X Cubed: The Beginning of 3-D Volatility Interpolation

Qinying Chen, Nellie Garcia, Emily Gullerud,
Shaoyu Huang, PK Kataboh, Matthew Williams

Mentor: Chris Bemis

July 30, 2023

1 Introduction

In options trading, *implied volatility* is an important measure which informs investors the level of risk associated to a particular options contract and allows for the assessment of the value of a security. In particular, volatility is a crucial component of the Black-Scholes Model, which gives us a way of calculating the prices of European options. Letting T be the time to maturity, K the strike price, and S_t the current price of the derivative at time $t \leq T$, the formulae for call price $C(S_t, K)$ and put price $P(S_t, K)$ are as follows:

$$C(S_t, K) = N(d_+)S_t - N(d_-)Ke^{-r(T-t)}$$
$$P(S_t, K) = N(-d_-)Ke^{-r(T-t)} - N(-d_+)S_t$$

These contain five parameters: S_t, K, the expiry $T-t$, the risk-free rate r, and the volatility σ. All of these can be observed in the market except σ. Thus, we need a reliable and robust method for estimating σ in order to accurately price options.

The original formulation of the Black-Scholes Model assumes that implied volatility is constant, but real world market trends have since proven otherwise. Fixing a specific purchase date and expiry, the trend of implied volatility with respect to strike price resembles a smile. In practice, this more often looks like a smirk due to the amount of in-the-money puts. Figure 1 illustrates this phenomenon using Apple options data from 2022-2023, a data set that we will utilize for the remainder of this paper.

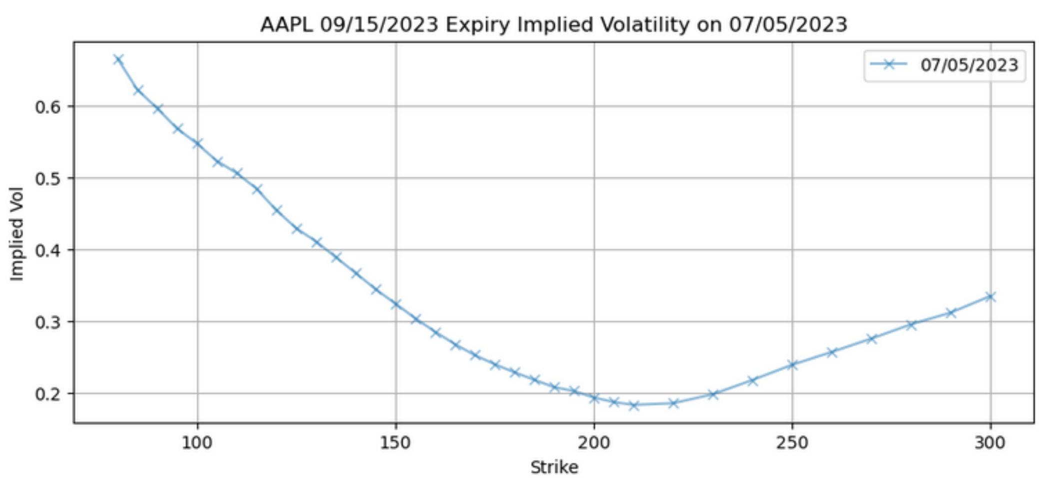

Figure 1: Example of a volatility smile using Apple data.

These smiles can be interpolated via various methods, most notably by Gatheral's stochastic volatility inspired (SVI) model [1]. The implementation of this model can be improved via a quasi-explicit calibration developed in 2012 by Zeliade Systems [2]. We explore this quasi-explicit calibration and propose a modification via a change in constraints for the model. This modification reduces the root-mean-squared error and produces a smile more tightly aligned with the raw data for implied volatility.

For a given security, we can build a volatility surface. One way to do that is using Gatheral's SVI square root fit (SSVI), where we are able to eliminate the butterfly and calendar arbitrage. Another way is to take many volatility smiles over time and interpolate between them. In particular, in the second method, we produce volatility surfaces by utilizing our modified quasi-explicit model and eliminate the butterfly arbitrage with cross penalty. These surfaces gives investors insight into the time dynamics for volatility and allow for more informed decisions for the purchase and sale of options in an effort to maximize gains.

