Barrier *Reverse Convertible* Callable CSGN.SE

Credit Suisse Group AG

Group 7
Agarwala Pratham
Choy Xin Yun
Ong Yu Lin, Jocelyn

Product Description

Denomination

CHF 1000

Early Redemption

22 February 2023
22 May 2023

1 Year Period

15 Aug 2022 to 17 Aug 2023

Barrier Event

Occurs when underlying asset falls to 50% of the initial price

2.5% Coupon Quarterly

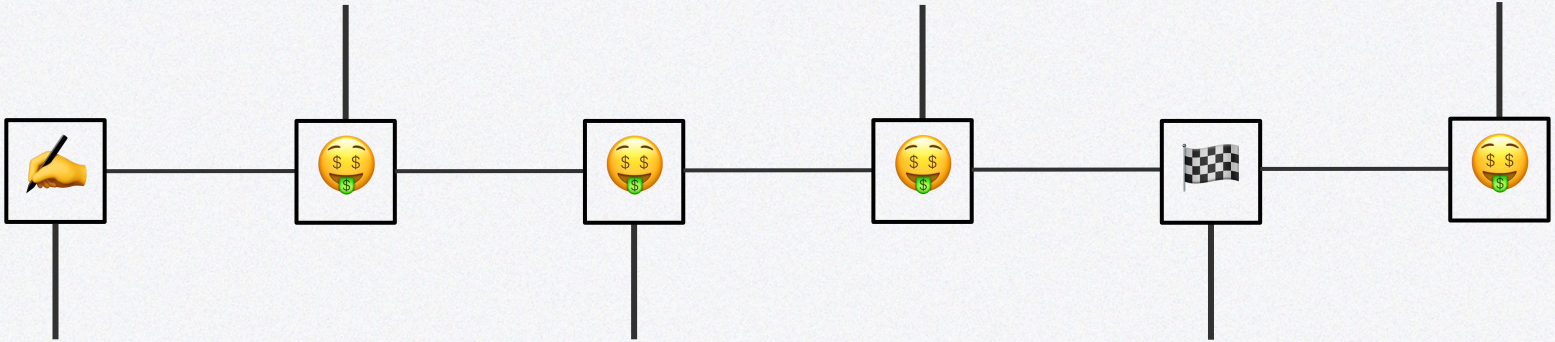
22 Nov 2022
22 Feb 2023
22 May 2023
22 Aug 2023

[Fact Sheet Available Here]

Scenario Analysis

At maturity / Throughout its lifetime	**Closes above initial level**	**Closes below initial level**
Barrier not reached	100% of denomination and 10% coupon	100% of denomination and 10% coupon
Barrier Reached	100% of denomination and 10% coupon	Receive shares according to conversion ratio and 10% coupon

