

Mini projet 1: Calcul du prix d'une option asiatique

Valentin DE CRESPIN DE BILLY

Matthias LANG

30.11.2021

N. d'étudiant : XXXXXXXX et 313411

Université Catholique de l'Ouest

Mathématiques financières

1 Calculer le prix du sous-jacent

$$dS_t = S_t(rdt + \sigma\sqrt{S_t}dW_t) \quad (1)$$

$$\Longleftrightarrow \frac{dS_t}{S_t} = rdt + \sigma\sqrt{S_t}dW_t \quad (2)$$

On prend l'équation 1 :

$$= dS_t = S_t rdt + \sigma S_t^{1.5} dW_t$$

Puis

$$\begin{aligned} d\langle S_t, S_t \rangle &= \langle dS_t, dS_t \rangle = \\ &= \langle S_t rdt + \sigma S_t^{1.5} dW_t, S_t rdt + \sigma S_t^{1.5} dW_t \rangle = \\ &= \langle \sigma S_t^{1.5} dW_t, \sigma S_t^{1.5} dW_t \rangle = \\ &= S_t^3 \sigma^2 \langle dW_t, dW_t \rangle = \\ &= S_t^3 \sigma^2 dt \end{aligned} \quad (3)$$

On pose : $X_t = \ln(S_t)$

$$\text{Formule d'Ito : } d\ln(S_t) = \frac{dS_t}{S_t} + \frac{1}{2} \frac{1}{S_t^2} d\langle S_t, S_t \rangle$$

Avec les équations 2 et 3 : (4)

$$\begin{aligned} d\ln(S_t) &= rdt + \sigma\sqrt{S_t}dW_t - \frac{1}{2}S_t\sigma^2dt = \\ &= (r - \frac{1}{2}S_t\sigma^2)dt + \sigma\sqrt{S_t}dW_t \end{aligned}$$

$$\begin{aligned} \ln\left(\frac{S_t}{S_0}\right) &= \ln(S_t) - \ln(S_0) = \int_0^t d\ln(S_u) = \\ &= \int_0^t (r - \frac{1}{2}S_u\sigma^2)du + \int_0^t \sigma\sqrt{S_u}dW_u \end{aligned} \quad (5)$$

...

Donc on ne peut pas facilement dériver une formule pour le prix qui dépend que des variables fixées, mais on peut le simuler pas à pas avec (1) :

$$\begin{aligned}
S_0 &\text{ soit connu} \\
dS_0 &= S_0(rdt + \sigma\sqrt{S_0}dW_0) \\
S_1 &\approx S_0 + dS_0 \\
dS_1 &= S_1(rdt + \sigma\sqrt{S_1}dW_1) \\
S_2 &\approx S_1 + dS_1 \\
&\dots
\end{aligned}
\tag{6}$$

Appendices

A Code Matlab

```

%~~~~~%
% UTF-8 %
% 30.11.2021 %
% Valentin DE CRESPIN DE BILLY %
5 % Matthias LANG %
%~~~~~%

% ~~~~~~ %
% ~~~~~~ Mathematiques financieres: Mini-projet 1 ~~~~~~ %
10 % ~~~~~~ %

%% ~~~~~~ Parametres ~~~~~~ %%

15 S0 = 40; % Prix initial du sous jacent
N = 5; % Nombre des sous-intervalles
% verifier que N << n => a faire: ecrire un test
K = 46; % Prix d'exercice de l'option

20 r = 0.05; % Taux d'interet sous risque neutre
sigma = 0.04/sqrt(S0); % Variance partie fixe

