela signifie que sur un échantillon de 100 billes, 31 étaient marquées.

3. Modifier le programme pour qu'il retourne non plus le nombre de valeurs inférieures à 0.3 mais leur fréquence par rapport à la taille de l'échantillon. Tester plusieurs fois la fonction avec `echantillon(50)` puis `echantillon(500)`. Qu'observe-t-on ?

   Il suffit de remplacer `return marque` par `return marque/t` sur la dernière ligne. Plus l'échantillon est grand, plus la fréquence obtenue est proche de 0.3.

Représentation graphique avec matplotlib

On souhaite maintenant procéder en même temps à l'étude de n échantillons de taille t. Pour cela, nous allons représenter graphiquement les différents résultats obtenus pour observer leur fluctuation avec la fonction `graph(n,t)` présente dans le même script.

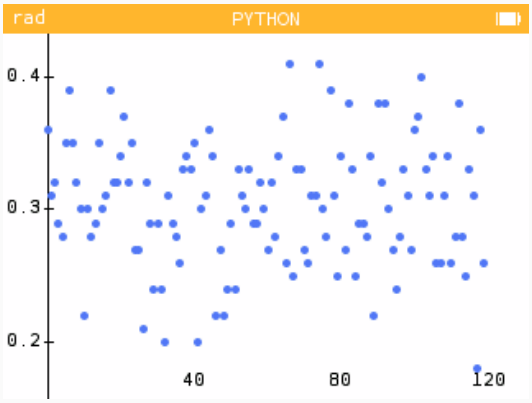
1. Expliquer ce que fait le programme de la fonction `graph(n,t)` sachant que la fonction `scatter(X,Y)` du module `matplotlib.pyplot` permet de représenter un nuage de points utilisant la liste `X` en abscisses et la liste `Y` en ordonnées.
   On ne s'intéresse pas pour le moment aux lignes précédées du symbole `#` et qui ne sont pour le moment pas actives.

```
def graph(n,t):
  x=[k for k in range(n)]
  y=[echantillon(t) for k in x]
  scatter(x,y)
  show()
```

En `x` est stockée une liste de `n` entiers. En `y` sont stockés les résultats successifs de la fonction `echantillon(t)` lancée `n` fois (autant de fois qu'il y a de nombres dans la liste `x`). La fonction va donc permettre de représenter un nuage de `n` points, chacun correspondant à une simulation de taille `t`, répartis sur l'axe des abscisses, et dont l'ordonnée correspond à la fréquence obtenue pour chaque simulation.

2. Lancer `graph(120,100)` plusieurs fois. Autour de quelle valeur se répartissent les fréquences observées dans les échantillons ? Retrouve-t-on les résultats obtenus précédemment ?

Les valeurs se répartissent autour de 0.3, ce qui est effectivement la proportion théorique de billes marquées dans la population initiale. La large majorité des résultats se trouve entre 0.2 et 0.4.



3. On supprime maintenant les symboles `#`, ce qui permet l'activation des deux lignes en fin de programme. Elles vont permettre de calculer la fréquence de résultats compris dans l'intervalle $\left[0.3-\frac{1}{\sqrt{n}};0.3+\frac{1}{\sqrt{n}}\right]$.

```
def graph(n,t):
  x=[k for k in range(n)]
  y=[echantillon(t) for k in x]
  compte=len([k for k in y if k>=0.3-1/sqrt(n) and k<=0.3+1/sqrt(n)])
  print(compte/n)
  scatter(x,y)
  show()
```

Lancer à nouveau `graph(120,100)` plusieurs fois.
Quelle valeur est retournée après l'affichage de la représentation graphique ?

On obtient généralement un résultat supérieur à 0.95.

4. Quelle conclusion peut-on faire à partir de toutes nos observations ?

   Lorsque l'on prélève un échantillon dans une population pour y étudier la fréquence d'un caractère, la fréquence à laquelle on observe ce caractère varie : c'est la **fluctuation d'échantillonnage**. Lorsque la taille de l'échantillon augmente, les fluctuations diminuent et, dans au moins 95% des cas, la fréquence observée est comprise entre $p - \frac{1}{\sqrt{n}}$ et $p + \frac{1}{\sqrt{n}}$.

Partie 2 : Capture Marquage Recapture

On s'intéresse à une méthode de recensement de la population animale appelée couramment « Capture Marquage Recapture » (CMR). Cette technique consiste à capturer une partie de la population étudiée et à en marquer tous les individus.

Par « marquage », on entend n'importe quelle technique permettant d'identifier un animal. On peut par exemple prendre en photographie ses marques naturelles. Il faut seulement s'assurer que ces marques sont uniques et ne peuvent pas être perdues, et s'il s'agit d'un marquage effectué par les chercheurs, elles ne doivent évidemment pas affecter la survie des individus. Les individus marqués sont ensuite relâchés dans leur environnement.

On procède par la suite à la capture d'un nouvel échantillon, dans lequel on recense les individus marqués. En faisant l'hypothèse que la fréquence d'individus marqués dans cet échantillon est proportionnelle au nombre d'individus marqués au sein de la population entière, on espère pouvoir établir un recensement relativement fiable de cette population.

On procède ainsi à la capture et au marquage de 654 perruches dans une zone donnée. Lors de la recapture, on dénombre 289 individus marqués sur un échantillon de 1189 perruches.

1. En suivant le principe de la méthode CMR, à combien peut-on estimer la population totale de perruches dans cette zone ?

