

PSTAT 222C Homework 1

Problem 1

We consider a corridor option for asset X_t with payoff of \$1 if $X_t \in [L_1, L_2]$ and \$0 otherwise. X_t is governed by the SDE

$$dX_t = rX_t dt + \sigma^2 X_t^\gamma dW_t \quad (1)$$

where γ is the elasticity of the variance. Applying Ito's Lemma to the price of the option, $V(t, x)$, we get:

$$dV = \frac{\partial V}{\partial t} dt + \frac{\partial V}{\partial x} dX_t + \frac{1}{2} \frac{\partial^2 V}{\partial x^2} dX_t^2 \quad (2)$$

$$= \frac{\partial V}{\partial t} dt + \frac{\partial V}{\partial x} (rX_t dt + \sigma^2 X_t^\gamma dW_t) + \frac{1}{2} \frac{\partial^2 V}{\partial x^2} (rX_t dt + \sigma^2 X_t^\gamma dW_t)^2 \quad (3)$$

$$= \left(\frac{\partial V}{\partial t} + rX_t \frac{\partial V}{\partial x} + \frac{\sigma^2 X_t^{2\gamma}}{2} \frac{\partial^2 V}{\partial x^2} \right) dt + \sigma X_t^\gamma \frac{\partial V}{\partial x} dW_t \quad (4)$$

Under the assumption that this is the risk-neutral measure, we must have that the expected price of the option is equal to the present value of the risk free rate of the price of the option; that is:

$$\mathbb{E}[dV] = rV dt = \left(\frac{\partial V}{\partial t} + rX_t \frac{\partial V}{\partial x} + \frac{\sigma^2 X_t^{2\gamma}}{2} \frac{\partial^2 V}{\partial x^2} \right) dt \quad (5)$$

It therefore follows that the PDE for the CEV model can be written as:

$$\frac{\partial V}{\partial t} + rX_t \frac{\partial V}{\partial x} + \frac{\sigma^2 X_t^{2\gamma}}{2} \frac{\partial^2 V}{\partial x^2} - rV = 0 \quad (6)$$

Let the value of the discretization of the PDE at time $m\Delta t$ and $n\Delta x$ be $V(m\Delta t, n\Delta x) = V_n^m$. The explicit finite difference scheme is given by:

$$\frac{V_n^m - V_n^{m-1}}{\Delta t} + r(n\Delta x) \frac{V_{n+1}^{m-1} - V_{n-1}^{m-1}}{2\Delta x} + \frac{\sigma^2 (n\Delta x)^{2\gamma}}{2} \frac{V_{n+1}^{m-1} - 2V_n^{m-1} + V_{n-1}^{m-1}}{(\Delta x)^2} - rV_n^{m-1} = 0 \quad (7)$$

We can derive the implicit Finite Difference similarly

$$\frac{V_n^m - V_n^{m-1}}{\Delta t} + r(n\Delta x) \frac{V_{n+1}^m - V_{n-1}^m}{2\Delta x} + \frac{\sigma^2(n\Delta x)^{2\gamma}}{2} \frac{V_{n+1}^m - 2V_n^m + V_{n-1}^m}{(\Delta x)^2} - rV_n^m = 0 \quad (8)$$

Solving for V_n^{m-1} , we find:

$$V_n^{m-1} = V_n^m + \Delta t \left(r(n\Delta x) \frac{V_{n+1}^m - V_{n-1}^m}{2\Delta x} + \frac{\sigma^2(n\Delta x)^{2\gamma}}{2} \frac{V_{n+1}^m - 2V_n^m + V_{n-1}^m}{(\Delta x)^2} - rV_n^m \right) \quad (9)$$

$$= \underbrace{\frac{\Delta t}{2} \left(-rn + \frac{\sigma^2(n\Delta x)^{2\gamma}}{(\Delta x)^2} \right)}_{\tilde{a}_n} V_{n-1}^m + \left(1 - r\Delta t + \underbrace{\frac{\sigma^2(n\Delta x)^{2\gamma}\Delta t}{(\Delta x)^2}}_{\tilde{b}_n} \right) V_n^m + \underbrace{\frac{\Delta t}{2} \left(rn + \frac{\sigma^2(n\Delta x)^{2\gamma}}{(\Delta x)^2} \right)}_{\tilde{c}_n} V_{n+1}^m \quad (10)$$

$$= \tilde{a}_n V_{n-1}^m + (1 - \tilde{b}_n) V_n^m + \tilde{c}_n V_{n+1}^m \quad (11)$$

