Université de Nice
Département de Mathématiques

L3MASS et L3Maths, année 2018-2019
Calcul stochastique et Finance

NOM :
PRENOM : Corrigé

Date :
Groupe :

Feuille-réponses du TP 1
Simulation de trajectoires du modèle CRR

Prise en main de Scilab

Recherchez le programme Scilab et lancez-le. La fenêtre qui s'ouvre s'appelle la *console*. Elle sert essentiellement à l'affichage des résultats. On peut y saisir une commande, telle la demande d'affichage de la valeur d'une variable (on tape le nom de la variable, puis "return"), mais il convient d'éviter de l'utiliser pour des commandes plus compliquées telles les boucles, etc. car il n'est pas facile de demander la réexecution.

Pour saisir une succession de commandes, il convient d'ouvrir une *fenêtre d'édition*. Pour cela utiliser la commande de menu **Application-Editeur**. Il s'agit d'un assez simple éditeur de texte (il reconnait quand même le nom des commandes et constantes de Scilab qu'il affiche d'une autre couleur). Comme pour tout éditeur, il convient de commencer par donner un nom au fichier édité. Pour cela, utiliser la commande de menu **Fichier-Sauvegarder-sous** et choisir le nom `CalSto1.sce` et éventuellement le disque et répertoire où vous voulez sauvegarder votre séance (votre clé USB par exemple). Il est essentiel de garder les fichiers de vos séances : cela ne vous sera pas seulement utile pour vos révisions; dès la semaine suivante vous pourrez repartir des idées de la semaine précédente pour développer de nouvelles questions.

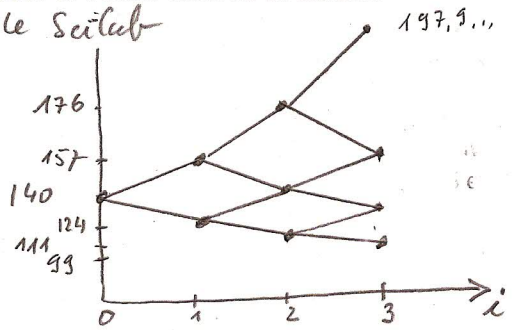
A présent, vous allez taper les instructions successives dans votre fenêtre d'édition, puis les marquer et demander l'exécution de la zône marquée par la commande de menu **Executer-Evaluer-la-selection**. Ainsi vous construisez progressivement une succession d'instructions et en testez l'effet en lisant dans la **console** les résultats produits. Vous pourrez ensuite réexécuter tout ou partie de votre code, dans lequel vous aurez peut-être modifié la valeur de quelques constantes.

Avant d'exécuter tout ou partie de votre code, tapez régulièrement **Ctrl-S** (ou exécutez la commande de menu **Fichier-Sauvegarder**) car il arrive que Scilab plante(!) (ou déjante(!!)) c'est-à-dire s'arrête complètement et il ne vous reste plus qu'à repartir, en double-cliquant sur le fichier que vous avez fort heureusement sauvegardé! Notez que la fenêtre d'édition s'affiche souvent par dessus la fenêtre console, ce qui gêne pour lire les résultats sur cette console. Pour cela, **modifiez la place et la taille des deux fenêtres :** à gauche la console et à droite léditeur plus large, les deux s'étirant du haut en bas de votre écran. Évitez tout chevauchement.

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 définit une fonction $S(i,j) = S_0 up^j down^{(i-j)}$, 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 le début de l'arbre CRR de ce modèle.

Le calcul au moyen de la fonction S de Scilab donne les résultats suivants

$S(0,0) = 140 = S(2,1)$
$S(1,0) = 124.73262 = S(3,1)$
$S(1,1) = 157,1\ldots = S(3,2)$
$S(2,0) = 111,1\ldots$; $S(2,2) = 176,3\ldots$
$S(3,0) = 99,0\ldots$; $S(3,3) = 197,9\ldots$



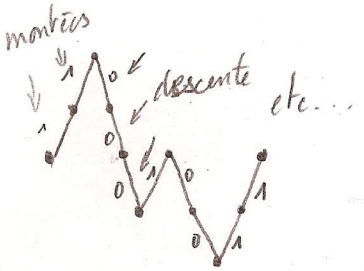
Exercice 2. : Représentation d'une trajectoire Une trajectoire de la marche CRR est caractérisée par une suite de `up` et de `down` que nous représenterons par une suite de 1 et de 0. On désigne par $J(i)$ la fonction qui donne pour une trajectoire le nombre de `up` entre l'instant $t = 0$ et l'instant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

1. Examiner la seconde partie du code et tacher d'en comprendre chaque instruction (avec l'aide en ligne au besoin). Exécutez-le, décrire la figure obtenue et expliquer.

