26.01.2011

Université de Nice
Département de Mathématiques
NOM :
PRENOM :

L3 MASS, année 2010-2011
Calcul Stochastique et finance (semestre 2)
Date : .
Groupe : .

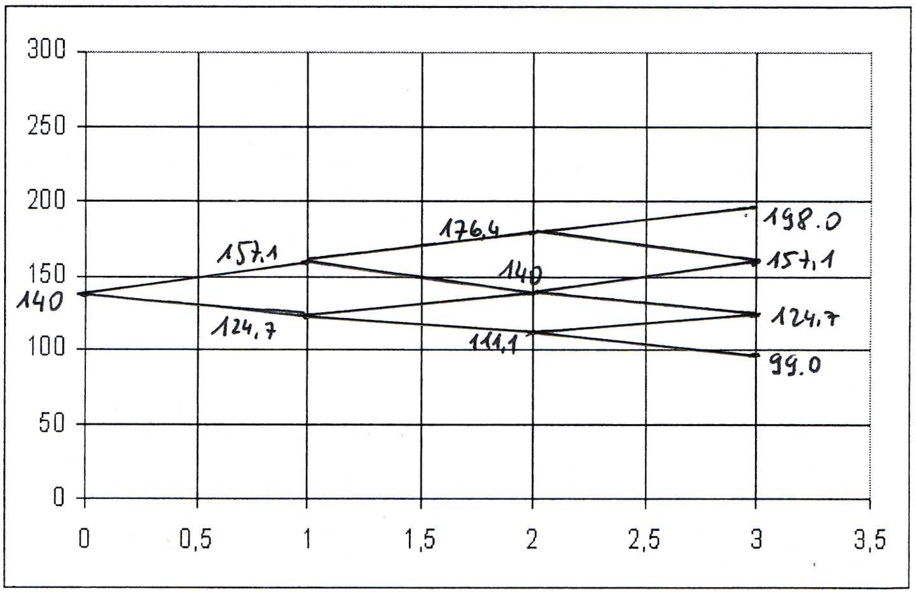
Corrigé

Feuille-réponses du TP 1
Les trajectoires d'un modèle CRR à n étapes

On répondra aux questions posées aussi clairement que possible dans les espaces prévus et on remettra cette feuille de réponses en fin de TP à l'enseignant.

Charger Scilab puis ouvrir un éditeur en cliquant sur `Editeur` dans le menu en haut de fenêtre. Saisir successivement les parties du programme ci-dessous se rapportant à l'exercice concerné. A noter qu'on execute une partie du programme saisi en marquant cette partie et en tapant `Ctrl-Y` .

Exercice 1. : Calcul de la marche CRR On rappelle que la marche CRR S_t est définie par sa valeur initiale S_0 et la relation $S_{t+\delta t} = S_t U_t$, avec $U_t \in \{\texttt{up}, \texttt{down}\}$. On choisit ici $up = e^{+\frac{\sigma}{\sqrt{n}}}$ et $down = e^{-\frac{\sigma}{\sqrt{n}}}$, où σ est une constante (la volatilité de l'actif) et n le nombre d'étapes, $T = n\delta t$. Pour les calculer on range les valeurs de S_t dans une matrice triangulaire[1] $S(i,j) = SS(i+1, j+1)$, où $i = 0, \ldots, n$ est l'indice du pas de temps $t = i\delta t$ et $j = 0, \ldots, i$ le nombre de up intervenus avant t. Calculer au moyen de Scilab les valeurs de S_t dans un modèles à $n = 3$ étapes puis représenter sur le dessin suivant les 10 points de coordonnées $(i, S(i,j))$ pour $i = 0, \ldots, 3$ et $j = 0, \ldots, i$) en les liant pour former l'arbre CRR de ce modèle. Attention au décallage d'une unité de i et j pour l'ordonnée $S(i,j) = SS(1+i, 1+j)$ de ces points !



Que vaut σ ?

σ = sigma = 0.2 selon le code indiqué

exercice complémentaire : recommencer avec $\sigma = 0.4$ et observer la modification du comportement de S pour cette augmentation de la volatilité σ.

[1]La numérotation des lignes et des colonnes par Scilab commence à 1 et non 0 !

Exercice 2. : Tracé d'une trajectoire Une trajectoire de la marche CRR est caractérisée par une suite de up, down que nous représenterons par une suite de 1 et de 0. On désigne par $J(i)$ la fonction qui donne pour une trajectoire le nombre de up entre l'instant $t = 0$ et l'instant $t = i\delta t$. Ainsi pour la trajectoire donnée par la suite $1, 1, 0, 0, 0, 1, 0$, $J(2) = J(5) = 2$ et $J(6) = 3$.