```

```

t0 = 0;                                % Debut de la periode
n = 2^4;                               % Nombre de intervalles
25 T = 3;                              % Fin de la periode
nt = 5;                                % Nombre de trajectoires

starttime = datetime('now');

30 fprintf('La programme a demarre a %s \n', starttime);
fprintf('%d -> Nombre de trajectoires \n', nt);
fprintf('%d -> Prix initial du sous jacent \n', S0)
syms func(x)
obligation(x) = S0*(1+r)^(x-t0);
35 %K = int(obligation,t0,T)/(T-t0);
bonds_T = obligation(T);
fprintf('%0.5g -> Prix d''une obligation a T\n', bonds_T)
fprintf('%0.5g -> Prix d''exercice de l''option \n', K);
fprintf(' . . . ')
40 tic

%% ~~~~~ Simulation ~~~~~ %%

45

dt = (T-t0)/n;
t = t0:dt:T;

S = zeros(n+1,nt);
50 S(1,:) = S0;

for i = 2:(n+1)
    dW_t = normrnd(zeros(1,nt),sqrt(dt));
    dSi = S(i-1,:).*( r*dt + sigma*sqrt(S(i-1,:)).*dW_t );
55 S(i,:) = S(i-1,:) + dSi;

```

```

end

%% ~~~~~ calcul de  $X_t$  ~~~~~ %%

60

X_T = 0.5*S0 + sum(S(2:n,:),1) + 0.5*S(n+1,:);
X_T = X_T/n;

65 C_inf = X_T-K;
C_inf = C_inf .* ( X_T - K >= 0 );

C_inf_0 = exp(-r*T)*C_inf;

70 % ~ Estimateur ~
%  $C_{inf} * \exp(-rT)$  est une martingale donc
%  $E[\exp(-rT) * C_{inf}] = C_{inf}(S_0)$ 

C_inf_est = mean(C_inf_0);
75 C_inf_est_var = var(C_inf_0);

fprintf('L''estimateur du C a t0 = %0.5g\n', ...
        C_inf_est);
fprintf('Son ecart type = %0.5g\n', sqrt(C_inf_est_var));

80

%% ~~~~~ calcul de  $X_{t\_prim}$  ~~~~~ %%

85 vecX_t_prim = S(1,:);
for i = 2:(n+1)
    vecX_t_prim = vecX_t_prim + S(i,:);
end
vecX_t_prim = vecX_t_prim/(n+1);

```

```

90 X_t_prim = mean(vecX_t_prim);
    vecC_N = vecX_t_prim-K;

    vecC_N = vecC_N .* ( vecC_N >= 0 );

95
    C_N = mean(vecC_N);
    %C_N * exp(-rT) est une martingale donc
    % E[exp(-rT)*C_N]= C_N(S_0)
    C_N_0 = exp(-r*T)*C_N;

100
    % Histogramm
    %  $E_{-}\pi(e^{-rT}(X_T - K)^+ / F_0) \sim 1/nt \sum \{C(T)\}$ 

105 %% ~~~~~~ Plot ~~~~~~ %%

    duree= toc;
    fprintf('%d trajectoires plotees\n', nt);
110 fprintf('Avec S0 = %d, K = %0.5g \n', S0, K);
    fprintf('L''integrale par (t_0 - T) de X_t, ');
        fprintf('le prix estime C(T) = %0.5g \n', C_inf_0);
    fprintf('La moyenne des X_t: C(T) = %0.5g \n', C_N_0);
    fprintf('Finis en %0.5g\n', duree);

115

    tiledlayout(2,1)
    nexttile
    hold on

120
    x_ax = t; % x-axe
    axis([0 T 0.8*min(min(S)) 1.5*max(max(S))]) %x-axe limits

