Lycée Thiers

Informatique
2BCPST1

TP : Feuille d'Exercices 4

Loi Hypergéométrique

Rappel : Une urne contient un nombre N de boules blanches et noires, dont une proportion p de boules blanches (soit Np boules blanches et $Nq = N - Np$ boules noires).
On effectue un tir sans remise de n boules dans l'urne. Soit X la v.a.r. égale au nombre de boules blanches obtenues. La v.a.r. X suit **la loi hypergéométrique** $\mathscr{H}(p, n, N)$.
Son univers image est $X(\Omega) = [[\max(0, n - Nq), \min(Np, n)]] \subset [[0, n]]$,
et pour tout $k \in X(\Omega)$:

$$\mathbb{P}(X = k) = \frac{\binom{Np}{k}\binom{Nq}{n-k}}{\binom{N}{n}}$$

Exercice 1. *Simulation d'un tir dans une urne sans remise.*
Pour simuler la loi hypergéométrique $\mathscr{H}(p, n, N)$ on va simuler l'urne par une liste contenant Np boules blanches, représentées par des 1, et Nq boules noires, représentées par des 0 :
Créer l'urne puis la mélanger à l'aide de la fonction `shuffle()` :

```
Np = int(N*p)
Nq = N - Np
Urne = [1] * Np + [0] * Nq     # Urne = [1, ..., 1, 0, ..., 0]
from random import shuffle
shuffle(Urne)    # mélange la liste aléatoirement
```

(1) Ecrire une fonction `hypergeometrique(p,n,N)` qui simule un tir aléatoire sans remise de n boules dans une urne contenant N boules dont une proportion p de blanches, et qui retourne le nombre de boules blanches obtenues.
On pourra utiliser la fonction `randint(0,m)` du module `random` (après l'avoir importé) qui retourne un entier aléatoire entre 0 et m (inclus) et la méthode `pop()` des listes.
(2) Ecrire une fonction `frequence(p,n,N,f=10000)` qui appelle `f` fois la fonction `hypergeometrique(p,n,N)` et qui retourne la liste des fréquences d'obtention de 0, 1, ..., n boules blanches durant le tir.
(3) Tracer sur une figure l'histogramme des fréquences obtenues. On prendra `N=100`, `n=30`, `p=0.45` et pour `f=10000`.
(4) Tracer sur une nouvelle figure l'histogramme de la fonction de répartition.