Computational Finance, 2023/2024

April exam

Leonardo Risca

ID: 0900084051

Group:

De Pietro Jacopo 0001091710

Lunati Marco 0001080489

Risca Leonardo 0900084051

Volatility Surface

Abstract

This document reports the results for the Computational Finance final assignment for the April intake. The assignment here is to build volatility surfaces implied by the Monte Carlo call option prices based on three different models: the Constant Elasticity of Variance, the Displaced Diffusion and the Heston models. One will find this report divided into two main parts: the first one provides insights into the three models and technics used to perform Monte Carlo simulations to compute (call) option prices. The second part reports the results for the volatility surfaces implied by Monte Carlo call option prices as well as those obtained using the correponding (quasi) analytical solutions[1].

[1]For the computation of CEV analytical option prices we used the PyFENG python package (https://github.com/PyFE/PyFENG).

Contents

Part I

Monte Carlo setup

1 Model insights

1.1 The Constant Elasticity of Variance model

The CEV model is described by the following SDE

$$dS(t) = \sigma_X S^{\beta}(t) dW_t, \ \beta > \frac{1}{2}$$
$$S(t_0) = S(0)$$

Where β represents the elasticity of volatility: it indicates the sensitivity of volatility to changes in the underlying price, the degree of non-constant volatility in the model. A value of $\beta = 1$ implies that volatility remains constant over time (BSM model). If $\beta > 1$, volatility increases with increases in the asset price, indicating "leverage effect" where higher asset prices result in higher volatility. Conversely, if $\beta < 1$, volatility decreases as the asset price increases.

The Monte Carlo simulations performed to compute call option prices follow the Euler integration scheme for the path of the underlying. With $S(t) = S_t$, that is

$$S_{t_{n+1}} = S_{t_n} exp(-\frac{1}{2}\sigma^2 S_n^{2(\beta-1)} dt + \sigma S_n^{\beta-1} dW_{t_n}), \ dW_{t_n} \sim N_{0,\sqrt{dt}} \tag{1}$$

In cases where the CEV model incorporates a *beta* parameter of less than 1, indicating decreasing volatility with increasing asset prices, the underlying price will surely approach zero over time. In such situations, it becomes crucial to introduce a cutoff point ϵ for the asset price, to prevent it from reaching zero. Choosing an appropriate value for ϵ involves balancing the need to prevent unrealistic asset price behavior with computational efficiency and model accuracy. The cutoff value should be small enough to prevent the asset price from reaching zero but large enough to avoid introducing excessive numerical instability into the model. Practitioners may determine the cutoff value through empirical analysis, sensitivity testing, or calibration to market data. In this context, ϵ is set to 1% of the current underlying price.

$$\epsilon = S_0(.01)$$

From a practical point of view, calculating Monte Carlo option prices requires the computation of N trajectories integrated over $0 = t_0 < t_1 < ... < t_N = T$ for the underlying asset according to eq(1), considering that, for each t_n, if the new-step underlying price $S_{n+1} < \epsilon$ then $S_{n+1} = \epsilon$. The N option payoffs for each of the underlying path are then used to calculate the MC option price and its statistical error.

1.2 The Displaced Diffusion model

The SDE for the Displaced Diffusion model is

$$dY(t) = \sigma Y(t) dW_t$$
$$Y(t) = S(t) + \Delta$$
$$Y(0) = S(0) + \Delta$$

Where Δ represents the displacement of the diffusion process from its initial level and can be thought of as a shift or offset applied to the standard diffusion process. The displacement parameter Δ affects the behavior of asset prices by changing their starting point or initial level.

Defined $Y(t) = Y_t$,

$$Y_{t_{n+1}} = Y_{t_n} exp(-\frac{1}{2}\sigma^2 dt + \sigma dW_{t_n}),\ dW_{t_n} \sim N_{0,\sqrt{dt}} \tag{2}$$

Therefore, we implement Monte Carlo simulations as we would do for the BSM model, which implies generating N trajectories integrated over $0 = t_0 < t_1 < ... < t_N = T$ for the underlying asset according to eq(2). The only difference is that the starting point for the underlying is shifted of an amount equal to Δ. The N option payoffs for each of the underlying path are then used to calculate the MC option price and its statistical error.