Timeline



Initial Fixing Date
15 August 2022

1st Coupon Payment
22 November 2022

2nd Coupon Payment & Optional Early Redemption Date
22 February 2023

3rd Coupon Payment & Optional Early Redemption Date
22 May 2023

Final Fixing Date
17 August 2023

4th Coupon Payment
22 August 2023

Product Chart

Underlying Chart

1

Modelling of asset price without callable feature

Methods

1

Standard Monte-Carlo

2

Variance Reduction

Antithetic Variates
Control Variates
EMS

3

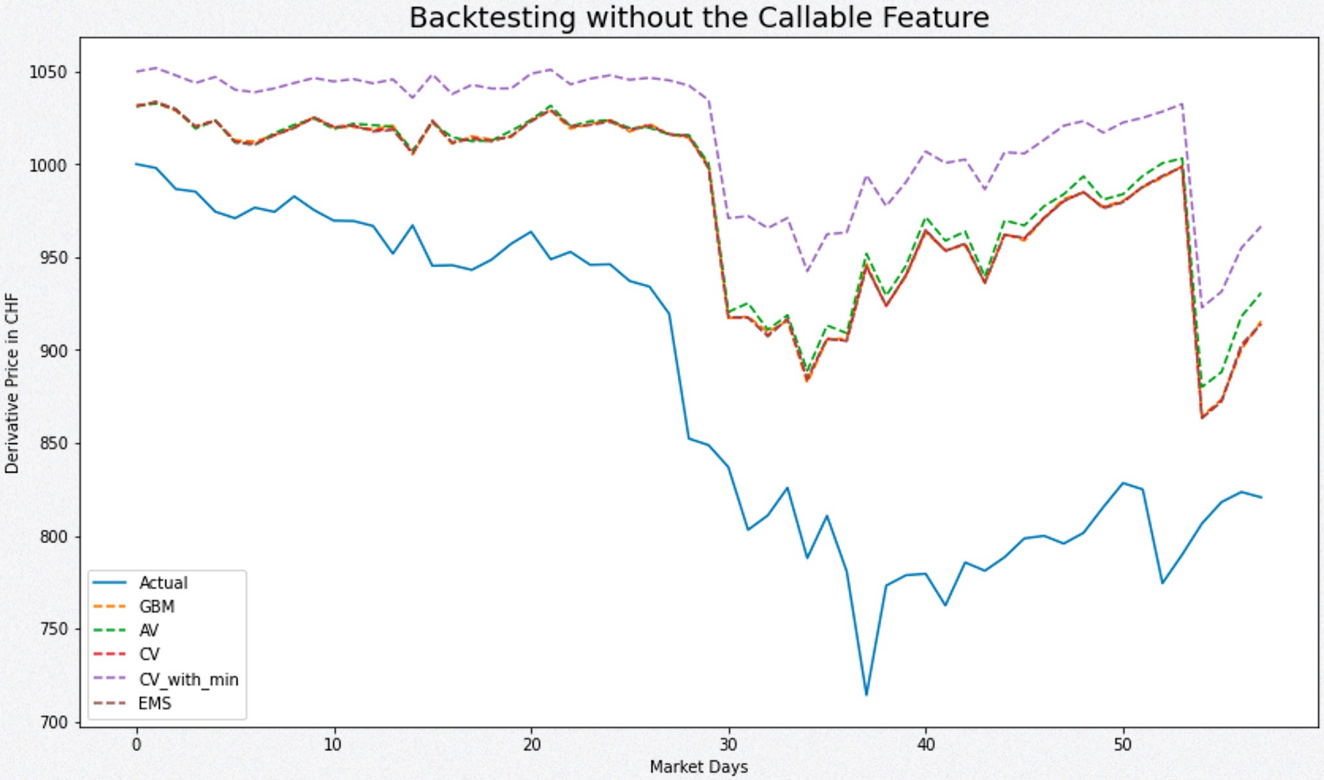
Empirical Martingale Correction

4

Sensitivity

Delta δ and gamma Γ

Derivative Price



Comparing Variance Reduction Techniques

- We ran 10000 simulations to calculate variance for each technique.
- We ensured same Randomness is being used to evaluate each method

Variance Reduction Technique	**Variance**
Without Variance Reduction	1.90189
Antithetic Variates	1.90411
Control Variate using Final Level as Control	**1.673986**
Control Variate using Min Level as Control	**1.776044**
Empirical Martingale Correction	**1.89823**

2

Modelling of asset price with callable feature

Methods

1

Standard Monte-Carlo

Compare with the MC without callable feature

2

Variance Reduction

By adding premium from MC

3

Empirical Martingale Correction

4

Sensitivity

Delta δ and gamma Γ

Calculating product price at time 0

Maturity date

t=1

- Calculate the payoff at t=1 with the simulated paths

2nd Early redemption date

t=0.75

- Regress the discounted payoff at t=1 on simulated prices at t=0.75
- Calculate redemption cost, R

1st Early redemption date

t=0.5

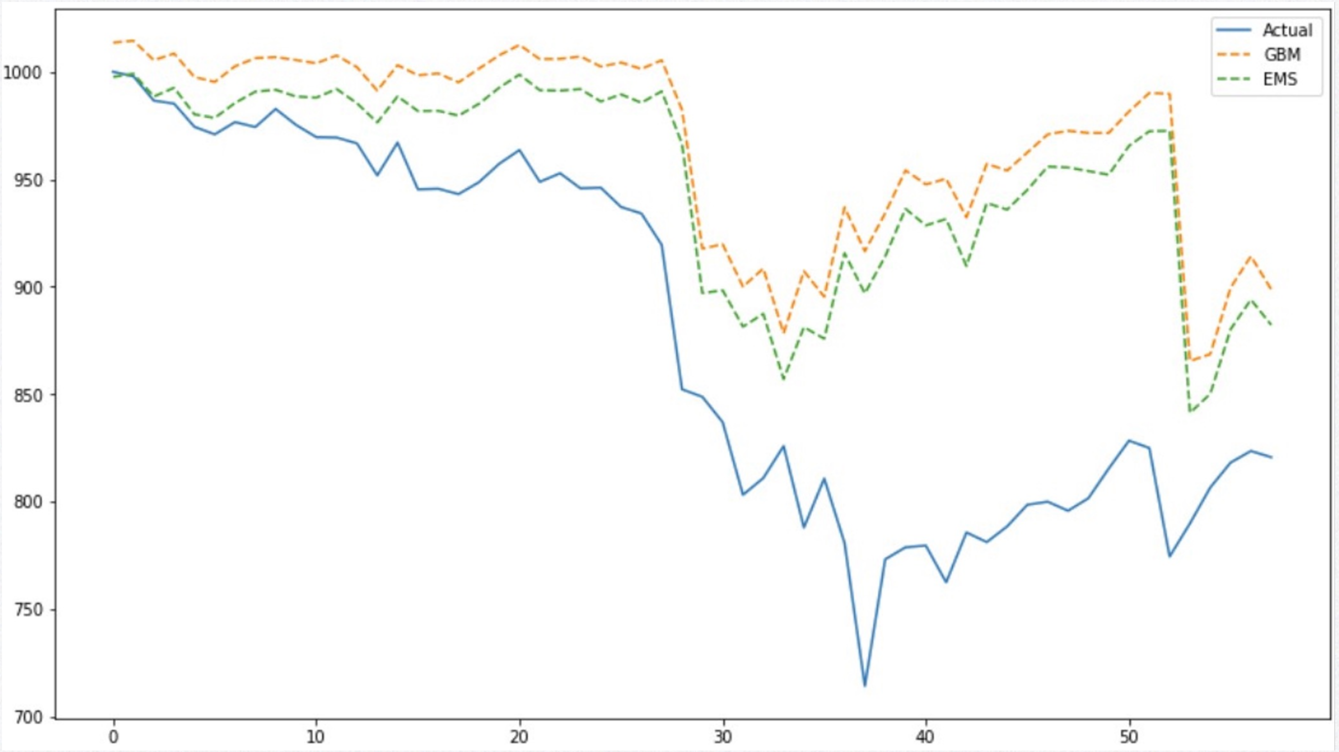
- Regress the discounted payoff at t=0.75 on simulated prices at t=0.5
- Calculate redemption cost, R