2 Interpolating Volatility Smiles

2.1 Gatheral's Raw SVI Parameterization

Jim Gatheral created the following *stochastic volatility inspired* (SVI) model [1], sometimes referred to as the *raw SVI parameterization*, to describe total implied variance:

$$w(x) = a + b\left(\rho(x-m) + \sqrt{(x-m)^2 + \sigma^2}\right),$$

where the output w is the total implied variance, x is the log-forward moneyness, and $a, b, \rho, m,$ and σ are model paramters. A more specific description of x is $x = \log(K/F_T)$, where K is the strike price, and $F_t = Se^{(r-q)t}$ is the forward price for a given spot price S, risk-free interest rate r, dividend yield q and t being years to expiration.

When graphing the raw SVI parameterization, the model parameters are represented in the following ways: a translates the graph vertically, b is the angle between the left and right asymptotes, ρ determines the orientation, m translates the graph horizontally, and σ affects the smoothness of the graph.

Given a set of data points for a real smile, we can use a curve-fitting optimizer (e.g. curve_fit from scipy.optimize in Python) to determine the optimal parameters which minimize the root-mean-square error (RMSE) between the curve and the data points. Figure 2 is one such example using data for AAPL stock options bought on 6/30/2023 and expiring on 9/15/2023. The parameters for this specific SVI model are as follows:

$$a = -0.4059326, b = 0.43541219, \rho = 0.47816111, m = 0.68403152, \sigma = 1.07998044$$

While Figure 2 looks nice, the raw SVI parameterization has issues with robustness and stability. Any least square optimizer will depend on good initial guesses for the model parameters, and we may not have the information available to make such guesses. This kind of optimization also has a tendency to create similar smiles from different "optimal" parameters, and this is an issue when trying to create a volatility surface. In practice, using curve_fit, we will not get any solution within reasonable iterations in many case ($\frac{1}{3}$ of the data from 6/30/2023).

2.2 Quasi-Explicit Calibration

One solution presented is a quasi-explicit calibration of Gatheral's SVI model by Zeliade Systems [2]. Their method reduces the number of parameters in the optimization process from five ($a, b, \rho, m,$ and σ) down to two (m and σ). Once the optimal values of m and σ

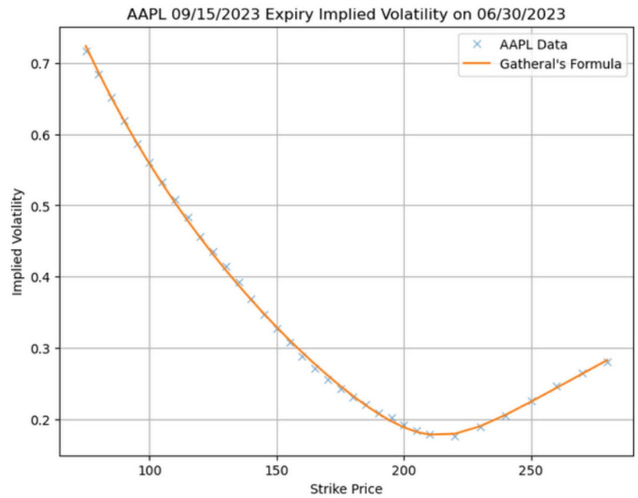

Figure 2: Example of a volatility smile fit with Gatheral's SVI model.

are determined, the other parameters a, b, and ρ are explicitly calculated based on m and σ. In addition to making a model that is more robust and stable, the quasi-explicit calibration also eliminates butterfly arbitrage.