1.3 The Heston model

We recall the the system of equations that define the Heston model:

$$\begin{aligned}
&S(t) = S(0)e^{X(t)} \\
&dX(t) = -\frac{v(t)}{2}dt + \sqrt{v_t}dW_t,\ X(0) = 0 \\
&dv(t) = \lambda(\bar{v} - v(t))dt + \eta\sqrt{v(t)}dY_t,\ v(0) = \sigma^2 \\
&dW_t = \rho dY_t + \sqrt{1-\rho^2}dZ_t
\end{aligned}$$

In order to simplify notation we define $S(t) = S_t$, $v(t) = v_t$, $X(t) = X_t$. With the Euler scheme we have

$$\begin{aligned}
S_{t_{n+1}} &= S_{t_n} e^{X_{t_{n+1}} - X_{t_n}} \\
X_{t_{n+1}} - X_{t_n} &= -\frac{v_{t_{n+1}} - v_{t_n}}{2}dt + \sqrt{v_{t_{n+1}} - v_{t_n}}(\rho\xi_Y + \sqrt{1-\rho^2}\xi_Z),\ \xi_Y, \xi_Z \sim N_{0,1} \\
v_{t_{n+1}} - v_{t_n} &= +\lambda(\bar{v} - v_{t_n})dt + \eta\sqrt{v_{t_n}}\xi_Y \\
v_{t_n} &= (v_{t_n})^+
\end{aligned}$$

Therefore, we proceed by generating NV trajectories integrated over $0 = t_0 < t_1 < ... < t_N = T$ for the volatility v_{t_n}. Secondly, for each of the volatility trajectory, we generate NS trajectories over $0 = t_0 < t_1 < ... < t_N = T$ for the underlying asset. Here it is important to remind that each block of $NS|nv_i$ cannot be considered fully independent. Thus we have to proceed by first calculating the average option payoff for each of the $NS|nv_i$ simulations with $i = 1, ..., NV$ and then using these NV variables to compute the Monte Carlo option price as well as the statistical error.

2 Variance Reduction

Reducing the Monte Carlo error can entail significant time and computational resources. Therefore, we present here two simple technics used to reduce the MC error without the need to increase the number of simulations.

2.1 Call Put Parity

Generally speaking, we know there are two ways of cumputing option prices. For instance, we can compute call option prices either via the call option formula or via the call put parity (i.e. computing put option prices and use them in the CP Parity to compute call option prices). In the context of Monte Carlo that means we can either discount the call average payoff or use the CP Parity with the put discounted average payoff.

Implementing MC for out of the money options, rather than in of the money options, will result in a lower number of positive payoffs which will contribute to the statistical error. As a result, the MC error for OTM options will be smaller than the MC error for ITM options.

Therefore, the MC call prices used to extract the implied volatility are computed as follows:

$$C = \begin{cases} e^{-rT}E[(S(T)-k)^+], & if\, k \geq e^{(r-q)T}S(0) \\ e^{-rT}E[(k-S(T))^+ + e^{-qT}S(0) - e^{-rT}k] & if\, k < e^{(r-q)T}S(0) \end{cases}$$

since $e^{-qT}S(0) - e^{-rT}k$ is deterministic and does not contribute to the statistical error.

2.2 Control variates

We can further improve the accuracy of the estimate of the average of our target random variable by using a correlated auxiliary random variable that shares with our target the property of having the same expected value.

In practice, we can use the deviations of the simulated underlying price at maturity from their mean to eliminate part of the fluctuations of the simulated discounted payoffs.

Defined

$$X_n = e^{-rT}max(S_n(T) - k, 0)$$
$$Z_n = e^{rT}S_n(T)$$

The new option price will be given by the estimator

$$R := \frac{1}{N}\sum_{n=1}^{N}[X_n + c_{opt}(Z_n - E[Z])]$$

end the new MC error

$$err = \sqrt{\frac{\sum_{n=1}^{N}\{[X_n + c_{opt}(Z_n - E[Z])] - R\}}{N}}$$

where

$$c_{opt} = \frac{Cov(X,Z)}{Var(Z)}$$

Part II

Volatility Surface

In this second part of the document we report the volatility surfaces obtained from call option prices. The first section provides the volatilities as implied by Monte Carlo prices, whereas the second one provides the volatilities as implied by (quasi) analytical prices.

