TP Génération de nombres pseudo-aléatoires suivant différentes distributions

Exercice pratique :

Avant de simuler une distribution non uniformes, tester un petit code C générant des nombres uniformément répartis entre A et B.

Introduction

Les techniques exposées précédemment nous permettent de générer des nombres pseudo-aléatoires uniformément distribués entre 0 et 1. Ces techniques sont à la base de tout modèle stochastique, cependant il est nécessaire de pouvoir simuler des lois de distribution réelles (non uniformes comme la distribution des graines autour d'un arbre, de boutures d'algues,...). En effet, lors de la modélisation de systèmes naturels, les données en entrée d'un modèle de simulation doivent être étudiées et validées avec soin. Nous avons déjà vu l'importance d'un ajustement des données recueillies sur le terrain avec une loi qui servira de modèle théorique (chapitre II). Même si cet ajustement des données n'est pas toujours possible, il est néanmoins possible de reproduire les mesures réelles à l'aide de fonctions de distributions discrètes (histogrammes).

Reproduction de distributions discrètes

Imaginons un terrain sur lequel les prélèvements déterminent la présence de 3 espèces différentes. La distribution constatée pourrait être la suivante : 50% pour l'espèce A, 10% pour l'espèce B et 40% pour l'espèce C. Si l'on sélectionne un individu au hasard sur le terrain (tirage supposé uniforme sur 2 dimensions), la probabilité que l'individu soit de l'espèce B est de 0.1. Ce type de procédé peut être utilisé pour créer des terrains virtuels présentant des caractéristiques réalistes (du moins en ce qui concerne les proportions d'individus de chaque espèce), mais d'une manière générale, il est possible de générer toute variable aléatoirement distribuée suivant un histogramme. La technique consiste à interpréter la variable aléatoire X (ici l'espèce d'un individu) comme une variable continue et non discrète. Ainsi, lors du tirage d'une probabilité avec un générateur uniforme entre [0,1[, si la probabilité est comprise entre 0 et 0.5 (strictement) l'espèce sera A, si la probabilité est comprise entre 0.5 et 0.6 (strictement), l'espèce sera B, enfin si la probabilité est supérieure ou égale à 0.6, l'espèce sera C. La figure VI.13 présente l'histogramme correspondant ainsi qu'une fonction de répartition générale. Dans la pratique, on constitue une table contenant les probabilités cumulées (celles qui correspondent aux ordonnées de la fonction de répartition). L'utilisation de la table correspond donc à l'équation (7), si le nombre tiré (uniformément entre 0 et 1) est inférieur à la somme des probabilités de « k à i » et supérieur à la somme de ces probabilité de « k à i - 1 », alors la variable aléatoire peut prendre la valeur x_i. Ainsi, il est possible de s'aiguiller vers différents choix mais surtout, on peut tirer un nombre suivant une loi dont on connaît l'histogramme cumulé (tabulation de la fonction de répartition associée). Cette technique doit être en général réservée au cas de variables aléatoires discrètes. Si elle est appliquée à un variable aléatoire continue, la précision sera bien sûr dépendante du nombre de valeurs de la table (qui correspond au pas d'intégration).

$$X = x_i \quad \text{si l'on a :} \quad \sum_{k=0}^{i-1} p_k \leq NombreTiré < \sum_{k=0}^{i} p_k \qquad (7)$$

Le cas particulier d'un histogramme à deux valeurs correspond à un simple aiguillage et ne nécessite pas la constitution d'une table. Par exemple, si l'on sait qu'en l'absence de motivation (par une nourriture ou autre,...), les chèvres se déplacent d'abord tout droit dans 80% (p = 0.8) des cas et à gauche ou à droite uniquement dans 20% des cas, il suffit de comparer le nombre pseudo-aléatoire tiré à p (ou à 1-p) pour déterminer son choix.