The quasi-explicit calibration starts with the same parameters a, b, ρ, m, and σ, but with the following constraints:

$$b > 0, \sigma \geq 0, \rho \in [-1, 1].$$

They also apply the following constraint to eliminate static arbitration:

$$b \leq \frac{4}{(1 + |\rho|)T}.$$

After setting up the constraints, the calibration makes the following change of variables and parameter substitutions:

$$y(x) = \frac{x - m}{\sigma}, \;\; z(x) = \sqrt{y(x)^2 + 1},$$
$$\hat{a} = a, c = b\sigma, d = \rho b \sigma.$$

Implementing these substitutions into Gatheral's SVI model yields the following equation:

$$w(x) = \hat{a} + dy(x) + cz(x),$$

The calibration also applies the following constraints on $\hat{a}, c$, and d:

$$0 \leq c \leq 4\sigma$$

$$|d| \leq c$$
$$|d| \leq 4\sigma - c$$
$$0 \leq \hat{a} \leq \max_i \{w_i\},$$

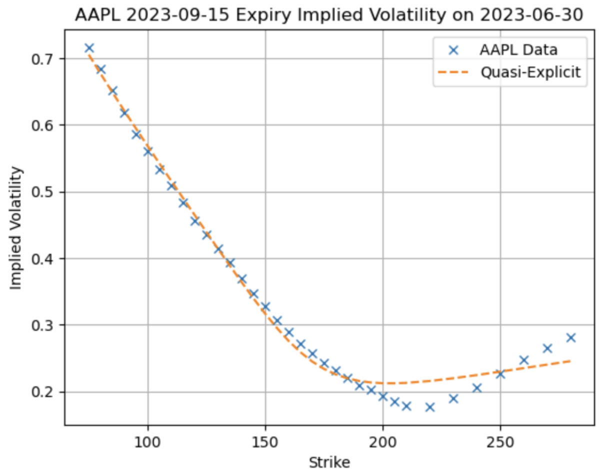

Figure 3: Example of a volatility smile fit with a quasi-explicit calibration.

where the w_i are the total implied volatility data points from the data set that we are modeling.

For fixed m and σ, the new cost function is

$$f_{x,w}(\hat{a}, c, d) = \sum_{i=1}^{n} (\hat{a} + dy(x_i) + cz(x_i) - w_i)^2.$$

Since all the constraints and the partial derivatives of f are linear, there exists a unique solution by Lagrange multiplier.

We then use minimize optimizer in scipy.optimize with Nelder-Mead method iterating on m and σ. One such example of this is shown in Figure 3.

At a glance, this is not as good as raw SVI parameterization. This is reasonable. If we indeed get an optimal answer without constraints, it should behave better than the answer with constraints. However, the stability of the parameters are better in this case, which can be seen from Figure 4. (If we do not get solution from particular date in raw SVI parameterization, we let all the parameters to be 0.)

2.3 Butterfly Arbitrage

Throughout this project, we have been concerned with having static arbitrage in our model. According to Gatheral,

Definition 2.1. A volatility surface is free of *static arbitrage* if and only if:

1. it is free of calendar spread arbitrage;
2. each time slice is free of butterfly arbitrage.[1, p. 3]

In this section, we will review butterfly arbitrage, as it mainly concerns our volatility smiles. The mathematical definition of butterfly arbitrage is as follows:

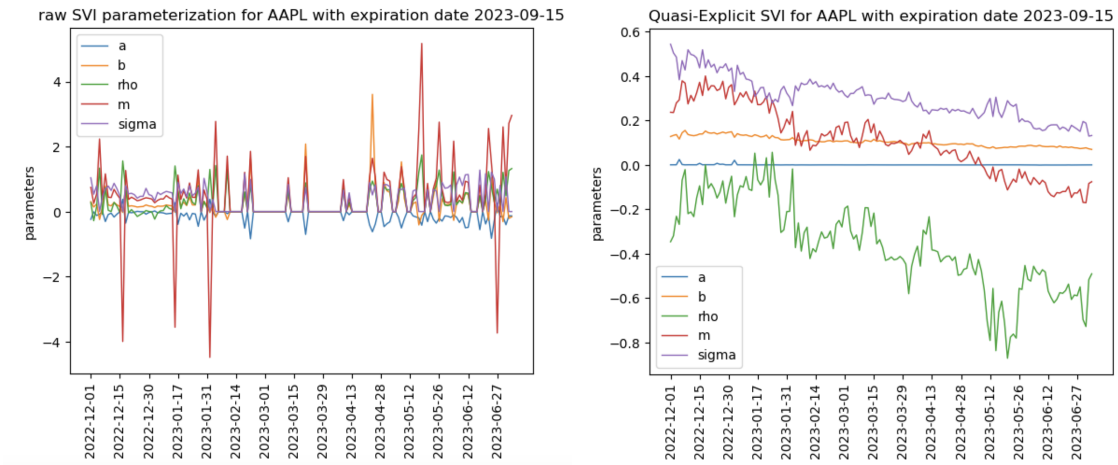

Figure 4: Parameters from raw SVI and Quasi-Explicit method

Definition 2.2. A slice is free of butterfly arbitrage if and only if $g(k) \geq 0$ for all $k \in \mathbb{R}$ and $\lim_{k\to+\infty} d_+(k) = -\infty$, where g is the *density function*

$$g(k) = \left(1 - \frac{kw'(k)}{2w(k)}\right)^2 - \frac{w'(k)^2}{4}\left(\frac{1}{w(k)} + \frac{1}{4}\right) + \frac{w''(k)}{2}$$

and w is our total implied variance $w(k,t) = \sigma^2_{\mathrm{BS}}(k,t)\cdot t$. [1, p. 4]

We tested our AAPL data for optioins bought on 6/30/2023 to see if it had butterfly arbitrage. Indeed, in raw SVI parameterization, there are some values of $g(k)$ which are negative according to the graph in Figure 5, which means the model contains butterfly arbitrage.

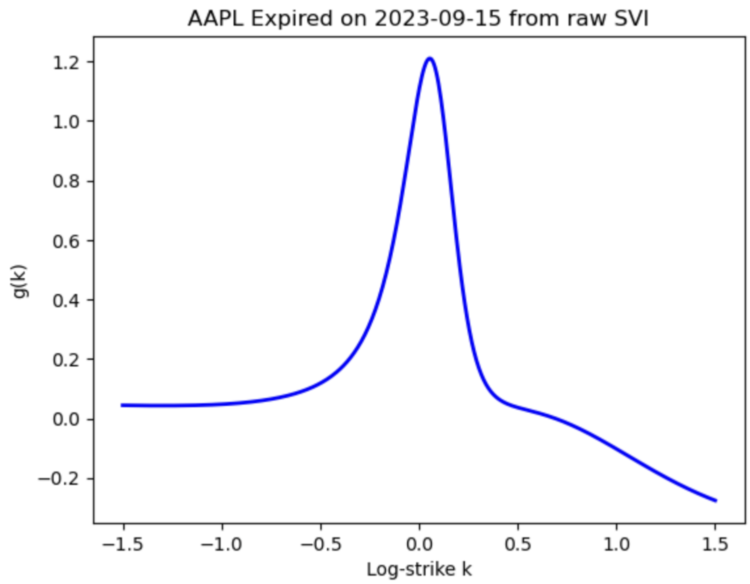

Figure 5: AAPL Butterfly Arbitrage Test

Meanwhile, as shown in Figure 6, the quasi-explicit method indeed gives us a density function without butterfly arbitrage.