1. Modifier la valeur de n et saisir la suite/seconde partie du programme. Noter qu'il utilise l'instruction `xset("window",1)` qui choisit (et éventuellement crée) la fenêtre 1 comme fenêtre graphique courante (qu'on peut effacer au moyen de `clf(1)`). Il utilise aussi l'instruction `cumsum`. En vous aidant éventuellement de l'aide en ligne de Scilab, indiquer ce que fait cette instruction.

cumsum(M) est une matrice $N = (n_{ij})$ de meme taille que $M = (m_{ij})$ Ses coefficients sont la somme des coefficients de M d'indice inferieur pour "l'ordre lexicographique" $(i,j) \prec (i_0, j_0) \Leftrightarrow (i < i_0) \vee (i = i_0 \wedge j \leq j_0)$

$n_{i_0,j_0} = \sum_{(i,j) \prec (i_0,j_0)} m_{ij}$

2. Faire tracer par `Scilab` plusieurs trajectoires en changeant la suite des 0 et 1. Quelles sont les valeurs atteinte par une trajectoire pour laquelle la succession des 0 et 1 est parfaitement régulière et alternée ? Expliquer le résultat que vous observez.

Pour une succession régulière deltaJ=[0,1,0,1,...] les valeurs sont en dent-de-scie ttraj = [140, 124.7, 140, 124.7, ...]

Exercice 3. : Simulation d'une suite aléatoire de 0 et de 1

1. La fonction `rand()` de Scilab (comme la touche `random` d'une calculette) renvoit un nombre aléatoire compris entre 0 et 1, distribué selon une loi uniforme sur $[0, 1]$. Si l'on précise `rand(m,n)`, on obtient une matrice de taille $m \times n$ dont toutes les composantes sont des nombres aléatoires distribués selon une loi uniforme sur $[0, 1]$. Faire quelques essais pour diverses (petites) valeurs de m et de n. La fonction `2*rand()-1` renvoit encore un nombre aléatoire, mais il n'est plus distribué selon une loi uniforme sur $[0, 1]$. Faire quelques essais puis déterminer quelle loi est simulée par cette fonction.

2 * rand − 1 retourne un nombre aléatoire entre $2 \cdot 0 - 1 = -1$ et $2 \cdot 1 - 1 = 1$ On observe (et surtout comprend) que ces nombre sont uniformément distribués sur $[-1, +1]$.

2. La suite des instructions suivantes permet de vérifier si la loi simulée est bien une loi uniforme : `x=rand(1,1000);histplot(12,x)` Expliquer ce que vous voyez sur la figure produite (dites en particulier ce que représente le "12") et ce que vous devriez voir si la loi était parfaitement uniforme.

La figure produite est constituée de 12 "bâtons" d'un histogramme en bâtons. Ces bâtons ont de taille irrégulière mais de taille à peu près égale : c'est l'histogramme typique d'un échantillon uniformément réparti entre −1 et +1.

3. En vous aidant éventuellement de l'aide en ligne de Scilab, indiquer ce que fait l'instruction `int`.

int(x) retourne un entier. Si x est positif int(x) est la partie entière $[x]$ de x, c'est-à-dire le plus grand entier inferieur ou égal Si $x < 0$, $\text{int}(x) = [x] + 1$. Exemples: int(3.6) = 3 int(−3.6) = −3.

4. La fonction `int(0.5+rand())` renvoit encore un nombre aléatoire, mais il n'est plus distribué selon une loi uniforme sur $[0, 1]$. Faire quelques essais puis déterminer quelle loi est simulée par cette fonction. Quelle est la loi simulée ?

0.5+rand() retourne un nombre aléatoire uniformément distribué sur $[0.5, 1.5]$, et donc de probabilité 0.5 d'être superieur à 1 Donc int(0.5+rand) retourne soit 0, soit 1, avec une probabilité 0.5 d'être égal à 1. Cette instruction simule donc une v.a. de Bernoulli $B(1, 0.5)$.