For each of the models we consider the grid (t_n, k_j), where t_n takes the values

$$1M, 2M, 3M, 6M, 12M, 18M$$

and for each t_n we consider the strikes

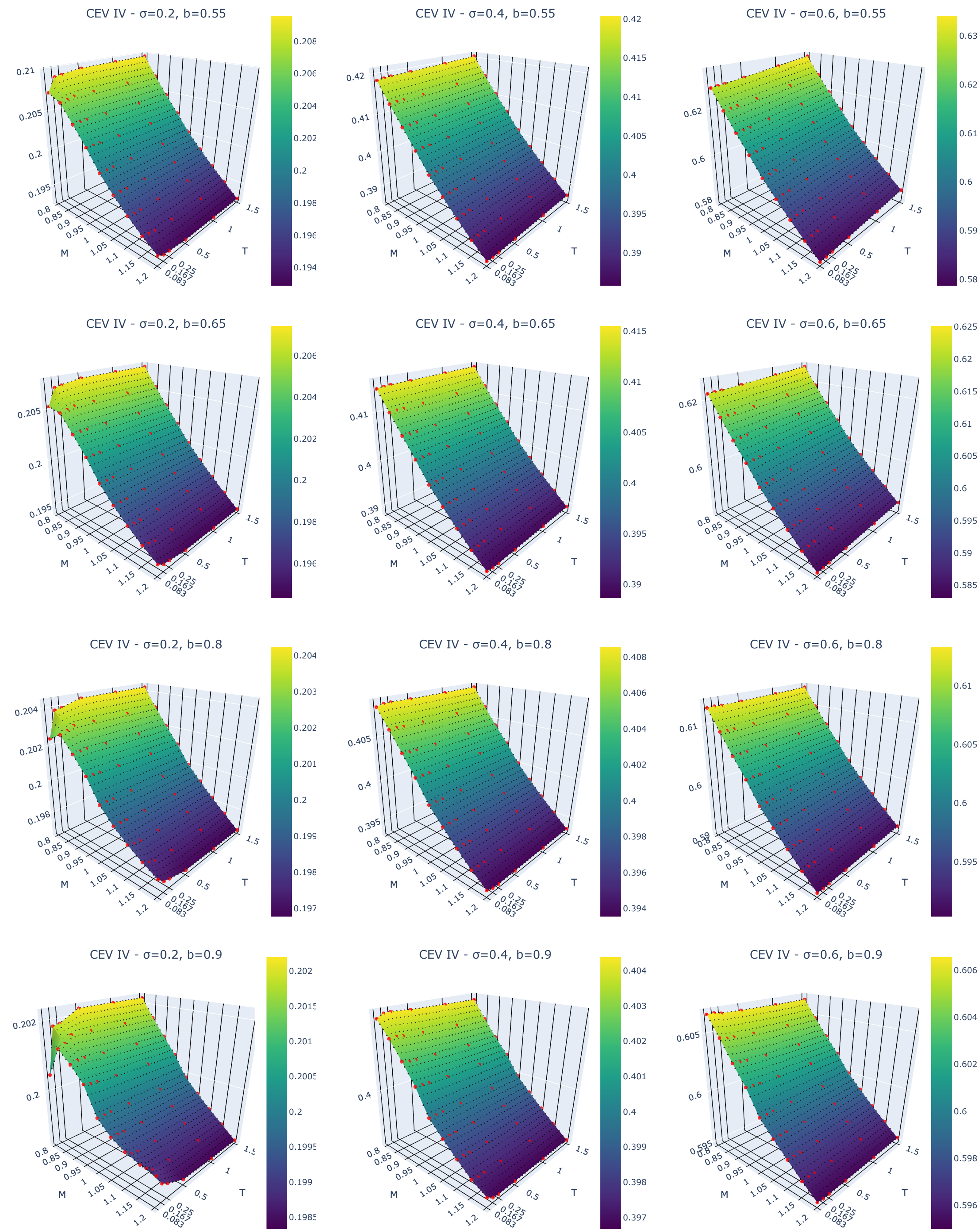
$$0.8, 0.85, 0.9, 0.95, 1.0, 1.05, 1.1, 1.15, 1.2$$

Besides, for every model we set $S_0 = 1.$.

