```
% LSM – Least Squares Montecarlo
% Método para valorar opciones americanas con Montecarlo
% Antonio Rivela 2008


1;


clear all;


%%%%%%%%%%%%%%parameters%%%%%%%%%%%%%%%%

S=30;

X=30;

T=5;

r=0.03;

dividend=0;

v=0.3; %volatility

nsimulations=2000;

nsteps=1000;   % 10

CallPutFlag="p";

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%AnalyAmerPrice=BjerkPrice(CallPutFlag,S,X,r,dividend,v,T)

r=r-dividend;

%AnalyEquropeanPrice=BlackScholesPrice(CallPutFlag,S,X,T,r,v)


if CallPutFlag=="c",

        z=1;

else

        z=-1;

end;




smat=zeros(nsimulations,nsteps);

CC=zeros(nsimulations,nsteps); %cash flow from continuation

CE=zeros(nsimulations,nsteps); %cash flow from exercise

EF=zeros(nsimulations,nsteps); %Excercise flag

dt=T/(nsteps-1);


smat(:,1)=S;
```

```
drift=(r-v^2/2)*dt;
vsqrdt=v*dt^0.5;
for i=1:nsimulations,
        st=S;
        curtime=0;
        for k=2:nsteps,
                curtime=curtime+dt;
                st=st*exp(drift+vsqrdt*randn);
                smat(i,k)=st;
        end
end


CC=smat*0; %cash flow from continuation
CE=smat*0;
EF=smat*0; %Excercise flag
st=smat(:,nsteps);
CE(:,nsteps)=max(z*(st-X),0);
CC(:,nsteps)=CE(:,nsteps);
EF(:,nsteps)=(CE(:,nsteps)>0);

paramat=zeros(3,nsteps); %coeff of basis functions

for k=nsteps-1:-1:2,
        st=smat(:,k);
        CE(:,k)=max(z*(st-X),0);

        %only the positive payoff points are input for regression
        idx=find(CE(:,k)>0);
        Xvec=smat(idx,k);
        Yvec=CC(idx,k+1)*exp(-r*dt);
        %Regress discounted continuation value at next time step
        % to S variables at current time step
        regrmat=[ones(size(Xvec,1),1),Xvec,Xvec.^2];

        p=ols(Yvec,regrmat);
        CC(idx,k)=p(1)+p(2)*Xvec+p(3)*Xvec.^2;
```

```
        %If exercise value is more than continuation value
        %choose to exercise
        EF(idx,k)=CE(idx,k) > CC(idx,k);
        EF(find(EF(:,k)),k+1:nsteps)=0;
        paramat(:,k)=p;
        idx=find(EF(:,k) == 0);
        %don't store regressed value of CC for next use
        CC(idx,k)=CC(idx,k+1)*exp(-r*dt);
        idx=find(EF(:,k) == 1);
        CC(idx,k)=CE(idx,k);
end

payoff_sum=0;

for i=1:nsteps,
        idx=find(EF(:,i) == 1);
        st=smat(idx,i);
        payoffvec=exp(-r*(i-1)*dt)*max(z*(st-X),0);
        payoff_sum=payoff_sum+sum(payoffvec);
end


MCAmericanPrice=payoff_sum/nsimulations


st=smat(:,nsteps);
payoffvec=exp(-r*(nsteps-1)*dt)*max(z*(st-X),0);
payoff_sum=sum(payoffvec);
MCEurpeanPrice=payoff_sum/nsimulations
```