Exercice 4. : Simulation de M trajectoires de CRR

1. On souhaite à présent simuler un nombre M de trajectoires de la marche CRR. Que retourne, à votre avis, `int(p+rand(n,M))` (pour un $p \in]0,1[$ donné) ?

p+ rand(n,M) retourne une matrice n×M de nombre uniformément distribués sur [p, p+1], donc ayant une probabilité p d'être supérieur à 1. Donc int(p+rand(n,M)) retourne une matrice n×M de nombres simulant une loi de Bernoulli B(1, p)

2. Et, si l'on pose `deltaJ=int(p+rand(n,M))`, que retourne `cumsum(deltaJ` ,"c" `)` ?

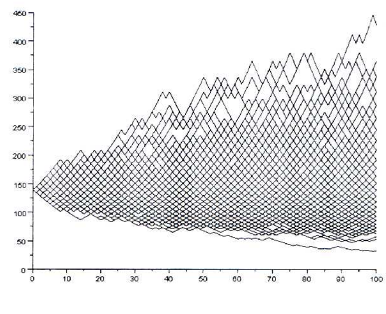
cumsum(deltaJ, "c") est la matrice de même taille que deltaJ dont les lignes sont constituées des "cumsum" des lignes de deltaJ. Comme deltaJ = int(p + rand(n,M)) est une matrice n×M de 0 et 1. MJ = cumsum(deltaJ, "c") est une matrice dont les lignes sont des suites croissantes d'entiers, augmentant d'un plus 1, avec probabilité p d'augmenter de 1.

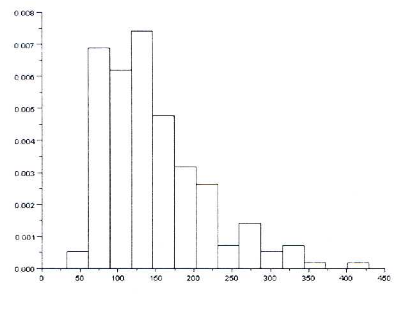
3. On suppose $n = 100$. Simuler $M = 40$ trajectoires en choisissant $p = 0.5$. En reproduisant la simulation avec d'autres valeurs de la volatilité σ, étudier et expliquer l'influence de ce paramètre sur la forme des trajectoires.

Les trajectoires partent toutes de 140 et se dispersent pour former un faisceau en forme de parabole "couchée" issue de (0, 140)
En augmentant la valeur de σ (de 0,2 à 0.4 par exemple) on observe une plus grande dispersion des trajectoires : les paraboles sont plus évasée.

Exercice 5. : Loi de la marche CRR à l'instant T. Représentez un histogramme des valeurs finales (on prendra par exemple $n = 200$) des trajectoires aléatoires de CRR ; pour cela, on simulera $M = 400$ trajectoires au moins et on prendra par exemple `int(sqrt(M))` pour nombre de classes. Indiquer l'allure de l'histogramme obtenu.

On obtient un histogramme en cloche asymétrique irrégulière





```
//Exercice 1 : Calcul de la marche de CRR /////////////
n=3 ;T=1 ;delta_t=T/n ;
SS=zeros(n+1,n+1) ;
S0=140 ;sigma=0.2 ;
up=exp(sigma*sqrt(delta_t)) ;down=1/up ;
SS(1,1)=S0 ;
for i=1 :n
SS(i+1,0+1)=SS(i,0+1)*down; // ici j=0
for j=1 :i
SS(i+1,j+1)=SS(i,j)*up ;
end ;
end ;
// Exercice 2 : Tracé d'une trajectoire de CRR ; ici on choisit n=10 ;
deltaJ=[1 1 0 0 0 1 0 0 1 1] ;
J=cumsum(deltaJ) ;
ttraj=zeros(1,n+1) ;
ttraj(1)=SS(1,1) ;
for i=1 :n
ttraj(i+1)=SS(i+1,J(i)+1) ;
end ;
xset("window",1) ;
plot2d(0 :10,ttraj) ;
// Exercice 3 : Simulation d'une suite de 0 et de 1 /////////////////////////
rand() ;
rand(3,2) ;
rand(3,3) ;
2*rand()-1 ;
x=rand(1,1000) ;
xset("window",2) ;
histplot(12,x) ;
int(0.5+rand(1,10)) ;
// Exercice 4 : Simulations de M trajectoires de CRR /////////////////
M=100 ; n=100 ;
MdeltaJ=int(0.5+rand(M,n)) ;
MJ=cumsum(MdeltaJ,"c") ;
Mttraj=zeros(M,n+1) ;
for m=1 :M
Mttraj(m,1)=S0 ;
for i=1 :n
Mttraj(m,i+1)=SS(i+1,MJ(m,i)+1) ;
end ;
end ;
xset("window",3) ;
for m=1 :M
plot2d(0 :n,Mttraj(m,1 :n+1)) ;
end ;
// Exercice 5 : Loi de la marche CRR l'instant T ; ici on choisira M=400 ;
xset("window",4) ;
histplot(int(sqrt(M)),Mttraj(1 :M,n+1)) ;
```