3 Monte Carlo implied

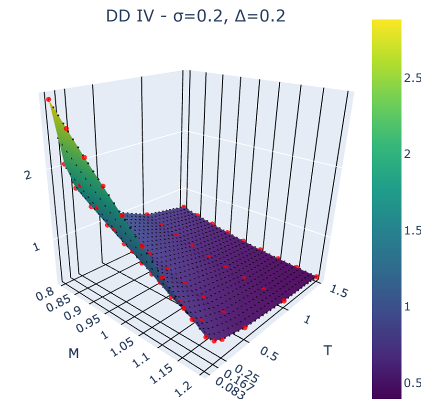
3.1 CEV

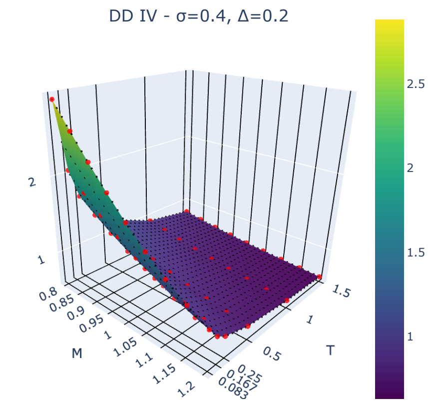
For the CEV model we explore $\sigma = 0.2, 0.4, 0.6$ and $\beta = 0.55, 0.65, 0.8, 0.9$.

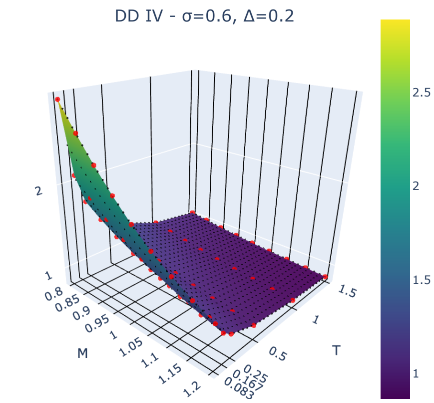


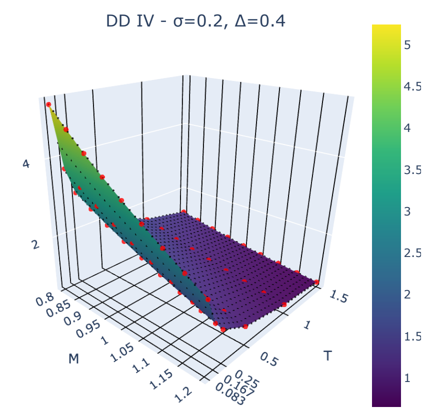
3.2 Displaced Diffusion

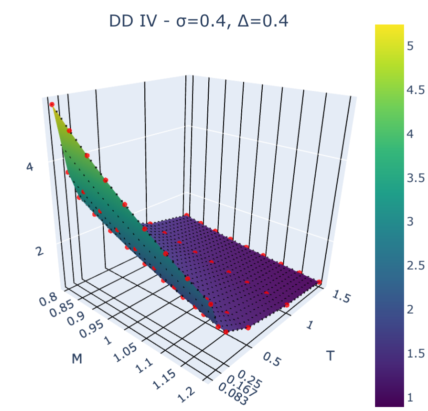
For the Displaced Diffusion model we explore $\sigma = 0.2, 0.4, 0.6$ and $\Delta = 0.2, 0.4, 0.5, 0.7$.