Initial date

t=0

- Discount the payoff at t=0.5 to t=0

Derivative Price



Error Analysis

```
Errors over a period of 60 Market Days(15 Aug - 4 Nov)

Mean Absolute Error for GBM : 102.31710 CHF
Mean Squared Error for GBM : 13610.4809 CHF

Mean Absolute Error for EMS : 102.2150 CHF
Mean Squared Error for EMS : 13594.361 CHF
```

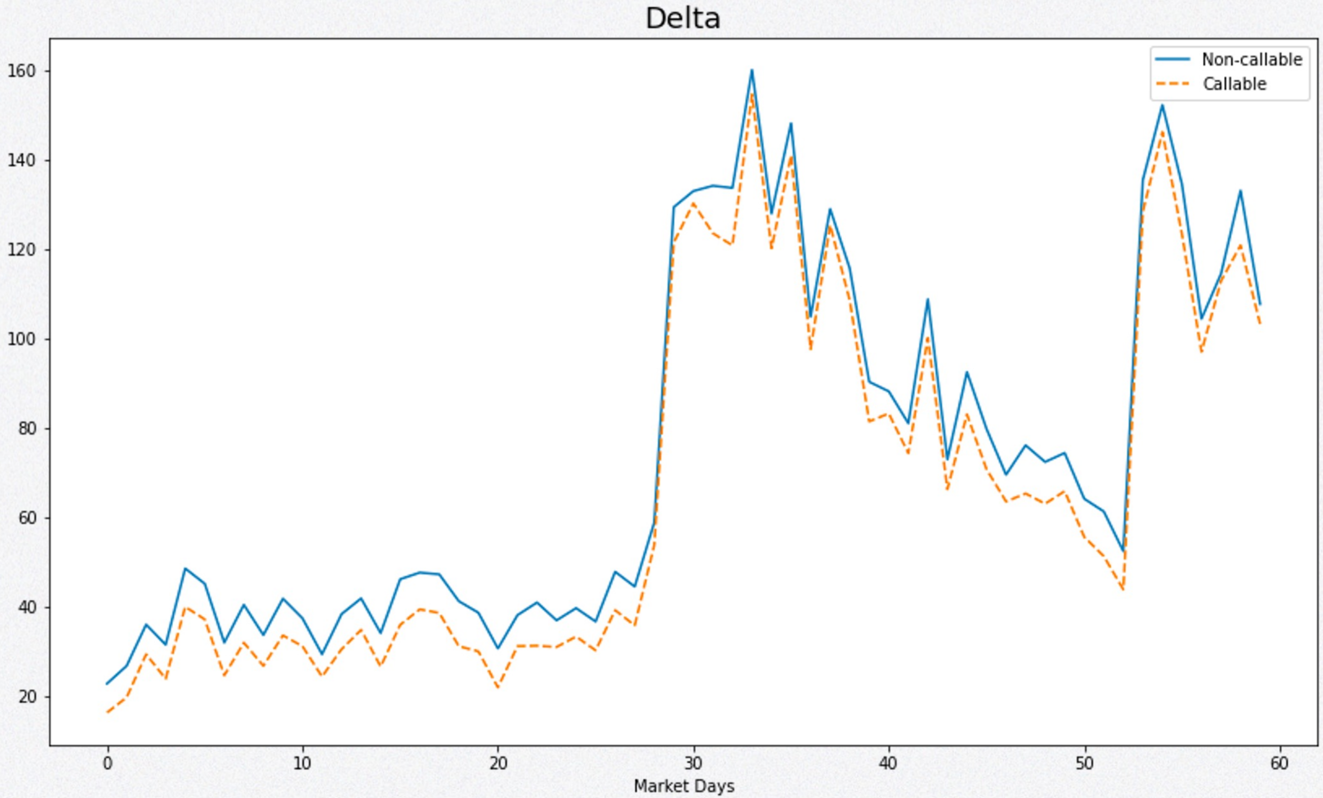
```
Using GBM the difference between Callable and Non-callable Estimation of Derivation Price => 11.1733 CHF

Using EMS the difference between Callable and Non-callable Estimation of Derivation Price => 11.1790 CHF
```

