MFE 405 Project 3
Yong Jia (Justin) Tan

1. Expected Values of SDEs

To simulate processes X_t and Y_t, we need to first simulate two independent Wiener Processes, W_t and Z_t. We use a similar approach to our last project: we divide the total time interval *T* into *n* subparts. Then, we generate two sets of *n* independent random normally distributed values. Using these two sets, we can generate a path for W_t and Z_t. We can now use the simulated W_t path to simulate a path for X_t. We repeat this process *m* times to produce *m* distinct paths for X_t, and similarly, using Z_t we can product *m* distinct paths for Y_t.

Throughout this project, *n* (number of divisions in time interval T) is set to be 100. We also know that, assuming *c* is the computational budget, the following allocation is more efficient: $n \sim c^{1/3}, m \sim c^{2/3}$, which implies that $m = n^2$. Thus, we also set *m* to be 10,000.

In order to find $P(Y_2 > 5)$, we first simulate *m* values of Y_2, then count the fraction of the total amount that has a value larger than 5.

In order to find $E(X_2^{1/3})$, we first simulate *m* values of X_2, compute *m* values of $X_2^{1/3}$, and find the mean of all values. A similar method is used to compute the other expected values. The final computational results are in the table below:

	Prob	**E1**	**E2**	**E3**
Value	0.9705	0.6382	25.2872	3.8444

2. Expected Value of SDE and Stochastic Process

For computing $E(1 + X_3)^{1/3}$, we use a similar method to Part 1. However, for computing $E(1 + Y_3)^{1/3}$, we only need to simulate *m* values of Y_3 by computing m values of Z_t. We do not need to simulate entire paths in this case. The final computational results are in the table below:

	E1	**E2**
Value	1.3395	1.3396

3. European Call Option Pricing & Greeks Estimation

We compute the empirical price of a European call option using Monte Carlo simulation methods similar to our last project. We also attempt variance reduction by using antithetic variables. We generate normal distributions that are perfectly negatively correlated, and then use them to simulate Wiener processes, ultimately using them to compute possible prices of a European call option with lower variance (higher precision).

We can also compute European call option prices using the Black-Scholes formula:
$C(S_t, t) = N(d_1)S_t - N(d_2)Ke^{-r(T-t)}$, where
$d_1 = \frac{1}{\sigma\sqrt{T-t}}[ln(\frac{S_t}{K}) + (r + \frac{\sigma^2}{2})(T-t)]$, and
$d_2 = d_1 - \sigma\sqrt{T-t}$.
In order to approximate $N(\cdot)$, we use the following method:

$$N(x) = \begin{cases} 1 - \frac{1}{2}(1 + d_1x + d_2x^2 + d_3x^3 + d_4x^4 + d_5x^5 + d_6x^6)^{-16}, & if\ x \geq 0 \\ 1 - N(-x), & if\ x < 0 \end{cases}$$

With the following choices of d_i this method will have an accuracy of 10^{-7}:

$d_1 = 0.0498673470, \quad d_2 = 0.0211410061, \quad d_3 = 0.0032776263,$

$d_4 = 0.0000380036, \ d_5 = 0.0000488906, \ d_6 = 0.0000053830.$

Now we attempt to estimate the five "Greeks". Δ will be used as an example.

We know that $\Delta = \frac{\partial C}{\partial S}$, with *C* being the call option price. By setting a very small change in S_0, we can see the change in *C*. Setting a very small change ϵ, we can estimate:
$\Delta = \frac{\partial C}{\partial S} = \frac{C(S_0 + \epsilon) - C(S_0)}{\epsilon}$.

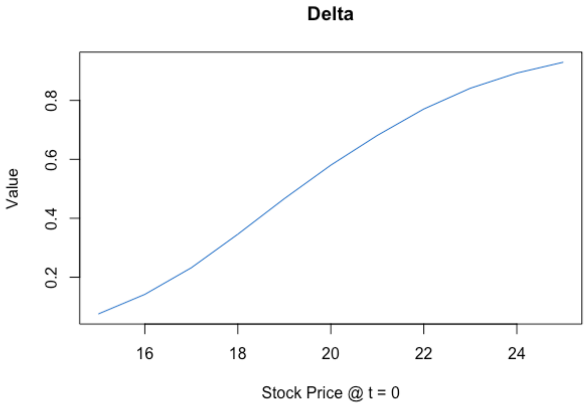
This method can be applied very similarly to find all the other "Greeks".