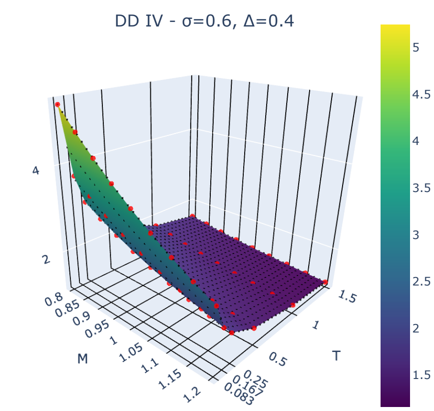


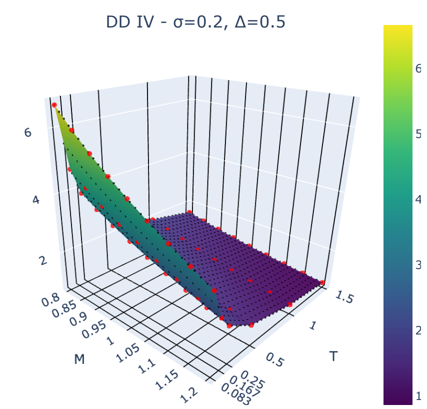


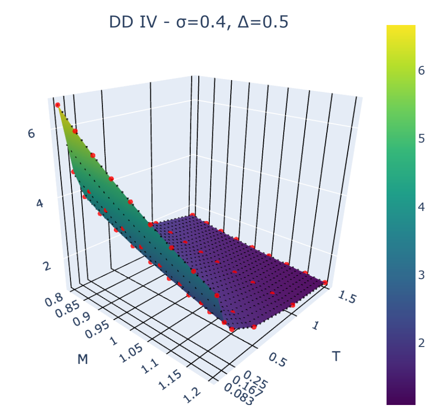


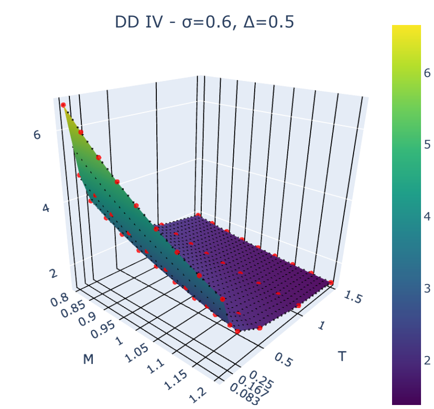


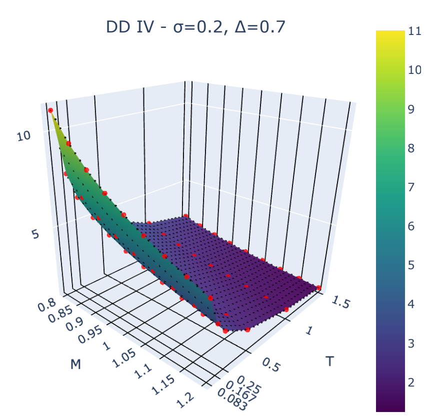


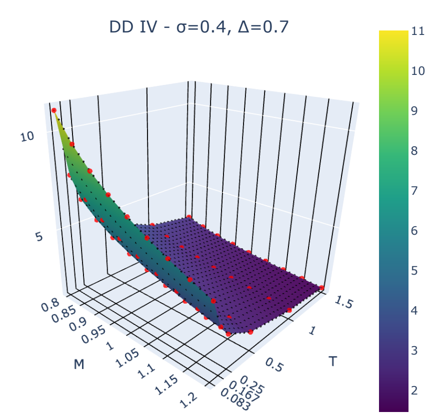