Imposing the exogenous boundary at L_1 and L_2 such that for all $0 \leq m \leq T/\Delta t$, $V_{L_1}^m = V_{L_2}^m = 1$ and $V_{L_1-1}^m = V_{L_2+1}^m = 0$ we find that we can represent the discretized PDE as:

$$AV^{m-1} + g^{m-1} = BV^m \quad (12)$$

$$V^{m-1} = A^{-1}(BV^m - g^{m-1}) \quad (13)$$

where $B = I$, the identity matrix.

To see how to apply the boundary conditions, it is helpful to visualize the matrix representation of this equation. We have:

$$\begin{pmatrix} \tilde{a}_1 & 1 + \tilde{b}_1 & \tilde{c}_1 & 0 & \cdots & 0 \\ 0 & \tilde{a}_2 & 1 + \tilde{b}_2 & \tilde{c}_2 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \cdots & 0 \\ 0 & 0 & 0 & \tilde{a}_{N-1} & 1 + \tilde{b}_{N-1} & \tilde{c}_{N-1} \end{pmatrix} \begin{pmatrix} V_0^{m-1} \\ V_1^{m-1} \\ \vdots \\ V_{N-1}^{m-1} \\ V_N^{m-1} \end{pmatrix} = \begin{pmatrix} V_1^m \\ V_1^m \\ \vdots \\ V_{N-1}^m \\ V_{N-1}^m \end{pmatrix} \quad (14)$$

and it becomes clear that g^{m-1} is given by:

$$g^{m-1} = \begin{pmatrix} \tilde{a}_1 V_0^{m-1} \\ 0 \\ \vdots \\ 0 \\ \tilde{c}_{N-1} V_N^{m-1} \end{pmatrix} \quad (15)$$

Implicit Finite Difference with Exogenous Boundary Conditions

```

gam<-0.8
dt <- 0.1
T = 1/2
dx <- 0.2
L1 <- 15
L2 <- 25

implicitfd<- function(T,dt,dx,X_min,X_max,gam,opttype,K=0,other_asset=0,x0){
  r <- 0.05
  sigma <- 0.4
  time_steps <- as.integer(T/dt)
  space_steps <- as.integer((X_max-X_min)/dx)+1
  vetS <- X_min + dx*(0:(space_steps-1))

  V = matrix(0,space_steps,time_steps) # initialize matrix

  vetI <- X_min/dx + 0:(space_steps-1)
  a_i <- dt/(2*(sigma^2*(vetI*dx)^(2*gam)/(dx^2) - r*vetI)
  b_i <- -dt*(sigma^2*(vetI*dx)^(2*gam)/(dx^2) + r)
  c_i <- dt/(2*(sigma^2*(vetI*dx)^(2*gam)/(dx^2) + r*vetI)

  Amatrix <- diag(1-b_i,space_steps)
  Amatrix[(row(Amatrix) - col(Amatrix)) == 1] <- -a_i[2:(space_steps)]
  Amatrix[(row(Amatrix) - col(Amatrix)) == -1] <- -c_i[1:(space_steps-1)]
  if(opttype=="cor"){
    V[,ncol(V)] <- 1 - ((vetS>L2 | vetS<L1)*1)
  }
  if(opttype=="call"){
    V[,ncol(V)] <- pmax(vetS-K,0)
    V[nrow(V),] <- X_max - K*exp(-r*T)
  }
  if(opttype=="compound"){
    V[,ncol(V)] <- pmax(other_asset-K,0)
    V[nrow(V),] <- other_asset[length(other_asset)] - 20*exp(-r*T) #hardcoded strike for other asset
  }

  V[1,] = 0

  Bmatrix <- diag(1,space_steps)
  for (k in 1:time_steps-1){
    t = time_steps - k
    g <- c(rep(0,space_steps-1),c_i[length(c_i)]*(V[nrow(V),ncol(V)-1]))
    g[1] = a_i[1]*V[1]
    V[,t-1]<-solve(Amatrix, Bmatrix %*% V[,t] - g)
  }
}

