Simulation de l'aléatoire

1 L'aléatoire et l'informatique

1.1 Prérequis Mathématique

Une Variable aléatoire réelle est une application X définie sur l'univers de probabilité Ω à valeur dans $\mathbb{R}$.

Elle est entièrement déterminer par

- L'univers image $X(\Omega)$
- La loi de X, c'est-à-dire $\mathbb{P}(X = x_k) = p_k$ pour tout $x_k \in X(\Omega)$

  Cette loi est soit donnée par une formule pour tout x_k, soit un tableau récapitulatif des probabilités.

Trois notions sont importantes :

- La fonction de répartition : $F_X(x) = \mathbb{P}(X \leq x)$

  Dans le cas des variables sur un ensemble fini ou dénombrable, la fonction de répartition est une fonction constante par morceaux sur $\mathbb{R}$ tel que $\lim_{x\to-\infty} F_X(x) = 0$ et $\lim_{x\to+\infty} F_X(x) = 1$.

- L'espérance : $E(X) = \sum_{x_k \in X(\Omega)} x_k \mathbb{P}(X = x_k)$ correspondant à la valeur moyenne de X.

- La variance : $V(X)$ qui correspond à la moyenne des distances au carré entre $E(X)$ et les valeurs de X.

  C'est la dispersion de la variable X par rapport à sa moyenne.

  On la calcule par la formule de Koenig-Huygens : $V(X) = E(X^2) - E(X)^2$, avec $E(X^2) = \sum_{x_k \in X(\Omega)} x_k^2 \mathbb{P}(X = x_k)$

1.2 Pseudo-Aléa

- L'"Expérience aléatoire" : Se dit d'une expérience dont on ne peut prévoir le résultat.
- Les "Machines déterministes" : Les ordinateurs que nous utilisons sont des machines déterministes. Cela signifie que les algorithmes que nous écrivons sont régis par la règle suivante : pour une même entrée, l'algorithme produit toujours la même sortie.

La notion de phénomène aléatoire est incompatible avec la prédictibilité des résultats issue du déterministe.

On introduit alors la notion d'aléatoire *prédictible* ! on utlise les générateurs pseudo-aléatoires. C'est une suite récurrente (u_n) telle que $u_0 = s$ et $u_{n+1} = f(u_n)$ avec f une fonction booléenne. s est la "seed" de la suite pseudo aléatoire.

1.3 En Python

Les commandes pour générer de l'aléatoire en python utilise la librairie `random`.

Plusieurs commandes :

- `random()` : retourne un nombre aléatoire entre 0 et 1 de manière uniforme.
- `seed(S)` : initialise la suite pseudo-aléatoire à une valeur précise (pour refaire l'expérience de manière identique)
- `randomint(a,b)` : retourne un nombre entier aléatoire entre a (inclue) et b (exclue)

2 Simulation d'une V.A.R.

2.1 La Variable de Bernouilli

Une variable de Bernouilli X de paramètre $p \in]0;1[$ se traduit par $X(\Omega) = \{0,1\}$ et pour loi le tableau :

x_k	0	1
$\mathbb{P}(X = x_k)$	$1-p$	p

Pour simuler cette variable aléatoire, on va choisir découper l'intervalle $[0;1[$ et $[0;p[\cup[p;1[$.

Si on choisit un nombre aléatoire dans $[0;1[$ de manière uniforme, il sera dans $[0;p[$ avec une probabilité de p.

Ainsi `random()<0` renvoie `True` avec une probabilité de p et `False` avec une probabilité de $1-p$.

2.2 Une variable définie à l'aide de sa loi

On va simuler les V.A.R. à l'aide des fonctions de répartition. En effet, on sait choisir un nombre entre $[0;1]$ aléatoirement. De plus cette fonction découpe l'intervalle $[0;1]$ en de petits intervalles de taille $\mathbb{P}(X = x_i)$ pour un $x_i \in X(\Omega)$.

2.2.1 Pour une loi finie avec peu de valeurs

On peut résumer la fonction de répartition à ses changements de valeurs c'est-à-dire au tableau des probabilités cumulées ($\mathbb{P}(X \leq x_k)$).

Chaque valeur du tableau nous donne un test à effectuer.

2.2.2 Une variable définie à l'aide d'une loi

Pour cette section, on ne connaît que la formule de la loi de X.

On va aussi introduire les probabilités cumulées $S_k = \mathbb{P}(X \leq x_k)$ qui est définie par $\begin{cases} S_0 = \mathbb{P}(X = x_0) \\ S_{k+1} = S_k + \mathbb{P}(X = x_{k+1}) \end{cases}$

Ainsi, on choisira un nombre aléatoire r entre 0 et 1. Il suffira de déterminer le premier indice k tel que $S_k > r$, alors la valeur de X peut être considéré comme x_k.

2.3 Une variable définie par une expérience

Certaines variables aléatoires ont une interprétation physique précise, comme le lancer d'un dé, la variable binomiale ou la variable géométrique. On va s'intéresser uniquement à ces deux dernières.

La loi binomiale $X \hookrightarrow \mathcal{B}(n,p)$

Elle mesure de nombre de succès de n expériences identiques et indépendantes avec p la probabilité de succès.

La simulation revient à effectuer n expériences aléatoires de Bernouilli et d'utiliser un compteur pour le nombre de succès

La loi géométrique $X \hookrightarrow \mathcal{G}(p)$

Elle mesure le temps d'arrivée du premier succès avec p la probabilité de succès

Il suffit donc d'effectuer une boucle while dont la condition est la négation d'une loi de Bernouilli de paramètre p.

3 Algorithme de probabilité

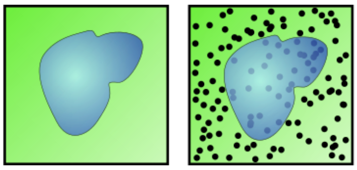
3.1 Calcul de l'espérance et la variance

Si X_i sont des simulations de l'expérience aléatoire, la loi des grands nombres nous donne alors

$$\lim_{n \to +\infty} \frac{1}{n} \sum_{i=1}^{n} X_i = E(X) \text{ et } \lim_{n \to +\infty} \frac{1}{n} \sum_{i=1}^{n} X_i^2 = E(X^2)$$

3.2 Algorithme de Monte-Carlo

On choisit un point au hasard dans une zone et on note X la V.A.R qui vaut 1 si on est dans la zone souhaitée, 0 sinon



$$E(X) = \frac{\mathcal{A}_{zone}}{\mathcal{A}_{totale}}$$