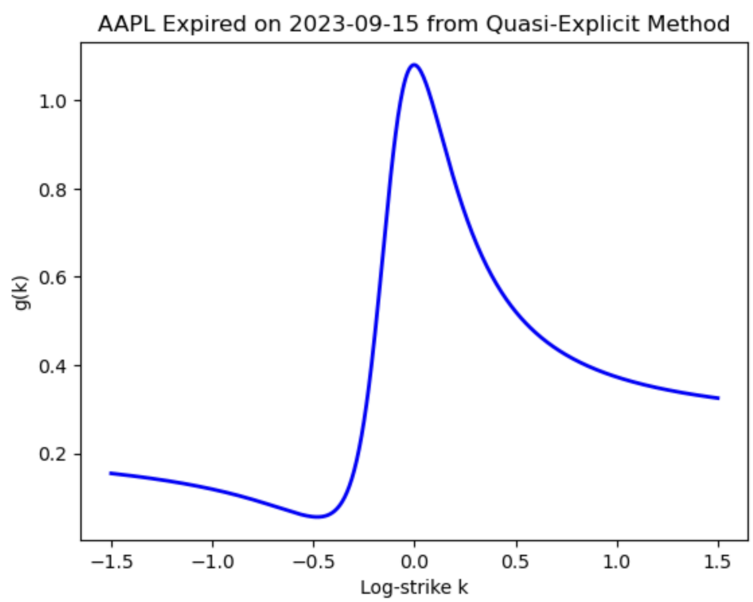

Figure 6: AAPL Butterfly Arbitrage Test

2.4 Quasi-Explicit Method Modified

In the quasi-explicit calibration, it requires $a \geq 0$. However, what we actually need is $f(y) = a + dy + c\sqrt{y^2+1} \geq 0$. Since $\frac{d}{c} = \rho \in [-1,1]$ and $c > 0$, f has minimum value $f(0) = a + c$ when $d \geq 0$ and $f(-\frac{d}{\sqrt{c^2-d^2}}) = a + \sqrt{c^2-d^2}$ when $d < 0$.

The modified constraint we set on a is $a + \sqrt{c^2-d^2} \geq 0$. Although it is no longer linear, this boundary does not change the uniqueness for global minimum for f. Figure 7 shows a good fitting by the modified quasi-explicit method.

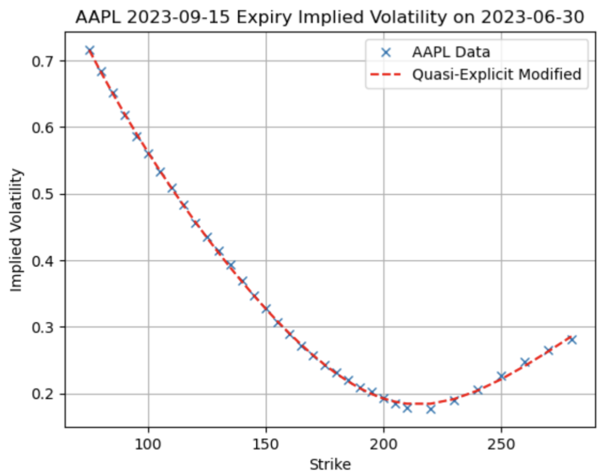

Figure 7: Example of a volatility smile fit with modified quasi-explicit method.

The comparison of RMSE is shown in Figure 8.

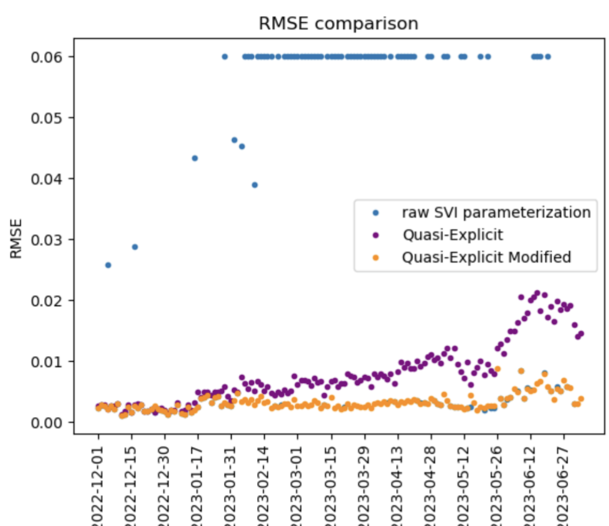

Figure 8: RMSE among different SVI parameterization methods.

We set the RMSE for data without solution from raw SVI parameterization to be 0.06

so that the graph is easier to read. The modified version indeed behaves better then original quasi-explicit method.

3 Volatility Smiles Over Time

This section examines the dynamics of implied volatility for AAPL options with fixed expiration dates. The smile curve for the same expiration also varies over time, reflecting changes in investor expectations and perceptions.

To investigate these dynamics, we employed Gatheral's SVI model and the quasi-explicit method by Zeliade Systems. Using these mathematical tools, we are able to generate an interpolated implied volatility time series that captures the changes in the smile curve. This time series offers valuable insights into market behavior and risk expectations, allowing for improved risk management and more informed option trading strategies.