Exercice pratique :

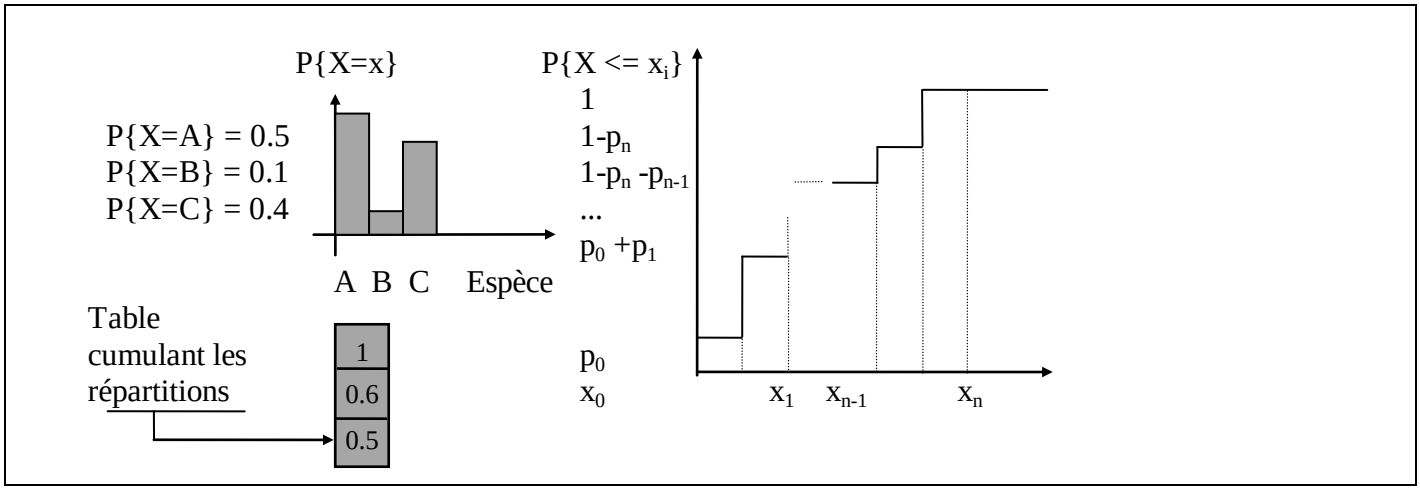
Simuler une distribution suivant l'histogramme de la figure suivante.

Effectuer 1000 tirages suivant cet histogramme et cumuler le nombre d'individus de chaque espèce (A, B ou C) dans trois variables.
Si le générateur utilisé pour la simulation est uniforme entre 0 et 1, vous aurez une répartition proportionnelle à celle donnée dans l'histogramme.

Coder une fonction plus générale en fonction d'un tableau donnant les effectifs par classe. Construire un histogramme de test avec les différentes classes statistique (rester en mode texte), puis initialiser une table cumulant les répartitions, puis normalisez les entre 0 et 1 en fonction de l'histogramme donné et du cumul des effectifs par classe (cf. cours).

Vérifier que la probabilité d'appartenance à chaque classe est respectée sur 1000 événements.

NB: Les tests peuvent être réalisés avec le générateur de base rand ou avec la version C de Mersenne Twister.



Exemple d'histogramme et de fonction de répartition par espèce.

Reproductions de distributions continues

Par analogie avec les distributions discrètes, une technique de génération de nombres pseudo-aléatoires suivant une loi continue est appelée *génération par loi inverse* (ou technique de l'*anamorphose*). Lorsque l'on tire un nombre pseudo-aléatoire, il est placé sur l'axe des ordonnées ce qui permet de déterminer en abscisse la classe de la variable aléatoire X (x_0 à x_n) distribuée suivant la fonction de répartition F. Ainsi nous avons :

$$\text{Nombre tiré} = F(x) \quad => \quad x = F^{-1}(\text{Nombre tiré})$$

Pour obtenir une variable aléatoire X possédant une fonction de distribution F, il suffit de générer un nombre pseudo-aléatoire (uniforme entre 0 et 1) et d'utiliser la forme explicite de la fonction inverse de la fonction de répartition F. Cette technique, bien que non généralisable, s'applique à de nombreuses lois de distribution (loi exponentielle, loi uniforme, loi de Weibull,...) (figure VI.14).

Loi uniforme entre A et B : $x = F^{-1}(\text{Nombre tiré}) = A+(B-A) * \text{Nombre tiré}$

$\text{Moyenne} = (B + A) / 2$ $\text{Variance} = 1/12 * (B - A)^2$

Loi exponentielle (Moyenne) : $x = F^{-1}(\text{Nombre tiré}) = - \text{Moyenne} * \text{Log}(1- \text{Nombre tiré})$

Moyenne = M	Variance = M^2

Fonctions inverses pour la loi uniforme et la loi exponentielle.

Exercice pratique :

Simuler 1000 tirages suivant une distribution exponentielle négative de moyenne 10.

