Project 3 by
Ashwin Kumar Ashok Kumar

Q1. Expected values for X2*Y2 is calculated for both the conditional and full. (Q1 d)

```
/Users/akumar/anaconda/bin/python /Users/akumar/Python/com/ashwin/computationalmethodsinfinance/Projecy3/Q1.py
Q1. a) P(Y2 > 5) = 0.979000                      ****[58.024408 sec]
Q1. b) Expected value of X2^(1/3) = 0.654199      ****[0.001206 sec]
Q1. c) Expected value of Y3 = 25.826058            ****[0.000276 sec]
Q1. d) Expected value of X2*Y2 , X2>1 = 3.805250   ****[0.002103 sec]
Q1. d) Expected value (conditional) of X2*Y2 | X2>1 = 14.359435    ****[0.002103 sec]

Process finished with exit code 0
```

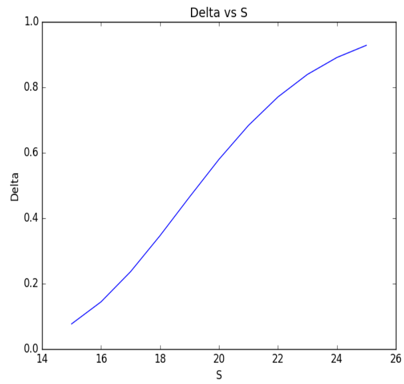
Q2.

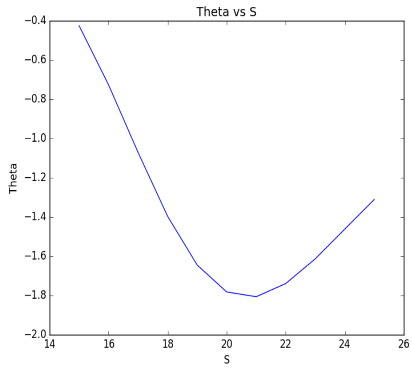
```
/Users/akumar/anaconda/bin/python /Users/akumar/Python/com/ashwin/computationalmethodsinfinance/Projecy3/Q2.py
Q2.  a) Expection of the expression = 1.315852                                        ****[3.951950 sec]
     b) Expected value of the function E[(1+Y)^(1/3)] = 1.344479 with var =0.149012     ****[0.098842 sec]

Process finished with exit code 0
```

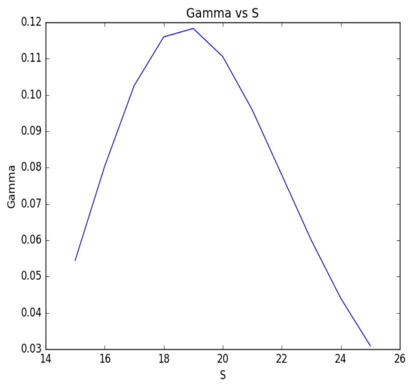
Q3.

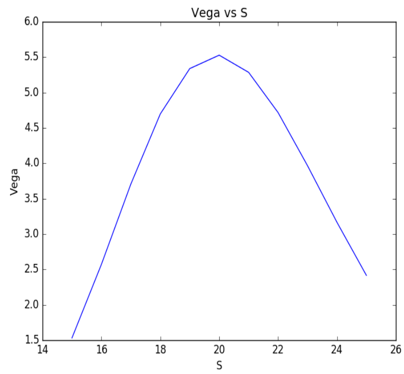
```
/Users/akumar/anaconda/bin/python /Users/akumar/Python/com/ashwin/computationalmethodsinfinance/Projecy3/Q3.py
Q3. a) MC Call Price for (S,k,r,sigma,t) =(100,80,0.04,0.30,5) = 42.198242 with 95% confidence in (39.242409,45.154075)
Q3. b) Black Scholes European Call Price for (S,k,r,sigma,t) =(100,80,0.04,0.30,5) = 42.946032
```

