Méthodes de Monté Carlo

Prof. Mohamed El Merouani

https://elmerouani.jimdo.com

e-mail: m_merouani@yahoo.fr

Contenu :

- Nombres aléatoires
- Simulation de variables aléatoires
- Simulation des processus stochastiques
- Méthodes de Monté Carlo
- Méthodes de quasi-Monte-Carlo
- Schémas numériques de discrétisation d’équations différentielles stochastiques: Euler, Milstein.
- Applications en finance: pricing de produits dérivés…

2

Bibliographie :

- B. bercu & D. chafaï, :« Modélisation Stochastique et Simulation », Dunod, 2007.
- James A. Payne: «Introduction to Simulation; Programming Techniques and Methods of Analysis », Mc Graw Hill, 1988.
- Reuven Y. Rubinstein:«Simulation and the Monte Carlo Method », John Wiley & Sons, 1981.

3

Bibliographie :

- Stewart V. Hoover/ Ronald F. Perry: «Simulation: A Problem-Solving Approach», Adisson-Wesley, 1989.
- Averill M. Law & W. David Kelton: Simulation Modeling and Analysis », Mc Graw Hill, 1982, 1991 (deux livres différents)

4

Générations des nombres aléatoires

Plan du chapitre

- Définition de la simulation
- Nombres aléatoires
- Générateur de milieu de carrée de Von Neumann
- Générateur (linéaires) de congruence
- Génération de nombre aléatoires sous R

Définition de la Simulation:

- Les méthodes de simulation consistent à faire intervenir le hasard dans un ou plusieurs phénomènes impliqués dans un problème à résoudre.
- La simulation permet d'appréhender un phénomène aléatoire en construisant un échantillon artificiel de sa distribution de probabilité.

7

Définition de la Simulation:

- Les méthodes de la simulation aléatoire nous permettent de construire à l'aide des ordinateurs, un échantillon $(x_1, x_2, \dots, x_n)$ à partir de n'importe quelle loi de probabilité donnée.
- Nous verrons dans ce cours quelques simples algorithmes de simulation des lois de probabilités usuelles.

8

Nombres aléatoires:

- Une réalisation d'une suite de nombres au hasard (aléatoires) indépendants et uniformément distribués sur le segment [0, 1]:

$$u_1, u_2, \ldots$$

est toujours utilisée pour obtenir une réalisation d'une suite de variables aléatoires indépendantes et arbitrairement distribuées.

9

Nombres aléatoires:

- Les tables de nombres au hasard
- Ces tables sont telles que la suite des nombres qui y figurent est assimilable à la réalisation de tirages avec remise dans une urne à dix catégories de boules figurant à proportions égales.
- L'ordinateur fabrique des nombres au hasard qui sont « pseudo-aléatoire ».

10

9. Random Numbers Uniformly Distributed on the Interval [0, 1]

0.5916	4448	9019	8861	5489	2645
6562	7042	2267	3584	7449	7919
3127	5978	4955	6484	3687	9664
0690	7147	8562	5270	2378	5050
3617	3406	1790	7880	3741	5470
7128	7463	4161	6512	2786	1255
6635	1413	5412	7803	2196	8954
9162	0120	7403	1889	2139	2383
9313	6700	6079	2748	9953	8382
8216	6396	8203	2558	7441	8546
9470	5477	9711	1932	4728	1479
3361	3901	3993	9279	5673	1653
5303	5889	5940	2154	1907	2311
1039	8851	6787	5033	5986	2198
9031	2099	9121	4101	0833	4776
9481	6387	7480	1643	4049	5132
8922	5501	0399	6253	9202	6570
5725	9430	4184	3136	8549	1019
4398	4215	1992	9629	4386	8382
3652	4474	3374	3147	9362	4387
9612	6145	1197	4513	6122	0115
6836	6542	4237	6319	0193	2943
4601	7198	6838	2226	1987	5803
9154	5889	8315	0471	5314	1904
2212	0448	5759	0836	5825	7005
7158	0101	8865	4963	4135	1693
9192	8247	2141	5307	3821	7586
3072	7397	5643	4188	1094	6005
2380	8036	6883	8764	2625	0596
3290	2036	9632	1752	6213	5809

11

Nombres aléatoires:

- Disposer d’un bon générateur de nombres aléatoires est fondamental en Simulation.
- En fait, il constitue aussi un élément essentiel en Informatique (algorithmes probabilisés, vérification d’algorithmes, complexité algorithmiques, cryptographie,…) en Statistique (méthodes d’échantillonnage, tests Monte Carlo, Inférence Bayesiene,…) et en général en n’importe quel problème dû au hasard…

12

Nombres aléatoires:

- Les méthodes de simulation sont souvent utiles, dans le contexte des mathématiques financières, car elle permettent de calculer le prix de n'importe quelle option pour peu que l'on sache l'exprimer forme de l'espérance d'une v.a. que l'on sait simuler.
- Dans ce cas, la méthode de Monté Carlo permet, alors, d'écrire très rapidement un algorithme permettant l'évaluation de cette option.
- Lorsqu'on ne sait pas expliciter le prix de l'option analytiquement, ou lorsque la stratégie de gestion de portefeuille est inaccessible analytiquement, la simulation est alors nécessaire.