Tester la répartition des nombres sur un petit histogramme (en mode texte ce sera suffisant)

Prenons l'exemple d'une distribution exponentielle (couramment appelée exponentielle négative), souvent utilisée pour modéliser un « grand désordre » (émission de particules, de pannes...). Si un ensemble de données suit une loi exponentielle négative de moyenne : M (et de variance M^2) la fonction de distribution est donnée par l'équation (8) qui conduit à la loi inverse de la figure VI.15 par l'équation (9).

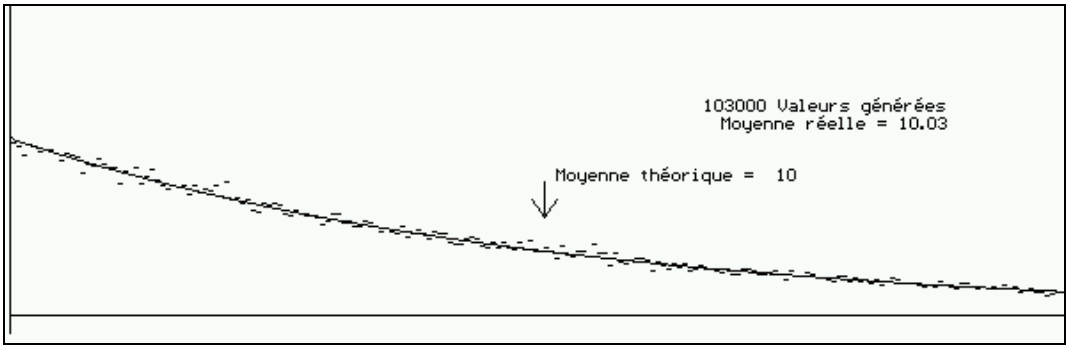
$$F(x) = \int_0^x \frac{1}{M} e^{-\frac{1}{M}z} dz = 1 - e^{-\frac{1}{M}x} \qquad (8)$$

$$NombreTiré = 1 - e^{-\frac{1}{M}x}$$

$$\Rightarrow 1 - NombreTiré = e^{-\frac{1}{M}x} \qquad (9)$$

$$\Rightarrow \ln(1 - NombreTiré) = -\frac{1}{M}x$$

$$\Rightarrow x = -M \ln(1 - NombreTiré)$$



Génération de nombres suivant un loi exponentielle négative de moyenne 10

Simuler des lois de distribution non inversibles

Cette technique s'inspire des techniques stochastiques de calculs d'intégrales (méthodes de Monte-Carlo). Elle se révèle dans la pratique très efficace si la fonction de densité de la loi que l'on souhaite reproduire peut être étudiée dans un intervalle [MinX, MaxX] et reste bornée par une ordonnée maximum MaxY. Il est possible de générer des nombres distribués selon cette loi. Cette technique est connue sous le nom de « méthode de rejection » (Gordon, 1978), et l'algorithme associé est présenté sur la figure VI.16.

```
(1)  Générer deux nombres pseudo-aléatoires Na1 et Na2

(2)  Calculer X = MinX + Na1 * (MaxX - MinX)

(3)  Calculer Y = MaxY * Na2
```

```
(4)  Si Y est inférieur ou égal à f(X)
     Alors X est considéré comme étant distribué
           suivant la loi qui a comme densité f(x)
     Sinon Rejeter X et tirer à nouveau deux nombres en (1)
     FinSi
```

Algorithme général de la méthode de réjection.

Le cas particulier d'une densité gaussienne (loi Normale)

La première technique mise en œuvre pour générer une gaussienne était basée sur le fameux théorème de la Limite Centrale, qui stipule que « *la somme de n variables indépendantes de même loi, de moyenne M et de variance V converge vers la loi normale de moyenne nM et de variance nV lorsque n croît* ». La densité de la loi normale standard (centrée et réduite) notée $N(0,1)$ pour une moyenne égale à zéro et une variance de 1 est donnée par l'équation (10).