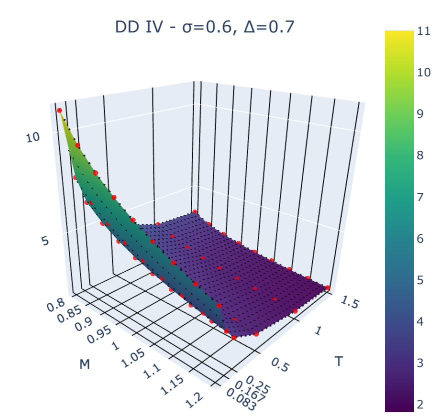








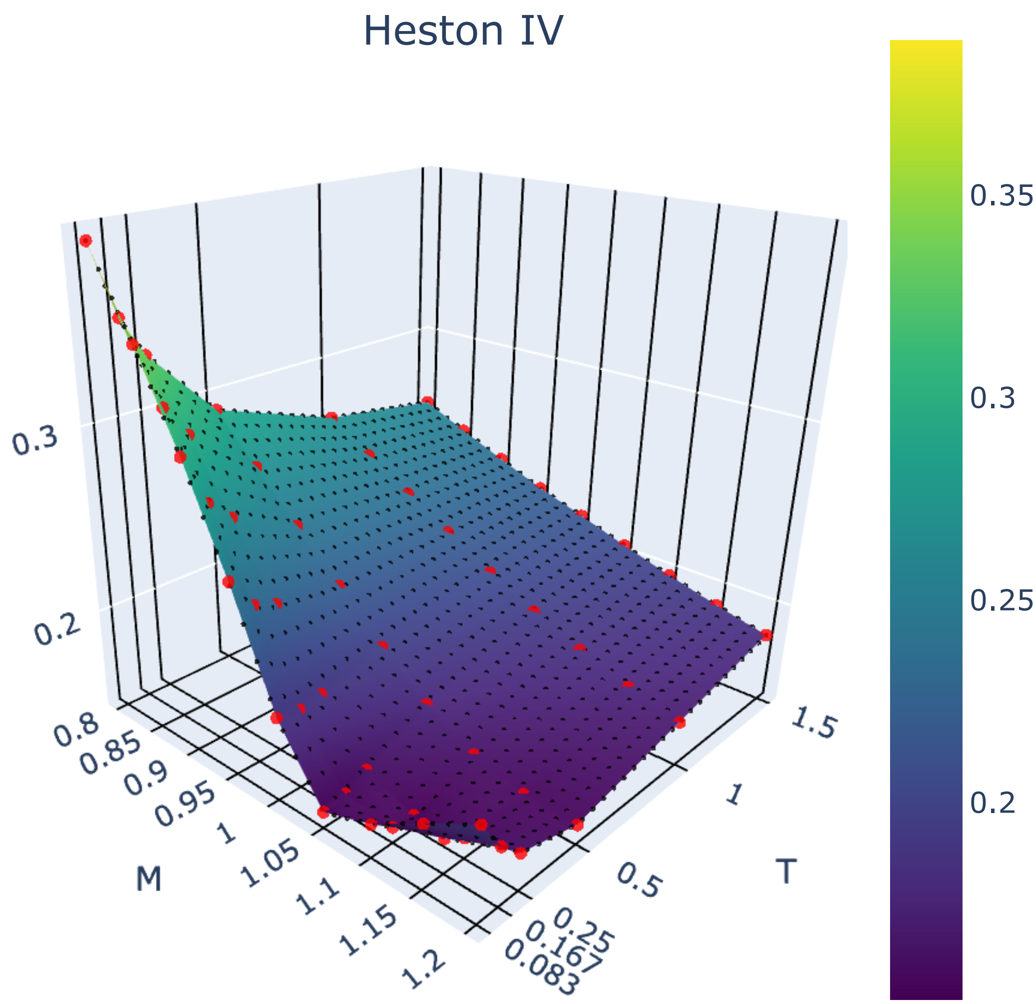




3.3 Heston

For the Heston model we set

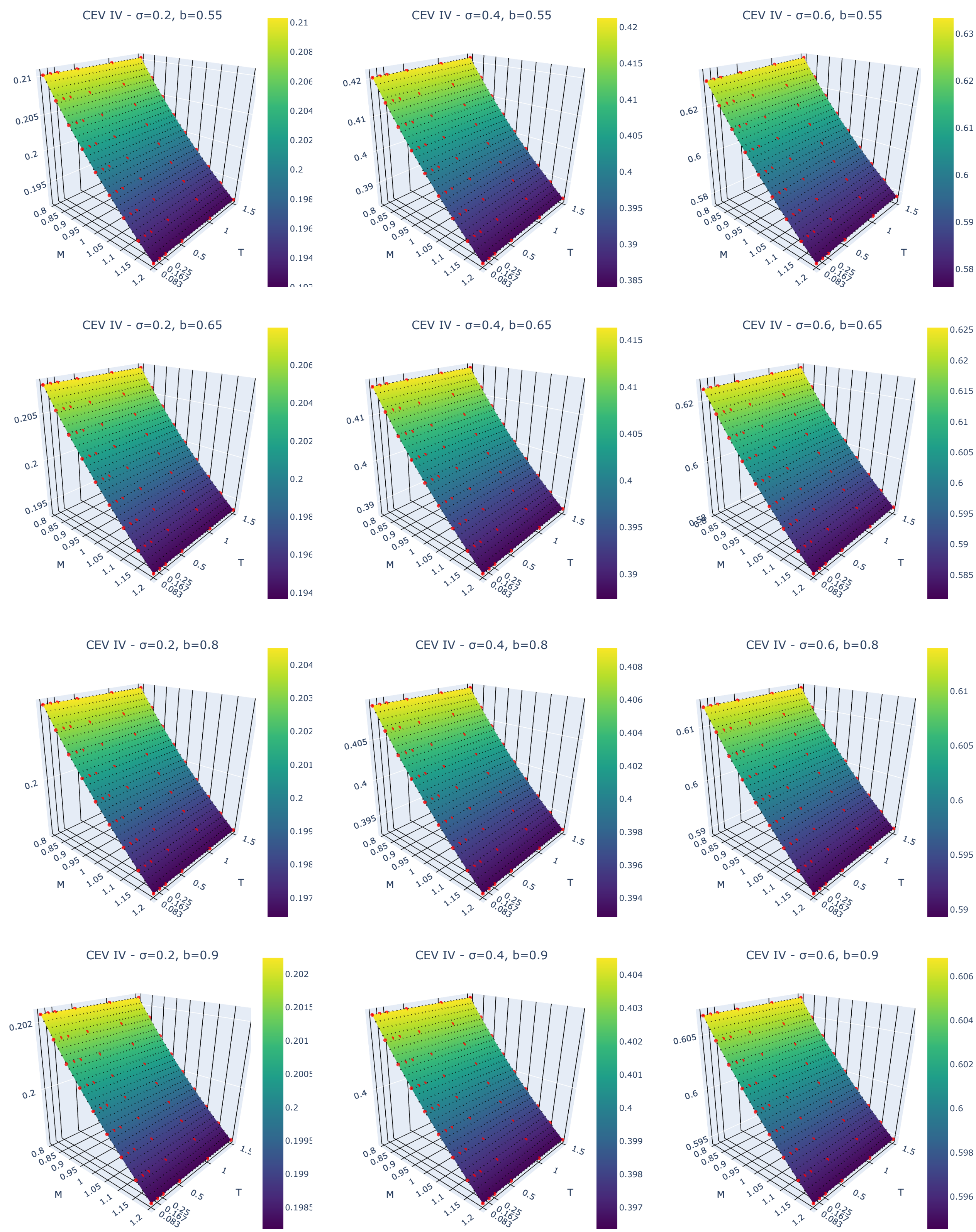
$$\lambda = 7.7648,\ \bar{v} = 0.0601\ \eta = 2.0170,\ \rho = -0.6952,\ v_0 = 00.0475$$



