Options Pricing using Neural Networks

Vaishali Devanand

September 13, 2020

1 Introduction

The common stochastic processes used to model markets are

- Geometric Brownian Motion (GBM)

$$dS_t = \mu S_t dt + \sigma S_t dW_t \tag{1}$$

- Geometric Brownian Motion with stochastic arrival (GBMSA) [also called Heston stochastic volatility model]

$$dS_t = \mu S_t dt + \sqrt{v_t} S_t W_S(t)$$
$$dv_t = \kappa(\theta - v_t) + \sigma\sqrt{v_t} W_v(t)$$

- Variance Gamma (VG)

$$X(t; \sigma, \nu, \theta) = \theta\gamma(t; 1, \nu) + \sigma W(\gamma(t; 1, \nu)) \tag{3}$$

- Variance Gamma with Stochastic Arrival (VGSA)

$$Z(t) = \theta\gamma(Y(t); 1, \nu) + \sigma W(\gamma(Y(t); 1, \nu))$$
$$Y(t) = \int_0^t y(u)du$$
$$dy_t = \kappa(\eta - y_t)dt + \lambda\sqrt{y_t} dW_t$$

In this project, we will choose to look at early exercise premium for VG process using shallow neural network with 15 neurons/layer. In VG process, labels for European and American options are generated using FFT and CONV algorithm respectively.

2 Summary of observations

At first, we use Adam optimizer with random initialization and zero dropout.

Table 1: Random normal initialization, zero dropout rate, Adam optimizer

	Input generated using non-quasi approach				Input generated using quasi approach			
	w/o European Prices		with European Prices		w/o European Prices		with European Prices	
	1 Layer	2 Layers	1 Layer	2 Layers	1 Layer	2 Layers	1 Layer	2 Layers
MSE	3.40E-08	3.40E-08	2.90E-08	1.80E-08	3.80E-08	1.80E-08	3.80E-08	2.00E-08
$R\hat{2}$	96.7479	96.7418	97.2352	98.2970	96.2326	98.2145	96.1614	97.9922

To switch the optimizer from Adam to RMSProp, change the code from

```
optimizer=tf.train.AdamOptimizer(learning_rate=learning_rate)
```

to

```
optimizer=tf.train.RMSPropOptimizer(learning_rate=learning_rate)
```

Table 2: Random normal initialization, zero dropout rate, RMSProp optimizer

	Input generated using non-quasi approach				Input generated using quasi approach			
	w/o European Prices		with European Prices		w/o European Prices		with European Prices	
	1 Layer	2 Layers	1 Layer	2 Layers	1 Layer	2 Layers	1 Layer	2 Layers
MSE	1.25E-07	1.04E-06	1.49E-07	1.04E-06	1.70E-07	1.01E-06	2.10E-07	1.17E-06
$R\hat{2}$	87.8790	-0.6762	85.5855	-1.1132	82.9448	-0.8790	78.8912	-17.4749

We can see that MSE and R-square is better for Adam compared to RMSProp optimizer. Now,we will switch the initialization from random normal to random uniform.

Table 3: Random uniform initialization, zero dropout rate, RMSProp optimizer

	Input generated using non-quasi approach				Input generated using quasi approach			
	w/o European Prices		with European Prices		w/o European Prices		with European Prices	
	1 Layer	2 Layers	1 Layer	2 Layers	1 Layer	2 Layers	1 Layer	2 Layers
MSE	1.20E-06	4.33E-07	1.22E-06	1.87E-07	2.58E-07	3.11E-07	3.13E-07	2.41E-07
$R\hat{2}$	-18.0384	58.0262	-18.0384	81.8399	74.1410	68.7844	68.6104	75.7937

We can see that random normal performs better than random uniform both in terms of MSE and R-square.