One notable observation from the implied volatility time series is the drastic volatility changes that occur close to the expiration date. As the expiration date approaches, the implied volatility tends to experience significant fluctuations. This phenomenon is essential for option traders and investors to take into account when making trading decisions, as it can affect the pricing and profitability of options.

However, when dealing with multiple dates, graphing the implied volatility smile can be challenging and result in graphs that are difficult to read (see Figure 9). To address this problem, we transform the graph into a three-dimensional representation. By including an axis for days to expiration, we can effectively visualize the implied volatility changes over time and expiration dates. This 3-dimensional approach provides a clearer and more intuitive understanding of how implied volatility varies throughout the option's expiration date.

In summary, our analysis of implied volatility dynamics for AAPL options with a fixed expiration date reveals significant variations in the smile curve over time. The interpolated implied volatility time series, obtained through Gatheral's SVI model and the quasi-explicit method, serves as a powerful tool for traders and investors to gain critical insights into market sentiment and risk expectations. Indeed, a 3-dimensional graph enhances the readability of the results, allowing for well-informed decision-making processes in the option market.

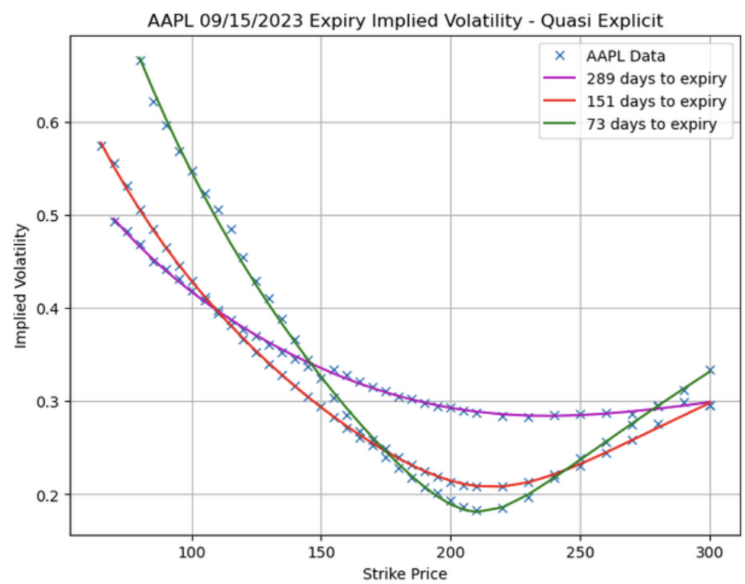

Figure 9: Volatility smiles over time fits with the Quasi-Explicit model.

4 3-Dimensional Plot of Smiles

While volatility smiles can be plotted over time, the graph can become increasingly difficult to read as we add more smiles. This impedes us from fully understanding the changes in implied volatility as we get closer to the expiration date. A solution to this is to graph a 3-Dimensional plot of volatility smiles. In other words, we can graph a volatility surface using our raw data.