	Actual	GBM	EMS
0	1000.0	999.446117	999.081514
1	997.9	1000.950393	1001.577184
2	986.6	997.004455	997.773096
3	985.2	988.858037	988.896271
4	974.4	992.128308	992.220319

3

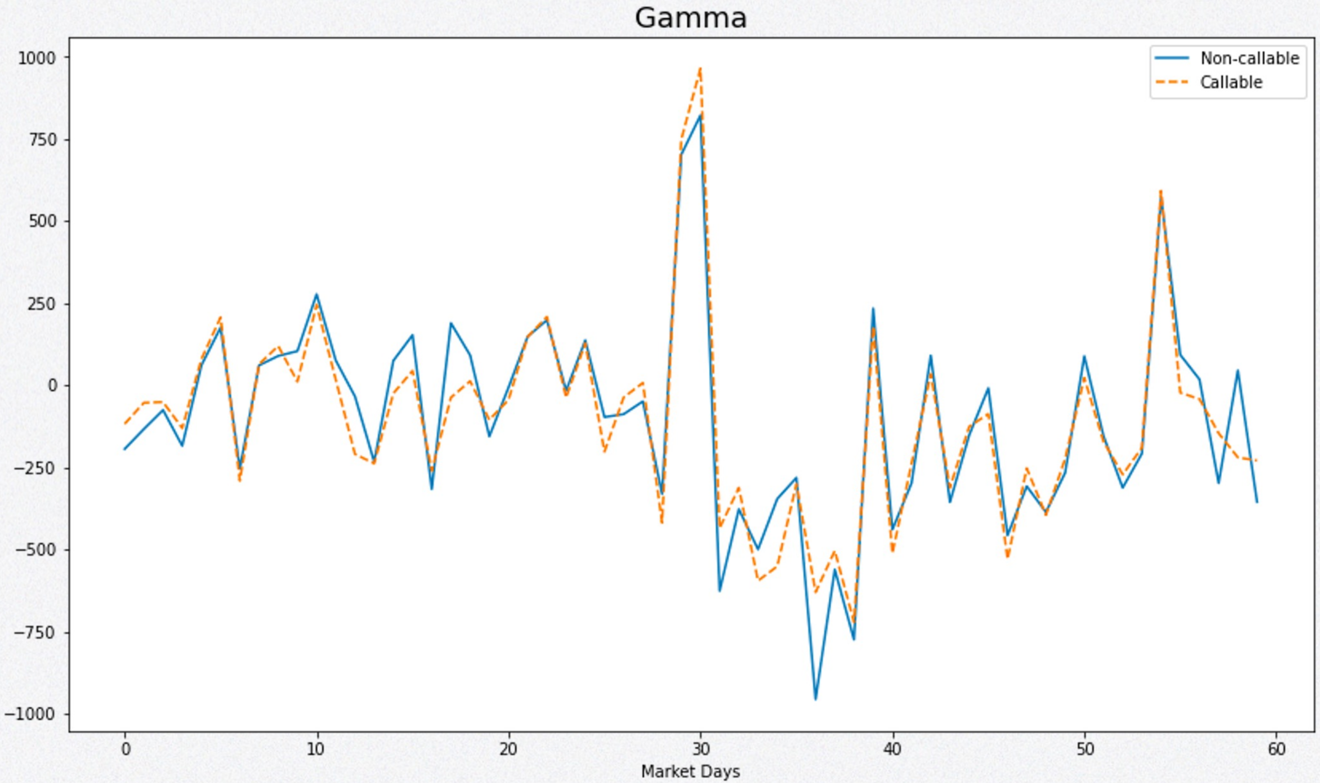
Estimating Sensitivities

Delta, Γ



Delta
Non-callable
Callable
160
140
120
100
80
60
40
20
0
10
20
30
40
50
60
Market Days

Gamma, δ



Gamma
Non-callable
Callable
1000
750
500
250
0
−250
−500
−750
−1000
0
10
20
30
40
50
60
Market Days

4

Modelling of asset price with short rate

Cox-Ingersoll-Ross (CIR) Model

The CIR model is used to model the interest rate movements. The model is a representation of a “one factor model” as it describes the interest rate movements driven by a sole source of market risk

$$dr_t = \kappa(\gamma - r_t)d_t + \sigma\sqrt{r_t}dW_t$$

$\kappa\ (kappa): continuous\ drift$
$\gamma\ (gamma): long\ term\ mean$
$\sigma\ (sigma): continuous\ volatility$