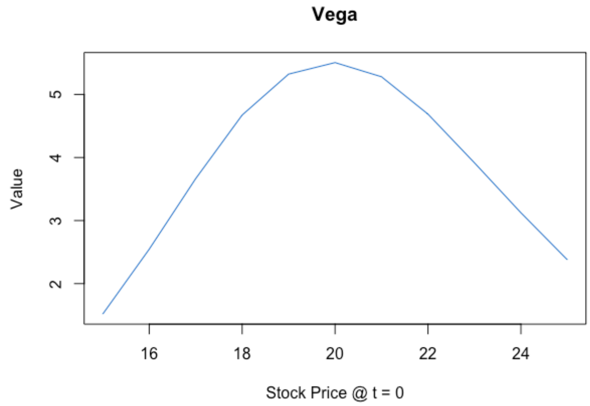
Note:

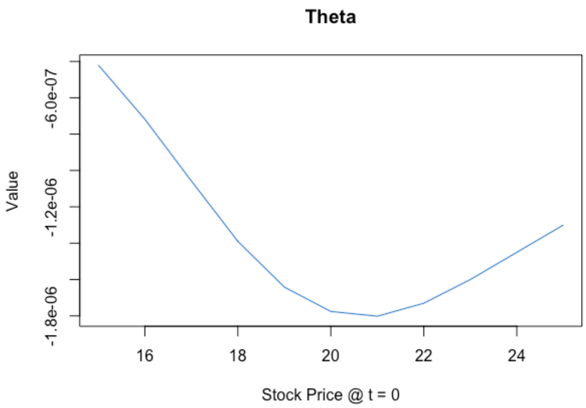
For Γ, we can write $\Gamma = \frac{\partial^2 C}{\partial S^2} = \frac{\partial \Delta}{\partial S}$, and thus estimate by seeing the change in Δ due to a small change in S_0.

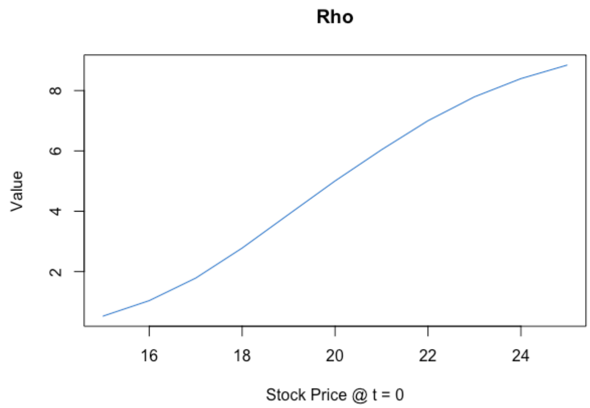
For Θ, since $\tau = T - t$, and $\Theta = \frac{\partial C}{\partial \tau}$, we estimate Θ by the following:
$\Theta = \frac{\partial C}{\partial \tau} = \frac{C(t+\epsilon) - C(t)}{(T - (t+\epsilon)) - (T - t))} = \frac{C(t+\epsilon) - C(t)}{-\epsilon}$.

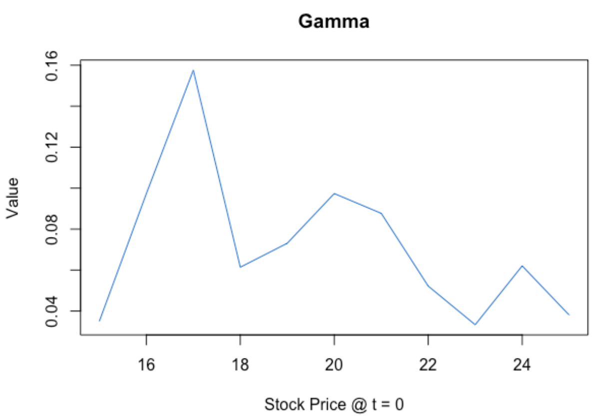
All plots for the “Greeks” are shown as below:











4. Heston 2-Factor Model

In order to simulate the 2-factor model, we need to first generate 2 random normal distributions Z_1, Z_2 with correlation ρ. Then we can use Z_1, Z_2 to simulate paths of W_t^1, W_t^2, and then use them to simulate V_t, S_t. We need to be careful as S_t depends on V_t.