13

Nombres aléatoires:

- Si l'expérience réelle coute chère
  $\Longrightarrow$ simulation (aléatoire).
- Une propriété souhaitable de la source des nombres aléatoires est sa reproductivité, de façon que l'on répète l'expérience plusieurs fois dans les mêmes conditions.
  $\Longrightarrow$ nécessité de stocker les données
  $\Longrightarrow$ possible problème de limite de mémoire et difficulté d'accéder aux données

14

Nombres aléatoires:

- Chercher des générateurs (algorithmiques) de nombres au hasard (pseudo-aléatoires).
- L'idée due à Von Neumann est de produire des nombres pseudo-aléatoires en utilisant les opérations arithmétiques (de l'ordinateur):

  commencer par une valeur initiale (u_0 ou u_{-1}) appelée « germe » et générer une suite par la formule $u_i = f(u_{i-1})$ avec f est une fonction.

15

Nombres aléatoires:

- On peut définir une suite de nombres pseudo-aléatoires comme une suite de nombres bien déterminés mais imprévisibles et qui satisfait pourtant à un certain nombres de propriétés.
- Une suite de nombres aléatoires (u_i) est une suite de nombres en [0,1] avec les mêmes propriétés statistiques de l'uniformité (équi-répartition des chiffres) et de l'indépendance des termes.

16

Nombres aléatoires:

- Autres propriétés:
  - Rapidité
  - Petite occupation de mémoire
  - Portabilité
  - Facilité dans l'implantation
  - Reproductivité et mutabilité
  - Périodicité suffisamment longue
- Pour vérifier si ces propriétés sont réalisés de façon raisonnable, on utilise un certain nombre de tests statistiques.

17

Nombres aléatoires:

- Tests Statistiques:
  - Test de $\chi 2$
  - Test de Kolmogorov-Smirnov
- Ce sont deux test d'ajustement ou d'adéquation, c'est-à-dire ils nous permettent de décider si les réalisations obtenus par une méthode sont conformes à la loi désirée.
- Ils sont des tests non paramétriques, permettant, sur la base d'un échantillon $X_1,\ldots,X_n$ de fonction de répartition F, de tester l'adéquation à une loi de fonction de répartition F_0 donnée.

Générateur de milieu de carrée de Von Neumann:

- On choisit un nombre qu'on élève au carré.
- On prélève une partie médiane de ce carré que l'on élève à son tour au carré et on recommence l'opération.
- On définit de proche en proche des nombres par le « milieu » du carré du nombre précédent.

$$\begin{cases} x_0 \text{ } donnée \\ x_{i+1} = \dfrac{x_i^2}{2} \end{cases}$$

19

Générateur de milieu de carrée de Von Neumann:

- Ces nombres ne sont pas aléatoires, ils dépendent du 1er choisi,
- et ils ont, malheureusement, une faible période
- et donc ils sont peu employés en pratique.

20

Générateurs (linéaires) de congruences:

- Lehmer (1951):

$$x_{n+1} \equiv (ax_n + b)[\mathrm{mod}\, m]$$

$$u_n = \frac{x_n}{m}$$

- Pour un
  - multiplicateur a
  - biais b
  - module m
  - valeur initiale x_0

21

Générateurs de congruences:

- Ou, sans perte de généralité, on suppose que

$$a, b \in \{0,1, \dots, m-1\}$$

- Si b=0, ils sont dits multiplicatives.
- Les nombres (u_n) ainsi obtenus ne sont pas vraiment des nombres aléatoires, ils sont dits des nombres pseudo-aléatoires.

22

Congruence (Rappel):

- La congruence modulo m est définie par:
- Un entier p est congrue à un entier r modulo m, si r est le reste de la division euclidienne de p par m.
- cela veut dire qu'il existe un entier q tel que $p=r+qm$, avec $0\leq r\leq m$.

$$p \equiv r[\mathrm{mod}\, m] \Leftrightarrow \exists q \in \mathrm{Z} \,/\, p = r + qm$$

23

GÉNÉRATEURS DE NOMBRE ALÉATOIRE SOUS R

- Pour générer « n » nombres aléatoires qui suivent une loi uniforme (0,1) la commande en R est:

  > runif(n, min=0, max=1)
- Si, nous voulons les représenter sur un graphique, on écrit:

  >x=runif(n, min=0, max=1)
- Alors R les garde le résultat dans x sans l'afficher. Puis on trace le nuage de points:

  >plot(x)
- Si on tape x tout simplement, R les affiche.

GÉNÉRATEURS DE NOMBRE ALÉATOIRE sous R

- Pour la reproductibilité et le contrôle, la graine pour les générateurs de nombre aléatoire devrait être mise explicitement, avant la première utilisation d'un générateur de nombre aléatoire dans le Script.
- La commande est :

  >set.seed(m)
- En changeant la graine et en lançant de nouveau le Script, on obtiendra de différents (indépendants) échantillons à partir des générateurs de nombre aléatoire.

La prochaine fois:

- La question qui se pose maintenant est comment faire pour générer des nombres aléatoires qui suivent une loi de probabilités bien déterminée au départ.
- D'abord, on sait que les méthodes que nous avons vu donne que des nombres qui suivent une loi uniforme.
- En plus, on doit voir comment générer des lois de probabilités discrètes et aussi continues.