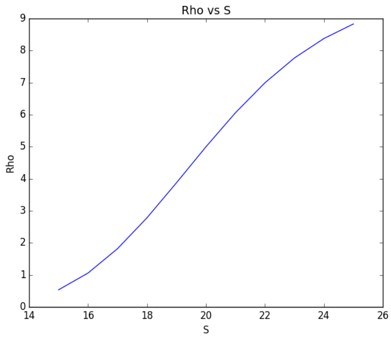
Greeks :











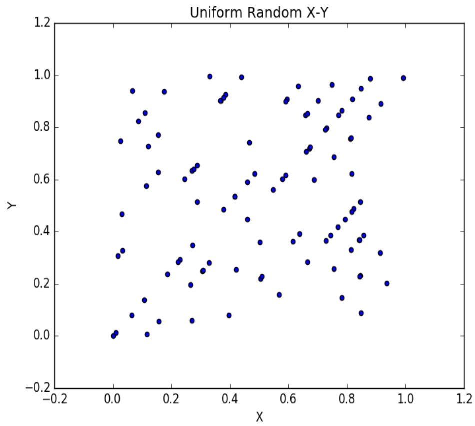
Q4.

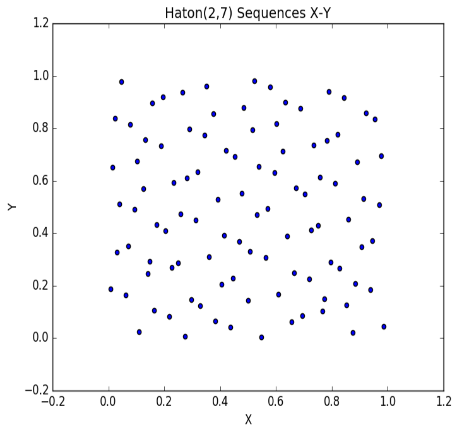
```
/Users/akumar/anaconda/bin/python /Users/akumar/Python/com/ashwin/computationalmethodsinfinance/Projecy3/Q4.py
Q4. European Call Option Prices and respective variances of MC Simulation  using stochastic volatalities :  ****[12.786032 sec]
    a) By Full Truncation = 2.623922 with var of MC = 0.017831
    b) By Partial Truncation = 2.623922 with var of MC = 0.017831
    c) By Reflection Method = 2.623922 with var of MC = 0.017831

Process finished with exit code 0
```

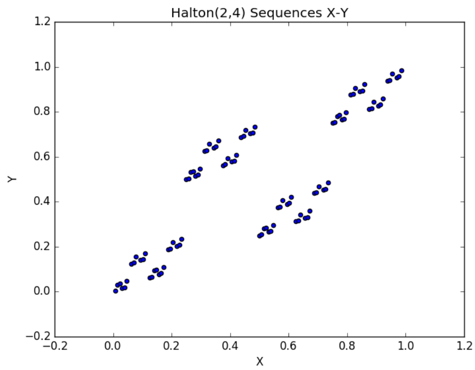
The European Call Option Price by the three estimates are the same. The reason is since we are using the same random number to generate the three sequences, the three methods will produce different estimated prices only when the variance goes negative. Since the simulation does not produce highly large values (far end in the tail) the variance did not go negative (for ndiv = 1000 and nsims = 1000 as well).

Q5. The area calculated using Halton number generation has very large variance.

```
/Users/akumar/anaconda/bin/python /Users/akumar/Python/com/ashwin/computationalmethodsinfinance/Projecy3/Q5.py
Q5.  e) The integral of the function using MC using Halton sequences:
        a) H(2,4) = -0.004884 with var = 0.883161
        b) H(2,7) = 0.026114 with var = 0.796011
        a) H(5,7) = 0.026164 with var = 0.026164
Q5.  e) The integral of the function using MC using Random sequences = 0.026559 with var = 0.790405

Process finished with exit code 0
```







The (2,4) graph does not produce completely random numbers which can be seen from the third graph. Other combinations of m's [(2,5), (2,7)] have uniformly dispersed scatter plot.