$$p(x) = \frac{1}{\sqrt{(2\pi)}} e^{-\frac{x^2}{2}} \qquad (10)$$

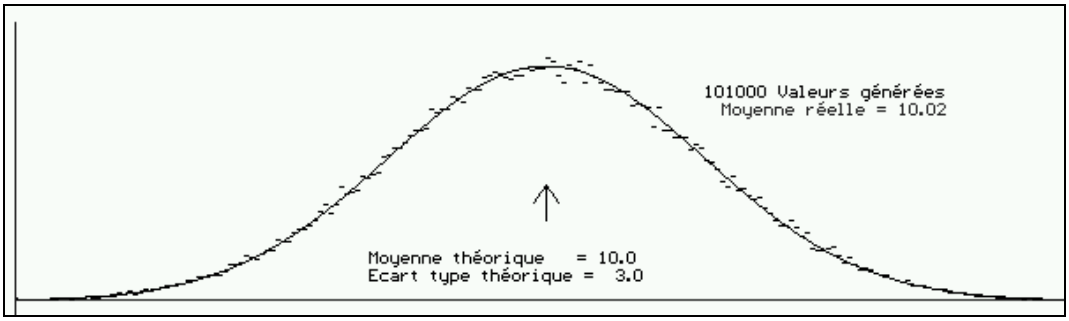
La technique basée sur le théorème de la Limite Centrale peut donc être implémentée par l'équation (11), qui utilise n tirages pseudo-aléatoires (nombres aléatoires Na_1 à Na_n) suivant une loi uniforme. L'équation (12) est obtenue en prenant en compte la moyenne et la variance des nombres tirés (cf. moyenne et variance d'une loi uniforme, Figure VI.17). Une transformation linéaire de la loi normale $N(0,1)$ permet d'obtenir une loi normale de moyenne et de variance quelconque $N(\mu,\sigma)$ en multipliant le résultat de l'équation (12) par l'écart-type souhaité et en additionnant la moyenne (équation (13)).

$$x = \frac{(Na_1 + Na_2 + ... + Na_n - Moyenne(Na))}{\frac{ecart - type(Na)}{\sqrt{n}}} \qquad (11)$$

$$x = \frac{(Na_1 + Na_2 + ... + Na_n - \frac{n}{2})}{\sqrt{\frac{n}{12}}} \qquad (12)$$

$$x = (Na_1 + Na_2 + ... + Na_n - \frac{n}{2})\sqrt{\frac{12}{n}} ecart - type + moyenne \qquad (13)$$

Dans la pratique, n est fixé à 12, d'une part afin d'obtenir des résultats statistiques raisonnables et d'autre part pour avoir le dénominateur de l'équation (12) égal à 1. Comme on l'aura constaté, cette technique est une méthode approchée, dépendante du nombre de tirages que l'on effectue, lente car gourmande en nombre de tirages, et qui, de plus, ne possède pas un grande précision.



Génération d'un loi normale, logiciel ALEATEST

En 1958, Box et Muller présentent une méthode exacte qui s'affranchit du théorème de la Limite Centrale et qui ne nécessite que deux tirages de nombres pseudo-aléatoires pour obtenir deux tirages indépendants distribués suivant la loi normale centrée et réduite. L'inconvénient majeur de cette technique réside dans l'utilisation du calcul d'un sinus et d'un cosinus qui reste lent (malgré l'intégration de ces fonctions dans les microprocesseurs actuels). L'équation (14) exprime la relation entre les nombres issus du générateur et ceux qui suivent une loi $N(0,1)$.

$$x_1 = \cos(2\pi N a_2)(-2\ln(Na_1))^{\frac{1}{2}}$$
$$x_2 = \sin(2\pi N a_2)(-2\ln(Na_1))^{\frac{1}{2}} \qquad (14)$$

Penser à tirer 2 nombres pour avoir les valeurs de chaque côté de la moyennes. Vérifier à l'aide d'un histogramme que vous générez bien une gaussienne. La méthode qui reste la plus usitée pour générer une loi normale reste la méthode de rejection découlant des travaux de Von Neumann et présentée dans le paragraphe précédent. Dans le cas d'une loi normale, il faut cependant considérer un intervalle borné de la fonction de densité associée (toutes les valeurs de la fonction de densité doivent être inférieures à un maximum sur cet intervalle). Deux algorithmes de la méthode de rejection adaptés pour obtenir des nombres distribués suivant une loi $N(\mu,\sigma)$ sont présentés en figure VI.18.

Exercice pratique :

Simuler 1000 tirages suivant une loi normale de moyenne 10 et d'Ecart-type 3 avec la technique de Box et Muller – tester avec un petit histogramme.

Recherche des algorithmes en C de simulation d'une gaussienne avec la méthode de réjection correspondante. Simuler 1000 tirages suivant une distribution gaussienne de moyenne 10 d'écart-type 3 en utilisant ces méthodes. Tester la répartition des nombres sur un histogramme (mode texte ou Excel sera bien suffisant)

Rechercher quelques grandes bibliothèques de génération de nombres pseudo-aléatoires suivant des lois de distribution non uniforme (en C, Java et C++)