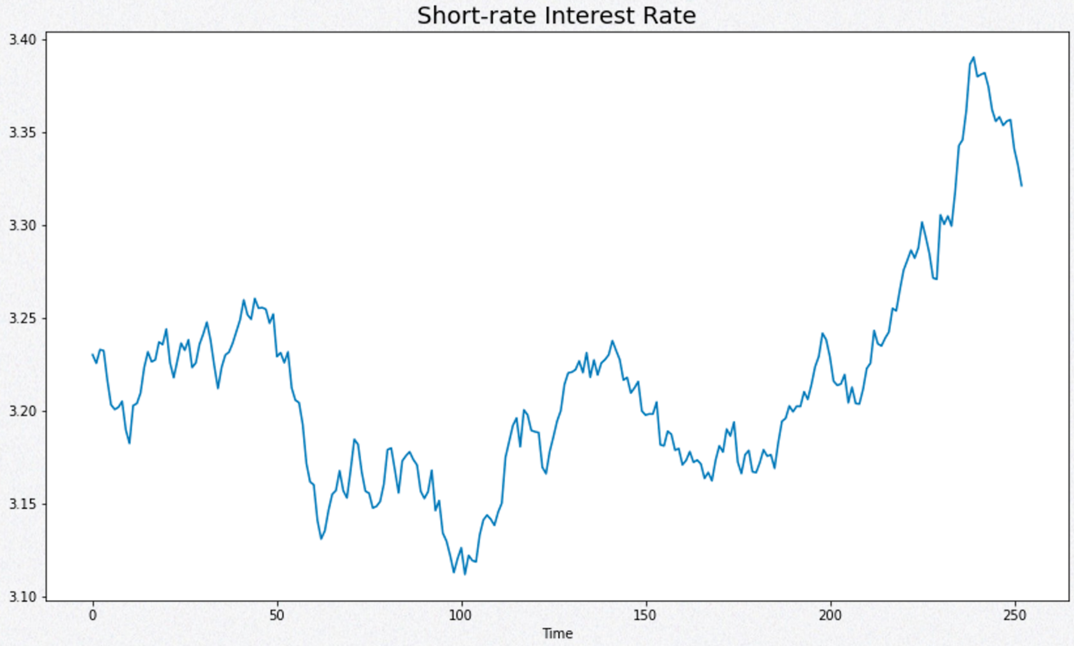
Calibration of CIR Model

1. Retrieve historical US Daily Treasury Par Yield Curve Rates
2. Set γ as the average interest rate
3. Calculate the sum of residual terms (RSS)

$$RSS = \frac{(r_t - \phi r_{t-1})^2}{r_t + \gamma}$$

1. Minimising RSS with respect to discrete drift, Φ
2. Calculate κ = log(1/Φ)
3. Calculate the discrete volatility, σa from RSS
4. Calculate σ from discrete volatility
5. Repeat steps 2-7 for a rolling window of 60 days