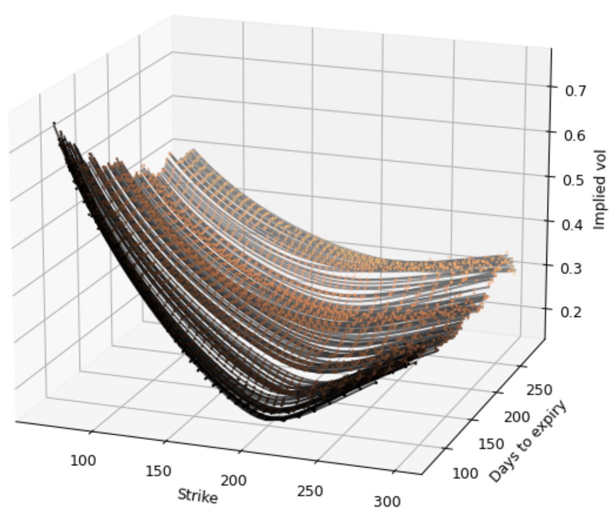

Figure 10: Multiple smiles plotted as a surface

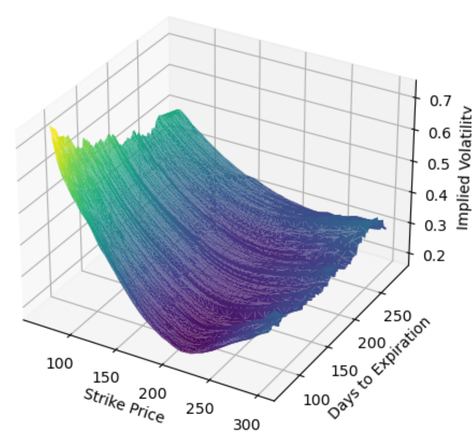

Figure 11: Volatility Surface

Figure 10 is created by plotting each smile individually through a for-loop in Python.

Namely, for each expiration date, we use scipy's optimize.curve_fit command on the Strike and Implied Volatility values. This allows us to see each volatility smile on a 3D-plot as opposed to a totally smooth surface. It does not have arbitrage conditions and uses the raw data, however, so refinement is needed.

In Figure 11, we let

$$\text{Strike Price} = \text{strike}$$
$$\text{Days to Expiration} = \text{expiration\_days}$$
$$\text{Implied Volatility} = \text{implied\_vol}$$

where the right-hand side is a column in our AAPL data. It is a simple plot using matplotlib's plot_trisurf. Similar to the previous graph, it does not have arbitrage conditions, nor does it use Gatheral's SSVI formula. As such, it may not be the most useful for assessing volatility over time. Therefore, we have worked on interpolating a volatility surface using Gatheral's SSVI which has constraints to avoid arbitrage.

5 Calendar Spread Arbitrage

Calendar arbitrage, also known as time spread, is an options trading strategy that exploits price discrepancies between options with different expiration dates on the same underlying asset. In this section, we delve into the concept of calendar arbitrage and its application to our AAPL option data, aiming to identify potential opportunities for profit in the market.

We discuss various methods of eliminating calendar arbitrage from our model. One effective approach is to ensure that the partial derivative of the total implied variance with respect to time remains nonnegative. Specifically, we seek to satisfy the condition $\partial_t w \geq 0$.

During our investigation, we observed that solely relying on the quasi-explicit calibration is insufficient to resolve the calendar spread arbitrage. After applying the method to our AAPL data, we notice crossings in the red and green lines, as well as a potential crossing in the orange and blue lines with fewer interactions. These findings suggest that numerous crossings may arise after several iterations, leading to violations of the calendar-free arbitrage principle. Figures 12 and 13 demonstrate that by combining the quasi-explicit method with Gatheral's SVI square root parametrization (which we will discuss in the next section) and fitting it to our AAPL data, we are able to create a model free from time spread arbitrage.