   On dénombre 289 individus marqués sur un échantillon de 1189 perruches. En faisant l'hypothèse que cette fréquence observée est proportionnelle au nombre d'individus marqués sur la population totale, on en déduit que $\frac{289}{1189} = \frac{654}{N}$ avec N le nombre de perruches total présent dans la zone. D'où une estimation de $N = 654 \times \frac{1189}{289} = 2691$ perruches au total dans cette zone.

2. Pourquoi s'agit-il d'une estimation et non d'un chiffre exact ?

   Il est impossible de recenser tous les individus d'une population. La méthode CMR permet de faire un calcul sur la base d'un échantillon, or on a vu dans la première partie que lorsque l'on observe la fréquence d'un caractère dans un échantillon, les résultats sont fluctuants. Le nombre de perruches

dans le milieu a été calculé à partir d'une fréquence, fluctuante, il s'agit donc d'une estimation.

3. On appelle **intervalle de confiance** l'intervalle dans lequel se trouve une proportion p avec une certitude de 95%.
   Cet intervalle se calcule $\left[f - \frac{1}{\sqrt{n}}; f + \frac{1}{\sqrt{n}}\right]$ avec f la fréquence du caractère observée dans un échantillon de taille n.

   Dans notre exemple, quelle fréquence de perruches marquées observe-t-on lors de la recapture ? En déduire l'intervalle dans lequel se trouve la proportion p (à 10^{-3} près).

   Lors de la recapture, la fréquence de perruches marquées est égale à 0,243 dans un échantillon de taille 1189. On obtient un intervalle de confiance égal à $[0,214; 0,272]$.

4. En déduire un intervalle dans lequel se trouve la population totale de perruches.

   D'après notre intervalle de confiance, la proportion minimale d'individus marqués est égale à 0,214. Cette proportion représente le nombre d'individus marqués sur la taille totale de la poulation, soit $\frac{654}{N}$. On en déduit qu'au maximum, la population compte un nombre d'individus égal à :

   $$N = \frac{654}{0,214} = 3056$$

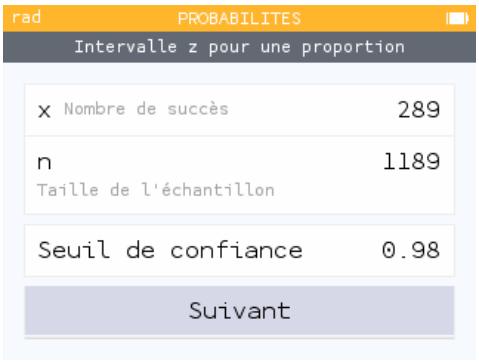
   A l'inverse, la proportion maximale d'individus marqués est à égale à 0,272 ce qui équivaut à une population de 2404 individus.
   Donc, la population totale dans cette zone compte entre 2404 et 3056 perruches.

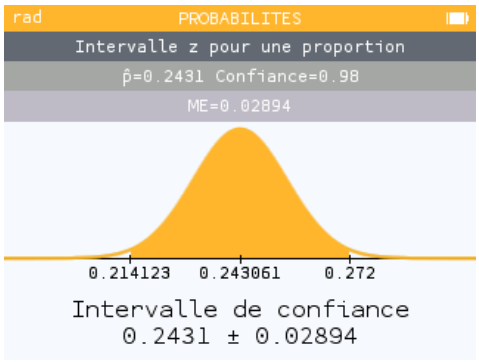
5. L'intervalle obtenu par cette méthode doit cependant lui aussi être envisagé avec beaucoup de précautions. Selon vous, quelles sont les limites de la méthode CMR ?

   Il faut choisir soigneusement le temps écoulé entre les deux captures afin d'éviter de gros écarts dans la population (migration, fort taux de mortalité, ...) et à l'inverse, laisser le temps aux individus marqués de se mélanger à la population totale sans pour autant quitter la zone étudiée.
   Cette méthode fonctionne donc plutôt bien dans des milieux fermés. On peut aussi s'interroger sur les méthodes de marquage qui, lorsqu'elles ne sont pas naturelles, peuvent induire un stress sur les populations étudiées.

Astuce : Il est possible de calculer un intervalle de confiance avec la calculatrice. Attention toutefois, car la formule utilisée pour ce calcul n'est pas au programme du lycée et aboutit à un résultat beaucoup plus précis ! Cela permet toutefois de vérifier la cohérence des calculs.
Il suffit, dans l'application `Probabilités`, de sélectionner le menu `Intervalles > Une proportion`.

Dans notre exemple, le nombre de succès correspond au nombre de perruches marquées, soit 289, pour un échantillon de taille 1189. Par défaut, on peut laisser le seuil de confiance à 95% mais il est possible d'augmenter ce seuil de confiance pour obtenir un résultat moins précis. Ici, un seuil de confiance de 98% nous donne un résultat proche de celui que nous avons calculé.
Une fois les données entrées, il suffit ensuite de passer aux écrans suivants pour aboutir à l'intervalle de confiance.


rad
PROBABILITES
Intervalle z pour une proportion
x Nombre de succès
289
n
1189
Taille de l'échantillon
Seuil de confiance
0.98
Suivant


rad
PROBABILITES
Intervalle z pour une proportion
p̂=0.2431 Confiance=0.98
ME=0.02894
0.214123
0.243061
0.272
Intervalle de confiance
0.2431 ± 0.02894