Now, we will switch from random normal to He initialization for Adam optimizer by simply multiplying random initialization with

$$\sqrt{2/size^{[l-1]}}$$

$W[l] = np.random.randn(size_l, size_(l-1)) * np.sqrt(2/size_l - 1)$

We can see that He initialization has better MSE and R-square compared to random initialization. Next, we will switch from random normal to He initialization for Adam optimizer by simply multiplying

Table 4: He initialization, zero dropout rate, Adam optimizer

	Input generated using non-quasi approach				Input generated using quasi approach			
	w/o European Prices		with European Prices		w/o European Prices		with European Prices	
	1 Layer	2 Layers	1 Layer	2 Layers	1 Layer	2 Layers	1 Layer	2 Layers
MSE	4.00E-08	4.50E-08	3.30E-08	1.60E-08	3.50E-08	1.90E-08	3.60E-08	1.80E-08
$R\hat{2}$	96.1610	95.6580	96.7606	98.4531	96.4633	98.0700	96.4233	98.2001

random initialization with

$$\sqrt{1/size^{[l-1]}}$$

$W[l] = np.random.randn(size_l, size_(l-1)) * np.sqrt(1/size_l - 1)$ and is commonly used for tanh() activation function.

Table 5: Xavier initialization, zero dropout rate, Adam optimizer

	Input generated using non-quasi approach				Input generated using quasi approach			
	w/o European Prices		with European Prices		w/o European Prices		with European Prices	
	1 Layer	2 Layers	1 Layer	2 Layers	1 Layer	2 Layers	1 Layer	2 Layers
MSE	3.80E-08	1.90E-08	3.20E-08	2.30E-08	4.90E-08	2.40E-08	3.70E-08	2.00E-08
$R\hat{2}$	96.3313	98.1549	96.9194	97.7678	95.1195	97.6001	96.3173	97.9986

The final step is to change the dropout rate to 0.5 to avoid overfitting.