```

```

plot(x_ax, S)
125
% pour comparaison, si j'epargne pour le taux r:
%plot([t0 T], [S0 S0*(1+r)^(T-t0)], "--k"); % obligation
fplot(obligation, [t0 T], "-k");

130 legend("les prix S_t des actions", "sans risque");

hold off

nexttile

135 histogram( vecC_inf );
title("Histogramm des C(T) pour X_{infinie}");

%legend("", "l'estime pour X_inf", "l'estime pour X_N'")
140 %plot([C_chapeau C_chapeau], ...
%      [-0.2*nt/sqrt(sigma*nt) nt/sqrt(sigma*nt)], "-k");
%plot([t0 T], [0 0], ":k"); % y=zero

145 %% ~~~~~ fonctions ~~~~~ %%

%% a effacer, aine

150 % for i = 1:nt
%      S(:,i) = brownmo(S0, r, sigma ,t0, T, n);
% end

function S = brownmo(X0, mu, sigma, t0, T, n) %x0
155     delta = (T-t0)/n;
     W = zeros(1,n+1);
     tseq = t0:((T-t0)/n):T;

```

```

    for i = 2:(n+1)
        W(i) = W(i-1)+normrnd(0,1)*sqrt(delta);
160    end
    S = X0 * exp( (mu-(sigma^2)/2) * (tseq-t0) + sigma*W );
end

165
function X_t = MoyMob(M, t_m)
    X_t = cumsum(M,1);
    t_m2= t_m+1;
    X_t = X_t(1:t_m2,:);
170    for i = 1:t_m2
        X_t(i,:) = X_t(i,+)/i; %vectoriel?
    end
end

175 % Xn = MoyMob(S, n);

% calculer X_T'
% X_pr = zeros(N,nt);
% for i = 1:N %vectoriel?
180 %     X_pr(i,:) = S(floor(i*n/N),:);
% end
% X_pr = (1/N)*sum(X_pr, 1);

```

B Code VBA

```

Rem Attribute VBA_ModuleType=VBAModule
Option VBASupport 1
5 Sub Macro1()

```



```

Dim T, n, nt, Nd As Integer
Dim r, sigma, S0, t0 As Double

10 Dim i, j As Integer

r = Range("A2").Value
sigma = Range("A3").Value
15 T = Range("A4").Value
n = Range("A5").Value
nt = Range("A6").Value
Nd = Range("A7").Value
S0 = Range("A8").Value
20 t0 = Range("A9").Value

Dim dt As Double
dt = ((T - t0) / n)

25 Dim temps() As Double
ReDim temps(n + 1)
temps(0) = t0
For j = 1 To n + 1
    temps(j) = temps(j - 1) + dt
30 Next

Range("I2:I" & UBound(temps) + 1) = WorksheetFunction.
    Transpose(temps)

35 Dim S() As Double
ReDim S(1 To n + 1, 1 To nt)
Dim W As Double
Dim x As Double

```

```

40 For j = 0 To nt - 1
    For i = 0 To n

        If i = 0 Then
            W = 0
        45 Else
            W = W + Sqr(-2 * Log(Rnd())) * Cos(6.283185307 * Rnd()
                ) * Sqr(dt)
        End If

        x = S0 * Exp((r - (sigma ^ 2) / 2) * (temps(i) - t0) +
            sigma * W)
    50 Cells(2 + i, 10 + j).Value = x ' copier s dans la
        worksheet
        'S(i, j) = x

    Next
Next

55

'Cells(i, 3).Value =

60

'S = zeros(length(t),nt);
'for i = 1:nt
'    S(:,i) = brownmo(S0, r, sigma ,t0, T, n); %brownmo
    est definie en bas
'End

65

'function S = brownmo(X0, mu, sigma, t0, T, n) %x0

'    delta = (T-t0)/n;
'    W = zeros(1,n+1);

```

```

70 ' tseq = t0:((T-t0)/n):T;
' for i = 2:(n+1)
'     W(i) = W(i-1)+normrnd(0,1)*sqrt(delta);
' End
' S = X0 * exp( (mu-(sigma^2)/2) * (tseq-t0) + sigma*W );
75
MsgBox "Simule pour " & nt & " trajectoires."

'Sheets("Dashboard").Activate
'Range("Parametres").Select
80 'Range("A13").Value = T

End Sub

```