Assignment 1 Calibration of a single underlier model

Table of Contents

written by Peili Guo (peili.guo.7645@student.uu.se) and Fran (@studnet.uu.se) This report is for Computational Finance: Calibration and Estimation Assignment 1. we use Black-Scholes model to calibrate a European call option. In the first part, we compute the volatility from one observation. in the second part, we compute the volatility from several observation with different strike prices. call publish('A1_publish.m','format','pdf') to publish pdf file

Assignment 1, part 1

in this part, we compute volatility from a single observed call option and strike price, at T = 1. we have the value input listed in the function below, fzero is called to compute σ.

```
dbtype('function1.m');
```

```
%the result is
sigma = fzero(@function1,0)
```

```
1     function y_out = function1(sigma)
2
3     C = 23.50604; %option price
4     K = 100; %strike price
5     T = 1; %time
6     S0 = 110; %current asset price
7     r = 0.1; %interest rate
8     q = 0.01; %divident with continuous rate
9
10    d1 = 1/(sigma*sqrt(T))*(log(S0/K)+(r-q+0.5*sigma*sigma)*T);
11    d2 = 1/(sigma*sqrt(T))*(log(S0/K)+(r-q-0.5*sigma*sigma)*T);
12
13
14    y_out = normcdf(d1)*S0*exp(-q*T)-normcdf(d2)*K*exp(-r*T)-C;
15    end
```

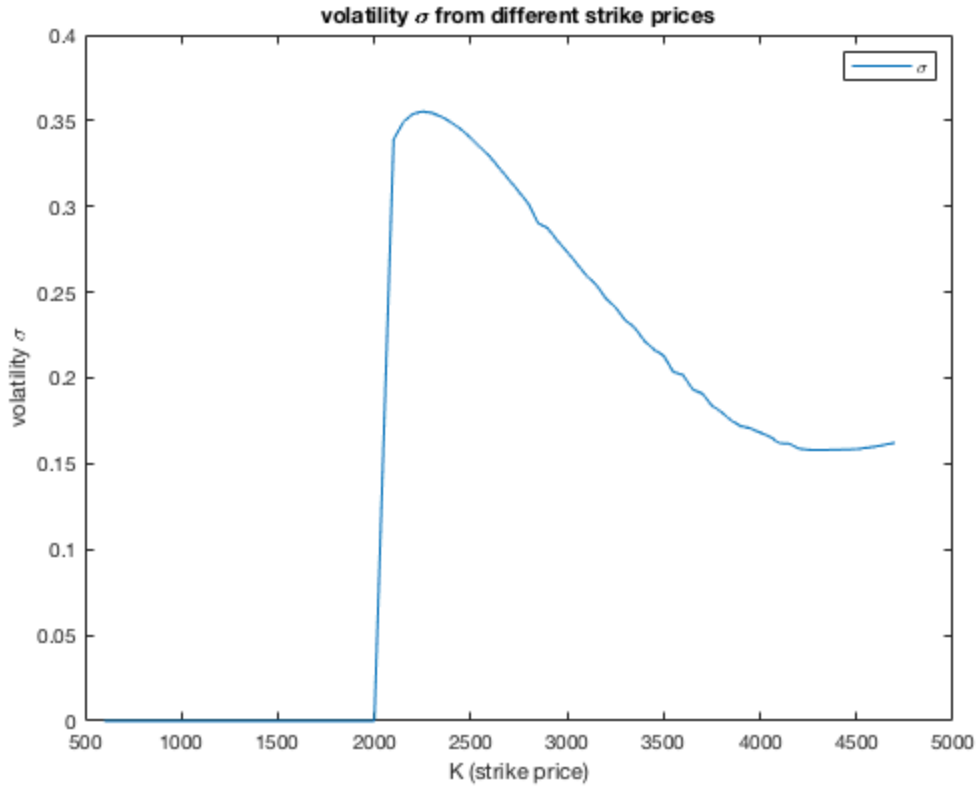
```
sigma =

   0.307197841638419
```

Assignment 1, part 2

in this program, we compute the volatility from real market data

```
format long
```

```
load('SX5E.mat'); %load market data
%sigma0 = 0;
%sigma = fzero(@function1,sigma0)
%function1(sigma)
%T = 1;

r = -0.0644; %interest rate

%d1 = 1/(sigma*sqrt(T))*(log(S0/K)+(r-q+0.5*sigma*sigma)*T);
%d2 = 1/(sigma*sqrt(T))*(log(S0/K)+(r-q-0.5*sigma*sigma)*T);

t = datenum(SX5E.t);
T = datenum(SX5E.T);

myfun = @(sigma,C,K,S0) normcdf(1/(sigma*sqrt((T-t)/365))*(log(S0/
K)+(r-0+0.5*sigma*sigma)*(T-t)/365))*S0*exp(-0*(T-t)/365)-normcdf(1/
(sigma*sqrt((T-t)/365))*(log(S0/K)+(r-0-0.5*sigma*sigma)*(T-
t)/365))*K*exp(-r*(T-t)/365)-C;

for i=1:70
    C = SX5E.C(i);
    K = SX5E.K(i);
    C0(i) = SX5E.C(i);
    K0(i) = SX5E.K(i);
    S0 = SX5E.S0;

    fun = @(sigma) myfun(sigma,C,K,S0);

    sigma(i) = fzero(fun,0);
end

%plot of result
plot(K0,sigma);
xlabel('K (strike price)');
ylabel('volatility \sigma');
title ('volatility \sigma from different strike prices');
legend('\sigma');
```



Published with MATLAB® R2018a