```

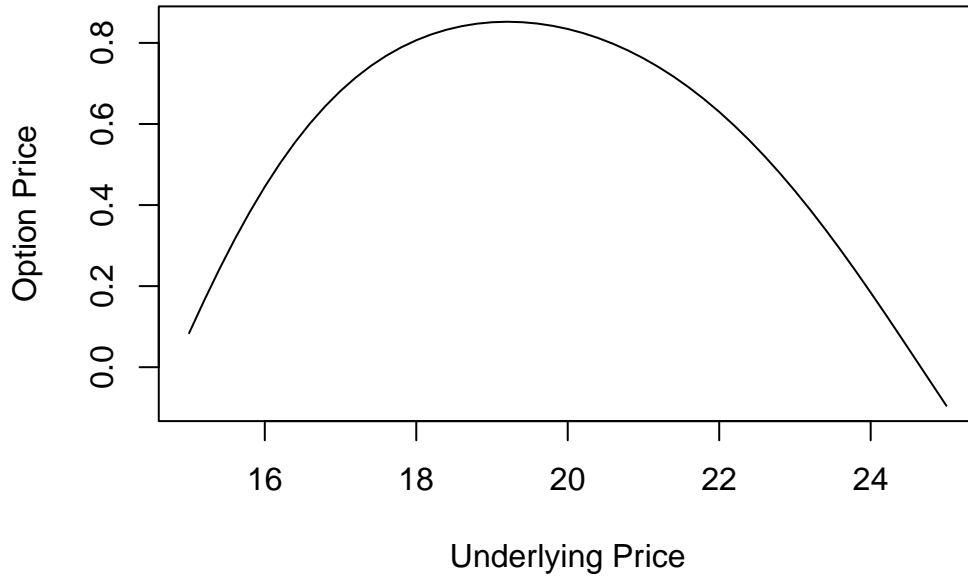
```

    }
    return(list(V=V,price=V[which(vetS==x0)]))
}
val<-implicitfd(T,dt,dx,L1,L2,gam,"cor",0,0,20)

val<-implicitfd(T,dt,dx,L1,L2,gam,"cor",0,0,20)
options(scipen=999)
plot(x = seq(15,25,dx), y= val$V[,1],type='l',main="Implicit Method for Corridor Option",
ylab="Option Price",xlab = "Underlying Price" )

```

Implicit Method for Corridor Option



Crank-Nicholson Solver

The Crank-Nicholson Method is essentially an average of the Explicit and Implicit Finite Difference Methods. Writing this out, we have:

$$\frac{V_n^m - V_n^{m-1}}{\Delta t} = \frac{1}{2} \left(-r(n\Delta x) \frac{V_{n+1}^m - V_{n-1}^m}{2\Delta x} - \frac{\sigma^2(n\Delta x)^{2\gamma}}{2} \frac{V_{n+1}^m - 2V_n^m + V_{n-1}^m}{(\Delta x)^2} + rV_n^m \right) \quad (16)$$