Short-rate Interest Rate



Expected Derivative Price

CHF 979.66

Variance

1.93

5

Modelling of asset price with local volatility

Methodology

1. Collect options data

Options that are available on that date

2. Calculate implied volatility, time to maturity (TTM) and Moneyness (ATM)

3. Spline fitting

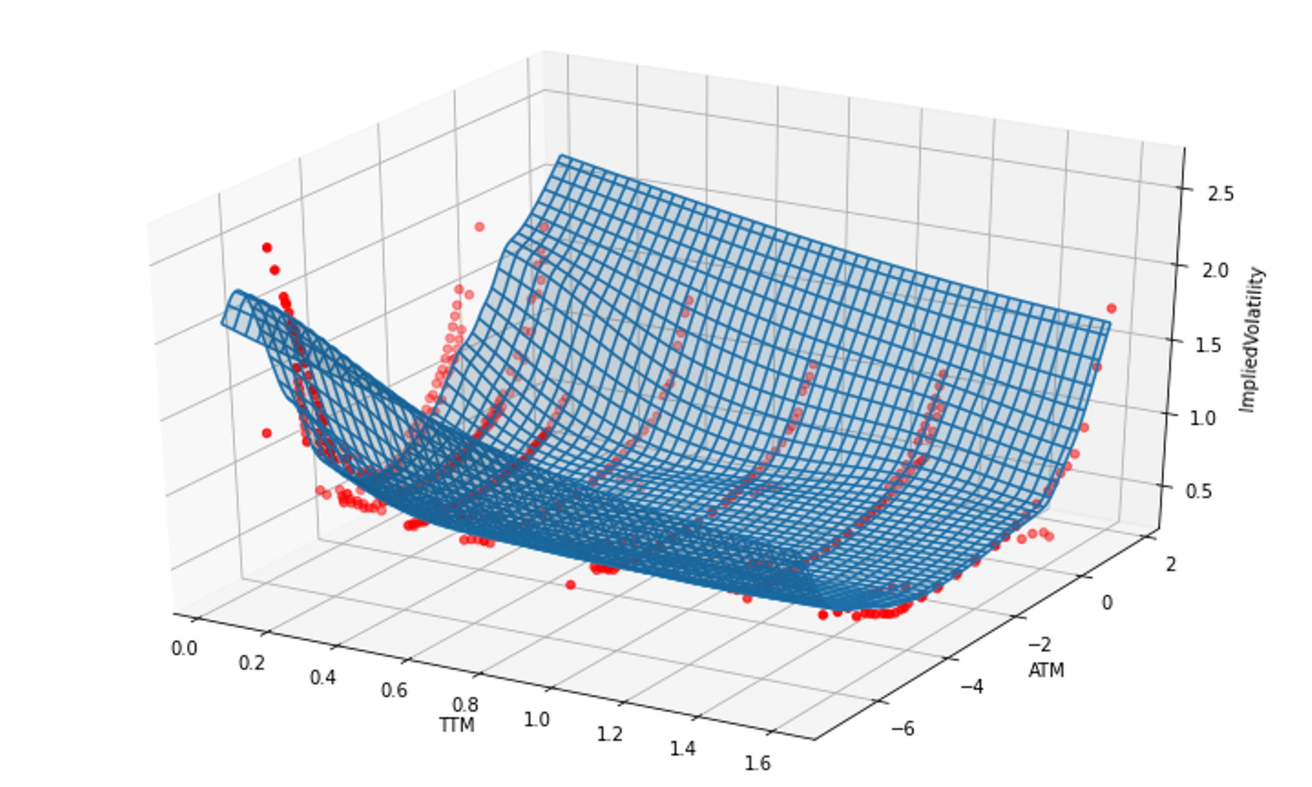
Using interpolation

4. Use fitted spline to determine local volatility at t, St

Using Dupire's formula

5. Simulate using standard MC

Spline Fitting



Using Dupire's Formula

Local Volatility will be dependent on stock price St and time t

$$d\ln S_t = \left(\mu - \frac{\sigma_{LV}^2(S_t,t)}{2}\right)dt + \sigma_{LV}(S_t,t)dW_t$$

$$\sigma_{LV}^2(S_t,t) = \frac{\frac{\partial C}{\partial T} + \mu K \frac{\partial C}{\partial K}}{\frac{K^2}{2}\frac{\partial^2 C}{\partial K^2}} \bigg|_{K=S,T=t}$$

Determining the Derivative Price

1. Simulate price paths using the formula:

$$S_{j+1} = S_j \ * \exp(\mu * dt + \frac{\sigma_{LV}^2(S_j, j) * dt}{2} + \sigma_{LV}(S_j, j) dt Z_{j+1})$$

1. Use the algorithm for calculating the price of a callable product

MC price	LV price	Actual price
910.5	678.35	826.6

6

Modelling of asset price with Heston model

Heston Model

The heston model is defined by a system of SDEs, to describe the movement of asset prices, where an asset's price and volatility follow random, Brownian motion processes. It is a stochastic volatility model introduced in early 1990s.

Under Risk Neutral Measure(Q) :

$$dS_t = rS_t dt + \sqrt{v_t} S_t dW_{S,t}^{\mathbb{Q}}$$

$$dV_t = \kappa^{\mathbb{Q}}(\theta^{\mathbb{Q}} - v_t)dt + \sigma^{\mathbb{Q}}\sqrt{v_t} dW_{v,t}^{\mathbb{Q}}$$

Note: It provides closed form solution for European Call and put options

How to Implement Heston Model

Monte-Carlo Estimate using Heston Model has two steps:

1. **Calibrate the Model Parameters**

1. **Use the parameters to simulate the assets prices using the Euler Approximation of Heston's Formula**

Euler Approximation

$$S_{t+dt} = S_t e^{\left(\left(r - \frac{1}{2} v_t\right) dt + \sqrt{v_t dt} Z_S\right)}$$

$$v_{t+dt} = v_t + \kappa(\theta - v_t) dt + \sigma \sqrt{v_t dt} Z_v$$

How to Find Euler Discretization

Pseudo Code for Heston Asset Pricing

1. i=0
2. Generate Standard Normal Variables Z_s and Z_v using correlation between dW_s and dW_v provided.
3. Above can be done by generating two Standard Normal variable Z_1 and Z_2 and set

$$Z_v = Z_1$$

$$Z_s = \rho * Z_1 + Z_2 * (1-\rho^2)^{½}.$$

1. Set S^i using the Euler Discretization formula.
2. Set V^i using the Euler Discretization formula.
3. Repeat steps 3-6 for M times where M is number of time units (Ex. 252 days for daily simulation of 1 year).
4. Repeat 1-6 for the amount of SImulations N.

Why We Calibrate

In practice, we try to obtain the parameters for the Heston model based on the prices we observe in the markets.

These parameters are then used in pricing complex/exotic options. For very complex options, we can’t really find a closed form formula for pricing exotic options.

For our case we are using the plain vanilla call options to find parameters of the Heston model and use the parameters we obtain from the market to calculate the price of the exotic options using Monte Carlo.