After inputting the parameter values and setting stick price $K = 50$, time to maturity $T = 5$, we find the following results:

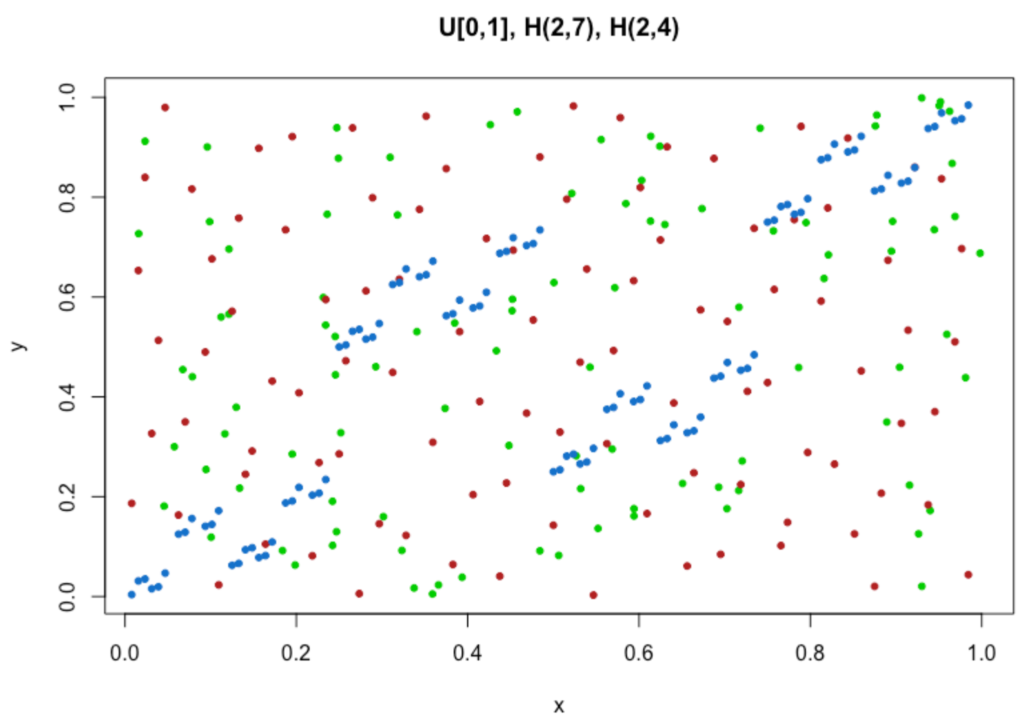
	Reflection	**Partial Truncation**	**Full Truncation**
Price	12.2383	12.8486	12.8477

We can see that the Full Truncation method seems to have the most precise result.

5. Quasi-Monte Carlo, Halton Sequences

First, we generate 100 2-dimensional vectors of Uniform [0,1] x [0,1] using the LGM method. Then, we generate 100 2-dimensional Halton sequences of base 2 and 7, as well as base 2 and 4. We generate a Halton sequence $\{H_1, H_2, \cdots, H_k, \cdots\}$ by the following:
For k, we write it in the form $k = a_0 + a_1 b + a_2 b^2 + \cdots + a_r b^r$, where $b^r \le k < b^{r+1}$.
Then we find $H_k = \frac{a_0}{b} + \frac{a_1}{b^2} + \cdots + \frac{a_r}{b^{r+1}}$.

After generating these three sequences, we can plot them in a graph to see their differences:



In this plot, the green dots are from the Uniform sequences, the red dots are from the Halton sequences of base 2 and 7, and the blue dots are from the Halton sequences of bae 2 and 4. As we can see, the Halton sequences of base 2 and 7 spread out the "unit square" of [1,1] most evenly, with the Uniform sequences doing a somewhat adequate job. However, the Halton sequences of base 2 and 4 have points clustered together very significantly. This is due to 2 and 4 having common factors. Thus, when we use Halton sequences, it would be best to use bases that are prime numbers to avoid this problem.

Using the same methods, we now create 2-dimensional Halton sequences (N = 10,000) to compute the integral. The results from using different bases are in the table below:

Bases	**(2,4)**	**(2,7)**	**(5,7)**
Integral Value	-0.004	0.0261	0.0262

As we can see, using bases (2,7) and (5,7) both gives us fine results, but using bases (2,4) gives us a very inaccurate result.