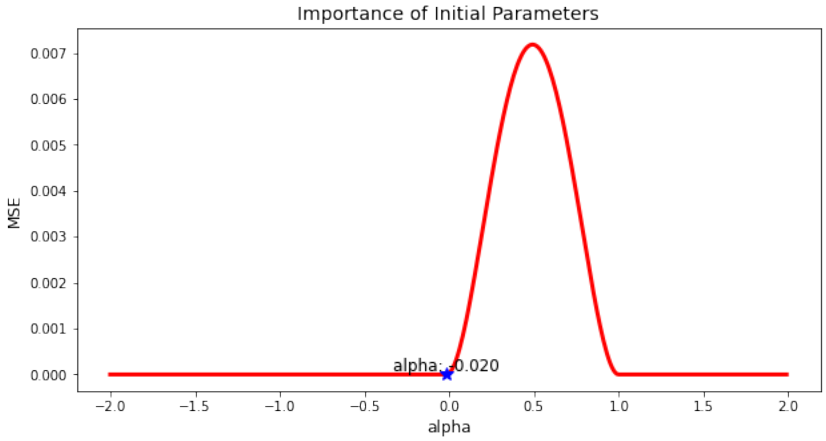

Figure 1: Random uniform initialization

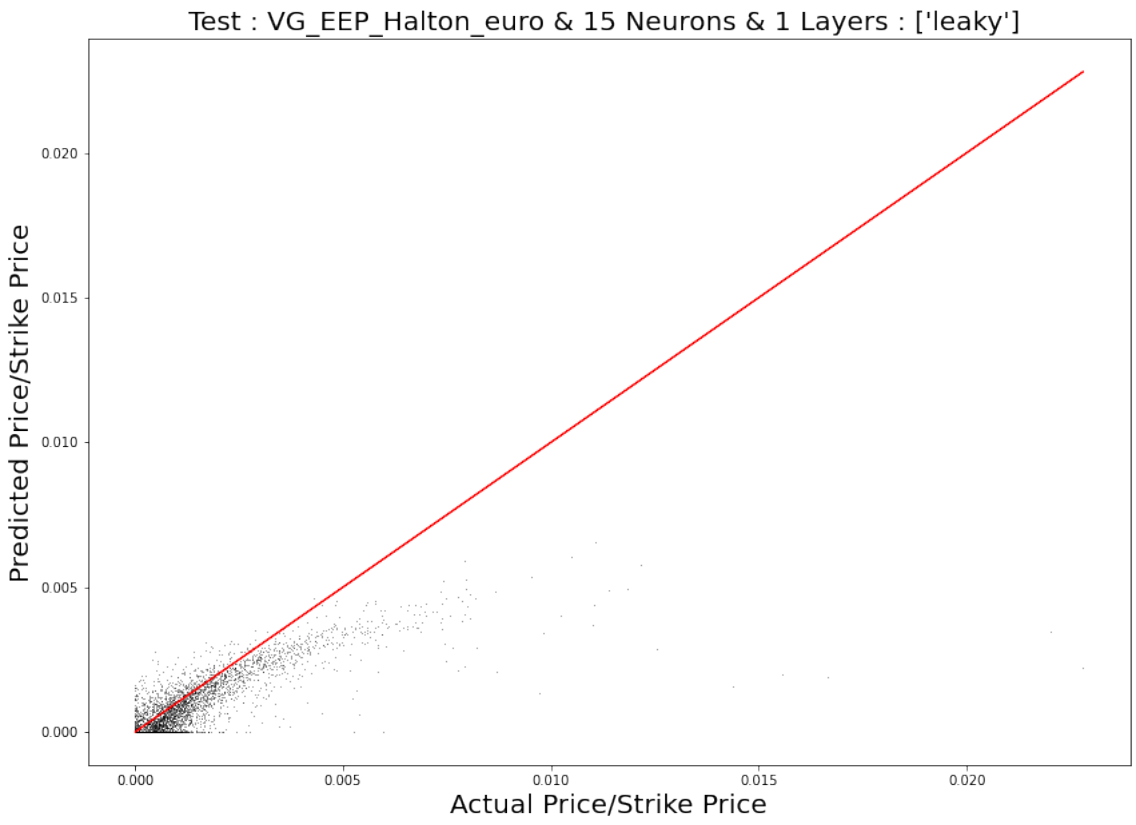

Figure 2: Random uniform initialization

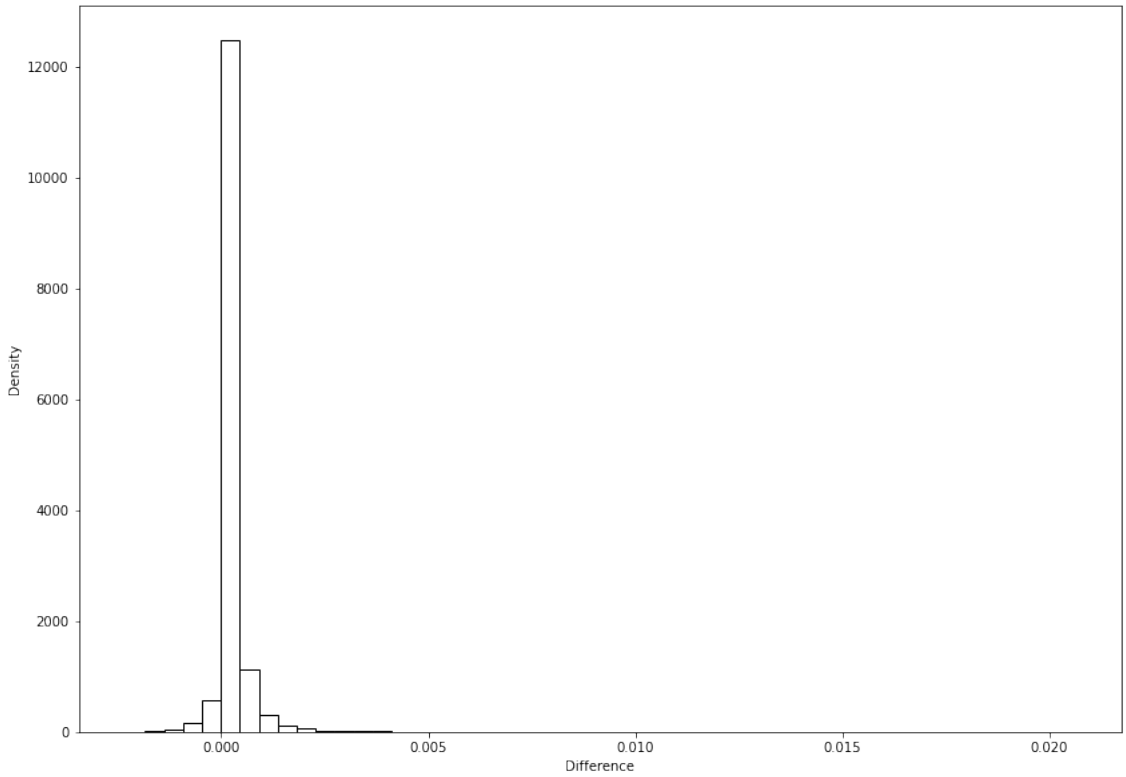

Figure 3: Random uniform initialization

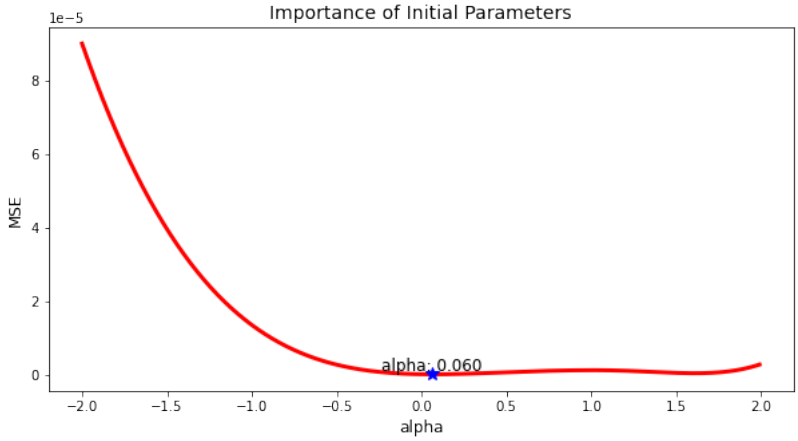

Figure 4: He initialization

3 Result

Below is the final table with best MSE and R-square from the above routines:

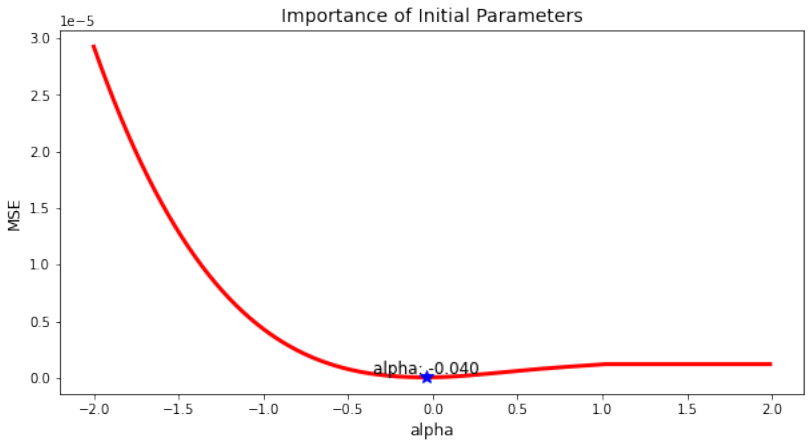

Figure 5: Xavier initialization

Table 6: Final

	Input generated using non-quasi approach				Input generated using quasi approach			
	w/o European Prices		with European Prices		w/o European Prices		with European Prices	
	1 Layer	2 Layers	1 Layer	2 Layers	1 Layer	2 Layers	1 Layer	2 Layers
MSE	3.40E-08	1.90E-08	2.90E-08	1.60E-08	3.50E-08	1.80E-08	3.60E-08	1.80E-08
$R\hat{2}$	96.7479	98.1549	97.2352	98.4531	96.4633	98.2145	96.4233	98.2001