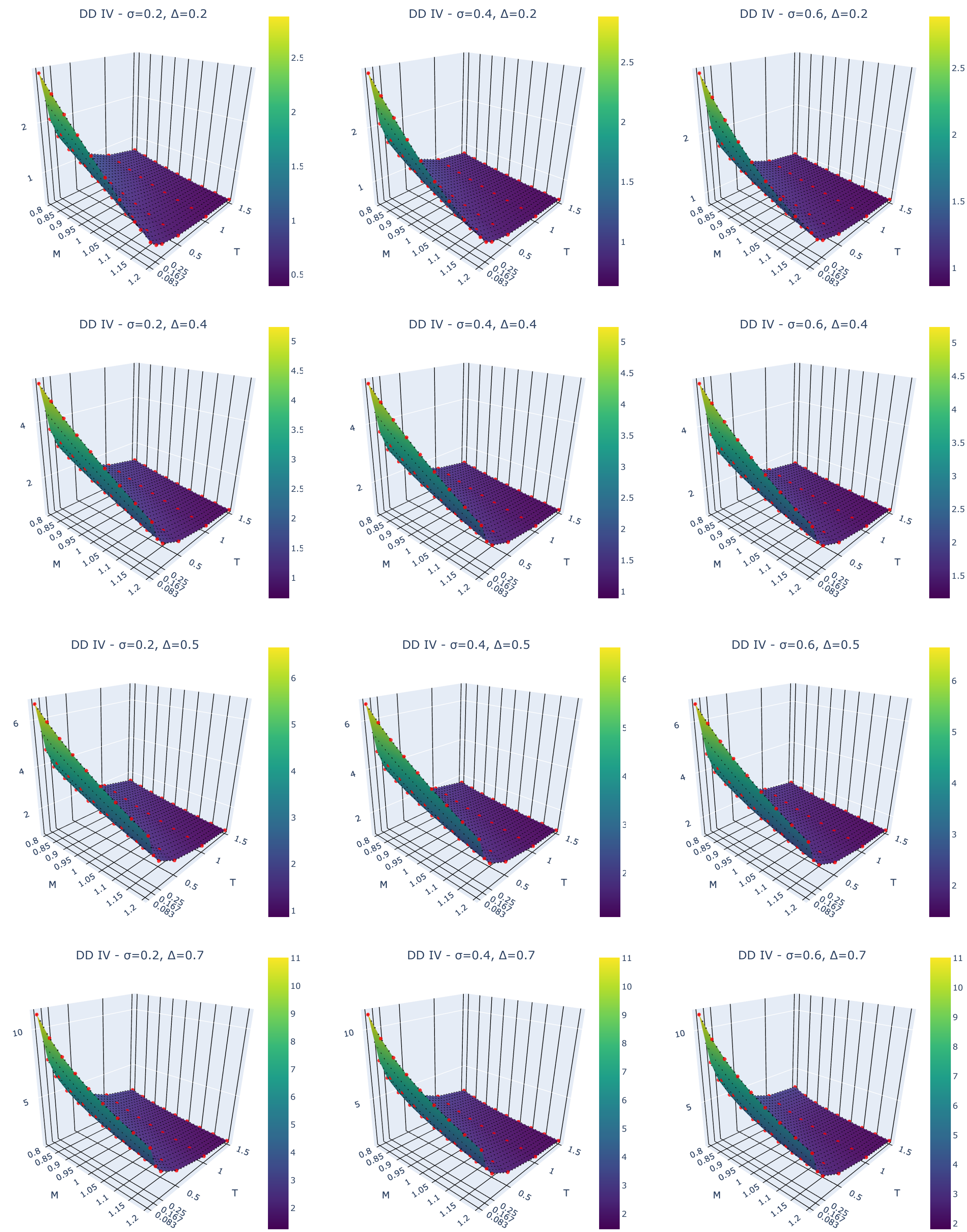
4 Super Bonus: analytical implied

Using the same set of parameters described in Section 3, we report here the volatility surfaces as implied by the (quasi) analytical call option prices.

