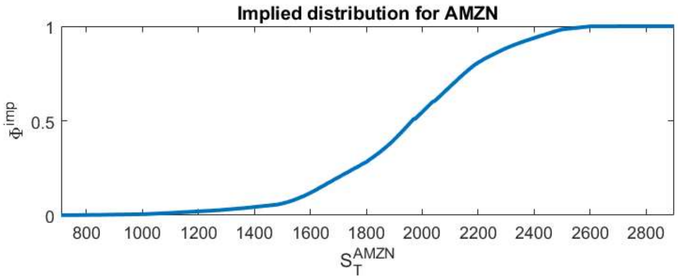
```
%======================================================================%
%========================== Advanced Derivatives ========================%
%============================== Problem Set 4 ===========================%
%======== BRODARD Lionel, MARCHAL Antoine, TISSOT-DAGUETTE Valentin ======%
%======================================================================%

close all; clear; clc; format short; warning('off')

rng(1331)
```

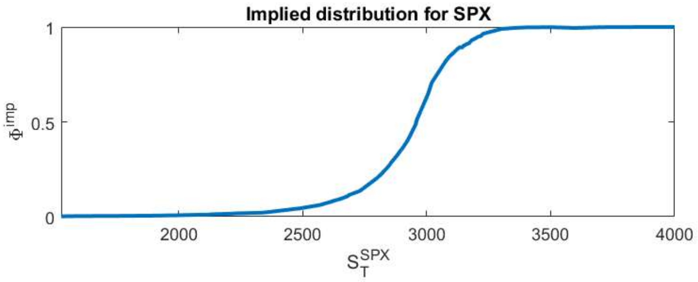
0. Setup

```
% Load the data
Table  = readtable("Impvols_SPX_AMZN.xlsx",'Range','A2:F280');

% Remove empty columns
Table  = removevars(Table ,{'Var3','Var4'});

% Remove the first row (empty)
Table = Table(2:end,:);

% Rename the columns (K: strikes, IV: ImpliedVols)
Table.Properties.VariableNames = {'SPX_K' 'SPX_IV' 'AMZN_K' 'AMZN_IV'};

% Parameters
T = 0.296; r = 0.024;

% Closing prices and dividend rates (amzn and spx respectively)
S_0 = [1971,2921]; delta = [0.019,0.018];

id = ["AMZN","SPX"];
```

I. Derive the marginal implied distributions

```
for i = 1:2

    % Rows where strikes and implied vols are available
    I = find(~ isnan(Table{: , id(i) + "_K"}));

    % Array of call prices for the different strikes (using BS_Price.m)
    C = exp(r*T) * BS_Price(S_0(i),Table{I , id(i) + "_K" },...
                        r , T, Table{I , id(i) + "_IV"},delta(i));

    % Strikes increments K_i - K_{i-1} for the Finite Difference Scheme
    D_K = diff(Table{I , id(i) + "_K" });

    % Use the first-order forward difference to derive the implied cdf
    Table{I(1:end-1),id(i) + "_cdf"} = 1 + diff(C)./ D_K;

    % Extrapolate the cdf by filling the end of the column with 1's
    Table{I(end):end,id(i)+"_cdf"} = 1;

    % Plot of the implied distribution
    subplot(2,1,i)
    plot(Table{:,id(i)+"_K"},Table{:,id(i)+"_cdf"},'Linewidth',2)
    xlabel(sprintf("S_T^{%s}",id(i)))
    ylabel("\Phi^{imp}")
    title(sprintf('Implied distribution for %s',id(i)))
    xlim([min(Table{:,id(i)+"_K"}),max(Table{:,id(i)+"_K"})])

end
```






```
% Display the head of the table
Table(1:20,:)
```

ans = 20�6 table

	SPX_K	SPX_IV	AMZN_K	AMZN_IV	AMZN_cdf	SPX_cdf
1	1525	0.4511	710	0.6033	0.0003	0.0019
2	1550	0.4440	720	0.5969	0.0004	0.0025
3	1575	0.4376	730	0.5911	0.0005	0.0028
4	1600	0.4315	740	0.5857	0.0006	0.0030
5	1625	0.4254	750	0.5806	0.0007	0.0032
6	1650	0.4193	760	0.5760	0.0008	0.0034
7	1675	0.4131	770	0.5716	0.0009	0.0036
8	1700	0.4069	780	0.5674	0.0010	0.0037
9	1725	0.4006	790	0.5634	0.0012	0.0038
10	1750	0.3941	800	0.5597	0.0014	0.0040
11	1775	0.3876	810	0.5563	0.0015	0.0042
12	1800	0.3811	820	0.5531	0.0017	0.0045
13	1825	0.3747	830	0.5499	0.0018	0.0047
14	1850	0.3682	840	0.5469	0.0020	0.0049
15	1875	0.3616	850	0.5439	0.0021	0.0050
16	1900	0.3550	860	0.5410	0.0023	0.0054
17	1925	0.3484	870	0.5382	0.0025	0.0058
18	1950	0.3419	875	0.5368	0.0028	0.0065
19	1975	0.3357	880	0.5356	0.0030	0.0069
20	2000	0.3294	890	0.5332	0.0032	0.0074

Monte Carlo and Gaussian Copula method

```
% Number of simulations
N_sim = 1e4;

% Inverse of the implied marginal distribution:
% (first index in the Table whose cdf value is above a given level x)
Impl_cdf_inv = @(x,id) Table{find(Table{:,id + "_cdf"} >= x,1), id + "_K"};

% Mean and covariance matrix for the multivariate Gaussian distribution
Mu = [0,0]; Sigma = [1,0.5;0.5,1];

% N_sim x 2 array for the inverse transform sampling
x = normcdf(mvnrnd(Mu,Sigma,N_sim));
```

```
% Array of terminal values for the stock prices
S_T = zeros(N_sim,2);

for i = 1:2
    for j = 1:N_sim

        % Use the inverse function defined above
        S_T(j,i) = Impl_cdf_inv(x(j,i),id(i));
    end
end

% Simulated payout values
H = max(S_T(:,2)/S_0(2) - S_T(:,1)/S_0(1),0);

fprintf('\nPrice of the exchange option %2.5f\n',exp(-r*T) * mean(H))
```

```
Price of the exchange option 0.05799
```