J = cumsum(deltaJ) retourne la ligne des sommes cumulées (de gauche à droite) des éléments de deltaJ

deltaJ = [1 1 0 0 0 1 0 0 1 1]

J = [1 2 2 2 2 3 3 3 4 5]



2. Faire tracer par `Scilab` plusieurs trajectoires en changeant la suite des 0 et 1. Comment obtenir une trajectoire dont les deux extrémités sont $S0$? Quelle est la valeur la plus petite possible qu'une trajectoire peut atteindre (lorsque $n = 10$) ?

Pour que $S(n,j) = S(0,0)$ il faut que $j = n/2$ pour qu'il y ait autant de "up" que de "down"

Pour atteindre la plus petit valeur il faut n'avoir que des "down" i.e $J(n) = 0$ et alors $S(n, J(n)) = S(n,0) = 74,37..$

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) renvoie 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()-3` renvoie 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. Expliquer.

2 * rand( ) est un nombre aléatoire, entre 0 et 2 et donc

2 * rand( ) - 3 est un nombre aléatoire entre -3 et -1

qui suit une loi uniforme sur $[-3,-2]$

2. La suite des instructions suivantes permet de vérifier si la loi simulée est bien une loi uniforme : `x=rand(1,1000);histplot(10,x)`. Que vaut x ? Que vaut `x(1,1:10)` ? Expliquer ce que vous voyez sur la figure produite. Indiquer quelle est la classe de plus grand effectif et celle de plus petit ? Voyez-vous comment calculer leurs effectifs ?

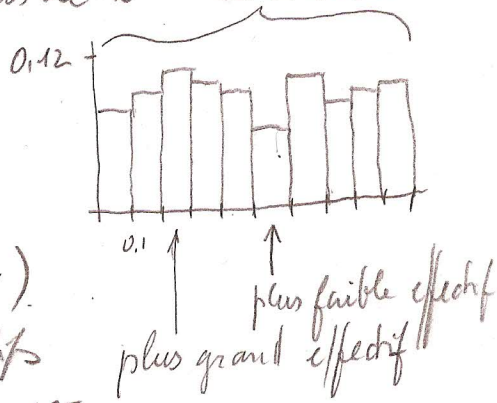
x est une matrice ligne de 1000 valeurs aléatoires entre 0 et 1

x(1,1:5) sont les 5 premières composantes de x

La figure produite est composée de 10 barres de même largeur, 0,1, et dont les hauteurs sont égales aux fréquences $\frac{\text{effectif}(i)}{\text{effectif total}}$;

on vérifie que sum(histplot(10,x)) = 1).

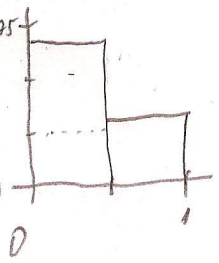
1000 * histplot(10,x) retourne les effectifs



chez moi : 84 99 116 102 97 81 113 100 103 105

3. La suite des instructions suivantes permet de simuler une suite de 1000 entiers 0 ou 1 distribués selon une loi de Bernoulli $\mathcal{B}(1,p)$: `y=grand(1,1000,"bin",1,p); histplot(2,y)`. Expliquer ce que vous voyez sur la figure produite pour `p=1/4`.

y est un vecteur ligne de 1000 composantes égales à 0 ou 1 tirée aléatoirement avec une probabilité $p=0.25$ de valoir 1 et $1-p=0.75$ de valoir 0.
Dans mon cas histplot(2,y) me retourne un histogramme comme ci-contre. La fréquence de 0 est bien voisine de 0,75 (en fait inférieure) et celle de 1 est (donc) voisine de 0,25 (mais supérieure)



Exercice 4. : Simulation de M trajectoires de CRR

1. Examiner la partie du code correspondant à cette question et tacher d'en comprendre chaque instruction. Que représente `MdeltaJ`, que représente `cumsum(MdeltaJ,"c")` ?