$$+ \frac{1}{2} \left(-r(n\Delta x) \frac{V_{n+1}^{m-1} - V_{n-1}^{m-1}}{2\Delta x} - \frac{\sigma^2(n\Delta x)^{2\gamma}}{2} \frac{V_{n+1}^{m-1} - 2V_n^{m-1} + V_{n-1}^{m-1}}{(\Delta x)^2} + rV_n^{m-1} \right) \quad (17)$$

```

cnFd<-function(dt,dx,gam,opttype,T){
  r <- 0.05
  sigma <- 0.4
  L1 <- 15
  L2 <- 25
  X_min <- 15
  X_max <- 25

  time_steps <- as.integer(T/dt)
  space_steps <- as.integer((X_max-X_min)/dx)+1

  vetS <- X_min + dx*(0:(space_steps-1))

  vetI <- X_min/dx + 0:(space_steps-1)

  a_i <- dt/4*(sigma^2*(vetI*dx)^(2*gam)/(dx^2)-r*vetI)
  b_i <- -dt/2*(sigma^2*(vetI*dx)^(2*gam)/(dx^2) + r)
  c_i <- dt/4*(sigma^2*(vetI*dx)^(2*gam)/(dx^2) + r*vetI)

  # A = [-a below, 1-b on diag, -c above ]
  Amatrix <- diag(1-b_i,space_steps)
  Amatrix[(row(Amatrix) - col(Amatrix)) == 1] <- -a_i[2:(space_steps)] # row > col, i.e. below diag
  Amatrix[(row(Amatrix) - col(Amatrix)) == -1] <- -c_i[1:(space_steps-1)] #row < col, i.e. above diag

  # B = [a below, 1+b on diag, c above]
  Bmatrix<-diag(1+b_i,space_steps)
  Bmatrix[(row(Bmatrix)-col(Bmatrix)) == 1 ]<- a_i[2:space_steps]
  Bmatrix[(row(Bmatrix)-col(Bmatrix)) == -1 ]<- c_i[1:(space_steps-1)]

  if(opttype=="cor"){
    V <- 1 - ((vetS>L2 | vetS<L1)*1)
  }
  else if(opttype=="call"){

  }

  g <- c(rep(0,space_steps-1),c_i[length(c_i)]*V[length(V)])
  g[1] = a_i[1]*V[1]
  for(k in 1:time_steps){
    V <- solve(Amatrix, Bmatrix %*% V)
  }
  return(V)
}

```

```

gamma <- 0.8
dt <- 0.01
options(scipen = 999)
for (dX in c(0.1, 0.05, 0.02)) {
  valX <- cnFd(dt, dX, gamma, "cor", T=1/2)
  print(valX[(10/dX)/2+1])
}

```

```

[1] 0.7688451
[1] 0.7625108
[1] 0.7586165

```

Compound Option

The Compound Option is priced like a Standard Call between $t = \frac{1}{4}$ and $T = \frac{1}{2}$, with strike 20. Let S be the underlying asset, with discretization grid between 0 and S_{max} . Between these $\frac{1}{4}$ and $\frac{1}{2}$, we have the terminal condition:

$$V\left(S, \frac{1}{2}\right) = \max(S - 20, 0) \quad (18)$$

and boundary conditions for all $t \in [\frac{1}{4}, \frac{1}{2}]$:

$$V(0, t) = 0 \quad (19)$$

$$V(S_{max}, t) = S_{max} - 20e^{-rt} \quad (20)$$

Denote this call option C_1 . The “mid-terminal” condition we have for the compound option at $t = \frac{1}{4}$ is:

$$V\left(S, \frac{1}{2}\right) = \max\left(C_1\left(S, \frac{1}{4}\right) - 2, 0\right) \quad (21)$$

and we have the same boundary condition for $S = 0$, and the upper bound now becomes:

$$V(S_{max}, t) = C_1\left(S, \frac{1}{4}\right) - 2e^{-rt} \quad (22)$$

```

gam<-0.8
dt <- 0.1
T = 1/2
L1 <- 0
K_end = 20
L2 <- 2*K_end
K_compound<-2
L2_compound<-2 *K_compound

options(scipen=999)

```

```

prices = c()
dx_size = c(0.1,0.05,0.02)
for(i in 1:length(dx_size)){
  dx = dx_size[i]
  v_call_End <-implicitfd(1/4,dt,dx,L1,L2,gam,opttype="call",K_end,0,20)$V
  v_total <- implicitfd(1/4,dt,dx,L1,L2,gam,opttype="compound",K_compound,v_call_End[,1],20)$price
  prices<-c(prices,v_total)
}

```