How to Calibrate

- Get Historical Options Data from Bloomberg as Yahoo Finance only provides current option prices
- Use the option data to create Volatility Surface.
- Get the US Treasury Bond Yield rate and perform Interpolation to get rates for any specific time.
- Use the Last Traded Prices as S_o and perform optimization using Least Square Estimate to obtain the parameters.

strike	0.5	2.5	3.5	4.0	4.5	5.0	5.5	6.0	8.0
0.019165	1.80	1.650	0.725	0.225	0.05	0.025	0.050	0.050	0.175
0.038330	0.00	1.725	0.725	0.300	0.10	0.025	0.025	0.025	0.050
0.057495	0.00	1.725	0.775	0.350	0.15	0.025	0.075	0.075	0.050
0.076660	0.00	1.725	0.775	0.350	0.15	0.050	0.025	0.075	0.050
0.095825	0.00	1.725	0.775	0.425	0.25	0.075	0.050	0.075	0.050
0.114990	3.70	1.700	0.825	0.425	0.25	0.100	0.050	0.050	0.050
0.134155	3.70	1.700	0.825	0.425	0.25	0.100	0.050	0.050	0.050
0.210815	3.75	1.725	0.975	0.550	0.30	0.175	0.125	0.050	0.025
0.364134	3.65	1.800	1.025	0.725	0.50	0.350	0.275	0.175	0.075
0.613279	3.65	1.850	1.200	0.925	0.70	0.525	0.450	0.300	0.100
1.207392	3.65	2.025	1.475	1.225	1.00	0.825	0.700	0.300	0.300
2.203970	3.65	2.275	1.775	1.500	1.35	1.175	1.050	0.300	0.300

Fig: Example of the Volatility surface created using the options data from 7th Nov

Options Data Available Here

Interpolated US Treasury Bond Curve

We need the interpolated curve so that we can calculate rates for any particular date corresponding to maturity date of our option.

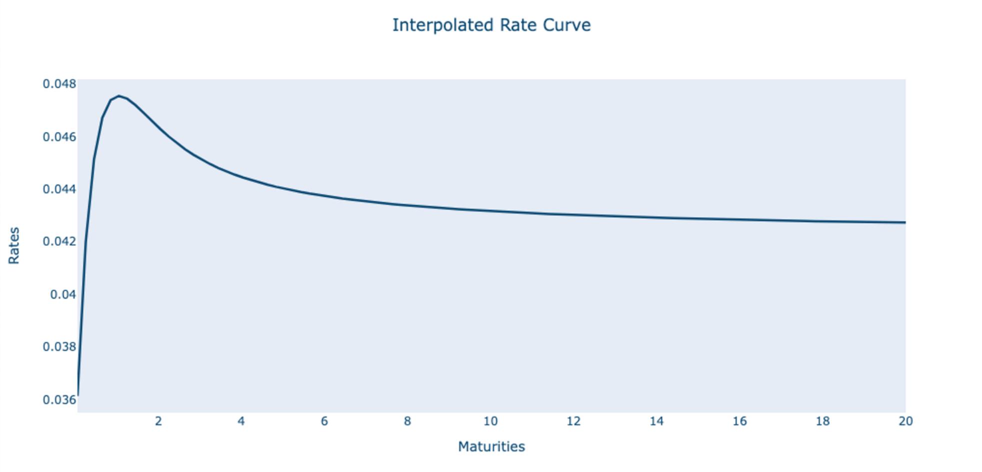


Fig: Example Rate Interpolated Curve

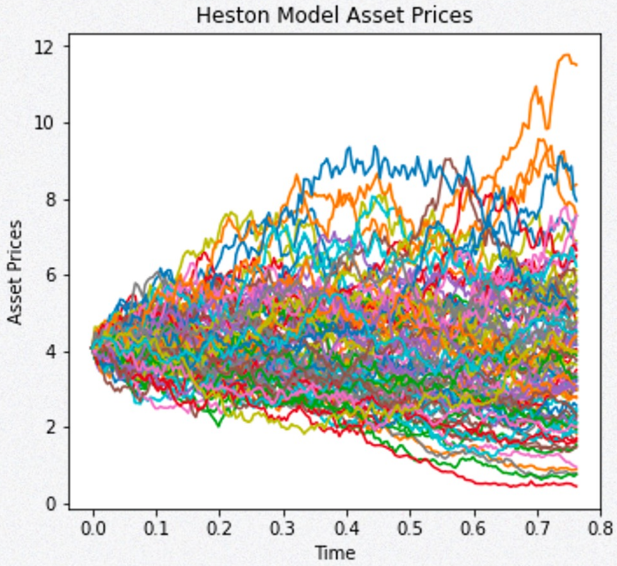
[US Treasury Yield Data Available here]

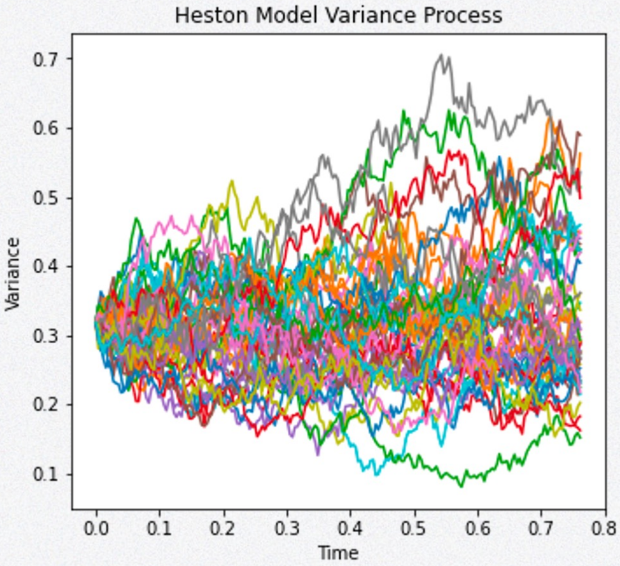
Bounds of the Parameters for the Optimization