MdeltaJ est une matrice de 0 et de 1 de 20 lignes et 40 colonnes, tirés aléatoirement avec même probabilité $\frac{1}{2}$ pour 0 et pour 1
MJ = cumsum(MdeltaJ,"c") est la somme cumulative des colonnes de MdeltaJ.

2. On suppose $n = 40$. Simuler $M = 20$ trajectoires en choisissant $p = 0.5$. Expliquer ce que vous voyez sur la figure obtenue.

Toutes les trajectoires sont issues de (0,140) puis se dispersent sur l'autre CRR. Comme elles se superposent exactement sur certaines portions on a du mal à les distinguer mais il semble bien qu'il y en a moins qui aboutissent aux valeurs extrêmes

Exercice 5. : Le fouet : ensemble de trajectoires coïncidant jusqu'à un instant donné. Pour tracer un tel ensemble de M trajectoires qui coïncident jusqu'à l'instant $i = 10$, il suffit de redéfinir la matrice `MdeltaJ` pour les 10 premiers coefficients de chaque ligne en les remplaçant par la même suite (ici la suite deltaJ de l'exercice 2). Si l'on compare ce fouet avec celui que l'on aurait obtenu en remplaçant seulement les 5 premiers coefficients, ~~lequel~~ l'un des deux fouets est contenu dans l'autre ? Expliquer.

Non ces deux fouets ne sont pas contenus l'un dans l'autre. Toutefois l'atome de celui dont toutes les trajectoires coïncident jusqu'à $i=10$ ($t=10\,deltat=10/40=0.25$) est contenu dans l'atome de celui dont toutes les trajectoires ne coïncident que jusqu'à $i=5$ ($t=0.125$)

Exercice 1 : la marche aléatoire CRR

```
clear; // supprime les variables, functions, ... déja introduites
n=3;T=1; deltat=T/n;S0=140; sigma=0.2;
up=exp(sigma*sqrt(deltat));down=1/up;
function s=S(i,j); // on doit avoir 0<=j<=i<=n
   s=S0*up.^j.*down.^(i-j);
endfunction;
```

Exercice 2 : Tracé d'une trajectoire de CRR de longueur 10

```
deltaJ=[1 1 0 0 0 1 0 0 1 1];
J=cumsum(deltaJ);
n=10; deltat=T/n ;up=exp(sigma*sqrt(deltat));down=1/up;
ttraj=zeros(1,n);
ttraj(1)=S(0,0);
for i=1:n
    ttraj(i+1)=S(i,J(i));
end ;
scf(1); //crée et active la fenêtre 1
plot2d(1:10,ttraj);
```

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

```
x=rand(1,40) ;
scf(2);
histplot(10,x)
y=grand(1,40,''bin",1,0.25);
scf(3);
histplot(2,y)
```

Exercice 4 : Simulations de M trajectoires de CRR de longueur n

```
M=20; n=40;up=exp(sigma*sqrt(deltat));down=1/up; p=1/2;
MdeltaJ=grand(M,n,''bin",1,p);
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)=S(i,MJ(m,i));
    end ;
end ;
scf(4) ;
for m=1:M
    plot2d(0:n,Mttraj(m,1:n+1));
end ;
```

Exercice 5 : Ensemble de trajectoires coïncidant au debut

```
for m=1:M
    MdeltaJ(m,1:10)=deltaJ;
end;
MJ=cumsum(MdeltaJ,"c");
for m=1:M
    Mttraj(m,1)=S0 ;
    for i=1:n
       Mttraj(m,i+1)=S(i,MJ(m,i));
    end ;
end ;
scf(4) ;
for m=1:M
    plot2d(0:n,Mttraj(m,1:n+1)) ;
end ;
scf(5);
for m=1:M
    plot2d(0:n,Mttraj(m,1:n+1));
end;
```