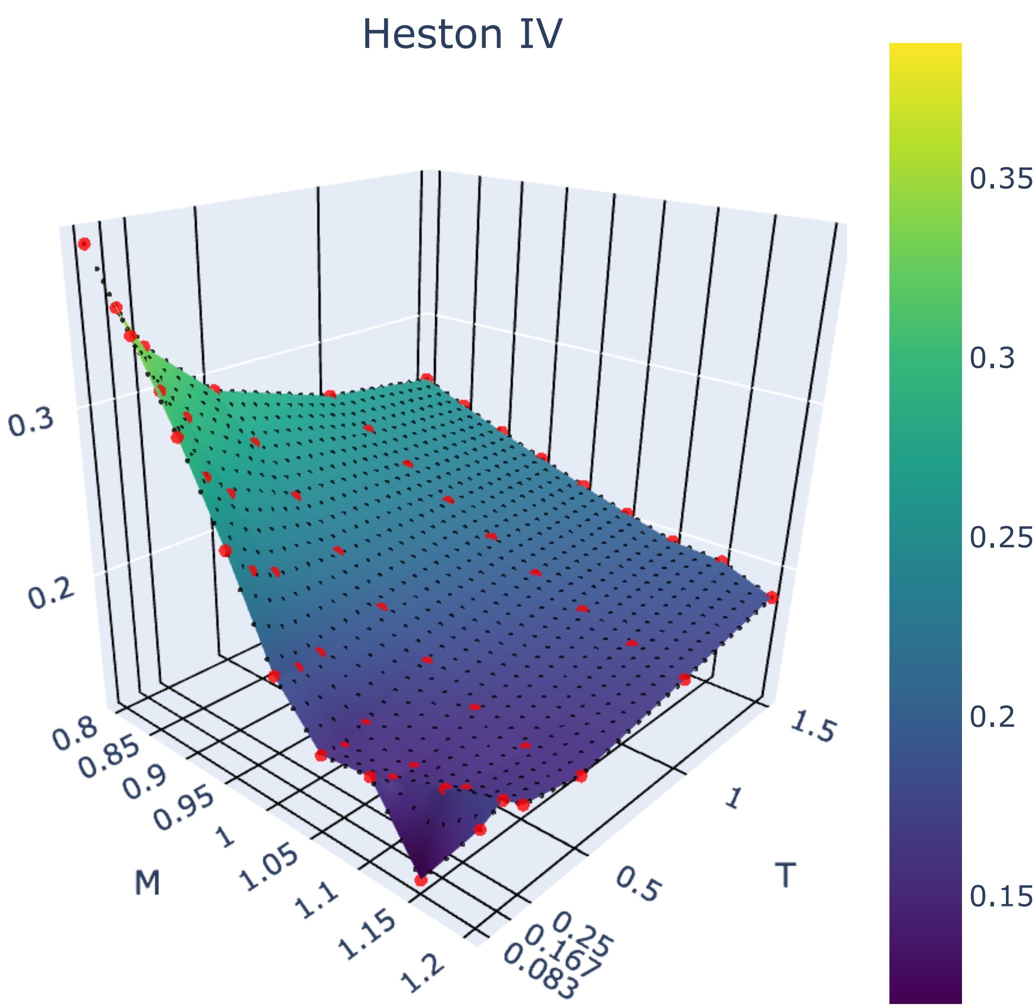
4.1 CEV



4.2 Displaced Diffusion



4.3 Heston


Heston IV
0.3
0.2
0.8
0.85
0.9
0.95
1
1.05
1.1
1.15
1.2
M
1.5
1
0.5
0.25
0.167
0.083
T
0.35
0.3
0.25
0.2
0.15