Parameters	Initial Value	Lower Bound	Upper Bound	Results
σ - volatility of volatility	0.5	0.001	1	**0.39975**
Θ - long-term price variance	0.5	0.001	0.5	**0.30284**
K - rate of mean reversion to the long term price variance(Kappa)	5	1	100	**2.97761**
ρ - correlation between $dW_{S,t}$ and $dW_{v,t}$	-0.5	-1	1	**-0.72355**
v_0 - Initial variance.	0.5	0.001	0.5	**0.318257**

Note: We are using LMFIT library optimization with 10000 Maximum Iterations and minimum change in variable in each iteration is set to 0.001

Bounds of the Parameters for the Optimization (on 8 Nov)





The Price estimate on 8 Nov using Heston model without callable

849.0663

The Price estimate on 8 Nov using Heston model with callable

840.5764

Bounds of the Parameters for the Optimization (on 8 Nov)

Actual Price on 8 November 2022 - 826.6

Additional: Speeding up the Simulation Process

We leveraged the use of parallel thread to speed up the process of Simulation on Python and Efficient just-in-time compilers to tailor the bytecode to our system capabilities

We are using Numba package with JIT decorator which uses the LLVM compiler library to generate a machine code version of your function, tailored to your CPU capabilities.

Python Threading library to creates threads and assign them to parallel workers.

Results : **It decreased the runtime execution by 60-70%. After Numba and parallel processing it takes 10 seconds to generate 1M simulations for GBM process.**

Numba Package Available Here

Appendix: Euler Discretization of Heston

$$v_{t+dt} = v_t + \int_t^{t+dt} \kappa(\theta - v_u)\,du + \int_t^{t+dt} \sigma\sqrt{v_u}dW_{2,u}. \qquad (9)$$

The Euler discretization approximates the integrals using the left-point rule

$$\begin{aligned}
\int_t^{t+dt} \kappa(\theta - v_u)\,du &\approx \kappa(\theta - v_t)\,dt \\
\int_t^{t+dt} \sigma\sqrt{v_u}dW_{2,u} &\approx \sigma\sqrt{v_t}\,(W_{t+dt} - W_t) \\
&= \sigma\sqrt{v_t dt}Z_v
\end{aligned}$$

where Z_v is a standard normal random variable. The right hand side involves $(\theta - v_t)$ rather than $(\theta - v_{t+dt})$ since at time t we don't know the value of v_{t+dt}. This leaves us with

$$v_{t+dt} = v_t + \kappa(\theta - v_t)\,dt + \sigma\sqrt{v_t dt}Z_v.$$

By Itō's lemma $\ln S_t$ follows the diffusion

$$d\ln S_t = \left(r - \frac{1}{2}v_t\right)dt + \sqrt{v_t}dW_{1,t}$$

or in integral form

$$\ln S_{t+dt} = \ln S_t + \int_0^t \left(r - \frac{1}{2}v_u\right)du + \int_0^t \sqrt{v_u}dW_{1,u}.$$

Euler discretization of the process for $\ln S_t$ is thus

$$\begin{aligned}
\ln S_{t+dt} &= \ln S_t + \left(r - \frac{1}{2}v_t\right)dt + \sqrt{v_t}\,(W_{1,t+dt} - W_{1,t}) \qquad (10) \\
&= \ln S_t + \left(r - \frac{1}{2}v_t\right)dt + \sqrt{v_t dt}Z_s.
\end{aligned}$$

Hence the Euler discretization of S_t is

$$S_{t+dt} = S_t \exp\left(\left(r - \frac{1}{2}v_t\right)dt + \sqrt{v_t dt}Z_s\right).$$

Appendix: Euler Discretization of Heston

Start with the initial values S_0 for the stock price and v_0 for the variance. Given a value for v_t at time t, we first obtain v_{t+dt} from

$$v_{t+dt} = v_t + \kappa\left(\theta - v_t\right)dt + \sigma\sqrt{v_t dt}Z_v$$

and we obtain S_{t+dt} from

$$S_{t+dt} = S_t + rS_t dt + \sqrt{v_t dt}S_t Z_s$$

or from

$$S_{t+dt} = S_t \exp\left(\left(r - \frac{1}{2}v_t\right)dt + \sqrt{v_t dt}Z_s\right).$$

To generate Z_v and Z_s with correlation ρ, we first generate two independent standard normal variable Z_1 and Z_2, and we set $Z_v = Z_1$ and $Z_s = \rho Z_1 + \sqrt{1-\rho^2}Z_2$.

Thank you!

Any questions?