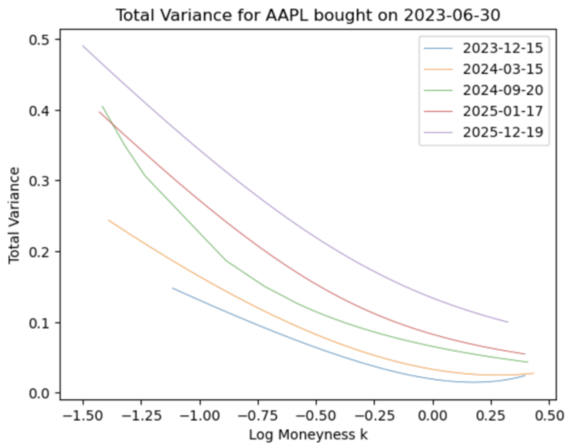

Figure 12: Quasi-Explicit Method

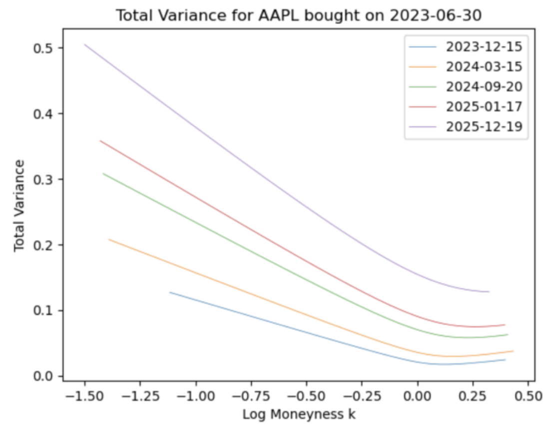

Figure 13: SVI Square Root

In practical terms, calendar arbitrage with AAPL options presents a unique opportunity to profit from time decay and potential price movements. Nevertheless, similar to any option strategy, it involves risks and necessitates vigilant monitoring. Traders employing this strategy must possess in-depth knowledge of options and a clear understanding of the potential behavior of the underlying asset. By effectively incorporating free calendar arbitrage in constructing our volatility surface, traders can position themselves for potential profits while managing their exposure to risk effectively.

6 Interpolating a Volatility Surface

6.1 Surface SVI: A surface free of static arbitrage

Gatheral introduced a class of SVI volatility surfaces called the SSVI (i.e. surface SVI) [1]. The SSVI is defined by

$$w(k, \theta_t) = \frac{\theta_t}{2}\{1 + \rho\varphi(\theta_t)k + \sqrt{(\varphi(\theta_t)k + \rho)^2 + (1-\rho)^2}\}$$

where w is the total implied variance, $\theta_t = \sigma_{BS}^2(0,t)t$ is at the money(ATM) total implied variance for $t \geq 0$, and φ is a smooth function from $\mathbb{R}_+^*$ to $\mathbb{R}_+^*$ such that $lim_{t\to 0}\theta_t\varphi(\theta_t)$ exists in $\mathbb{R}$. There are different choices of the function φ. For our experiment, we choose the square root parameterization $\varphi(\theta) = \dfrac{\eta}{\sqrt{\theta(1+\theta)}}$. Clearly, in this parameterization, we only have two parameters η and ρ to fit with.

Moreover, we can use interpolation function to find the ATM total implied variance θ_t at different expiration days t. Once we have η, ρ, and θ_t, we can then plot the volatility surface directly from the SSVI parameterization formula, or we can convert ρ and η back to the five parameters in SVI formula and plot the surface from there. Also, from corollary 4.1 in Gatheral's paper [1], it is quite straightforward to impose the arbitrage conditions for the square root parametrization. Indeed, if we require that $\eta > 0$ and $-1 < \rho < 1$, then the SSVI surface is free of static arbitrage if

$$(1 + |\rho|) \cdot \eta \cdot \sqrt{\frac{w_{max}}{1 + w_{max}}} \leq 4.$$

In our optimization step, the objective function we use is the sum of squared difference in the total variances, which are $\sigma^2 t$ from the data and the fitted term w.

Figure 14 shows the volatility surface plot using SSVI square root parameterization. It is plotted with the parameters η, ρ and θ_t. The surface plot looks very smooth because we interpolate θ_t over all t, not just the discrete expiration days. Moreover, it is free of static arbitrage as well. Next, we also want to use the Quasi-Explicit method to fit parameters and to plot the volatility surface. However, we still need to eliminate calendar spread arbitrage in this case. We impose a cross-penalty function that gives penalty for slices that cross each other on the total variance plot. Therefore, unlike the square root parameterization, here we have two steps: first is to use quasi-explicit method to find the parameters, and then use that as an initial guess to fit the total variance with a cross penalty function. Finally, we plot the volatility surface with the 5 parameters we obtained from output.

6.2 Eliminate Arbitrage from Quasi-explicit parameters

We use the crossness idea in [1]. Instead of considering the option price, we directly use the total variance. The implementation suggested in the paper does not work well in this case. One reason is that the method can not measure the case where there is only one intersection between two slices of total variance. Therefore, we use another method to measure the crossness.

We first get initial guess for parameters from modified Quasi-Explicit method. We iterate the slices from the one with longest expiration days. For the first slice, we do nothing on it. Starting from the second slice, we choose a sequence of points $\{x_i\}$ on log-strike axis. Let $w(x_i)$ and $w_{prev}(x_i)$ be the total variance of current slice and previous slice at x_i respectively. We define the crossness at x_i be $c_i = \max(0, w(x_i) - w_{prev}(x_i))$. We set the cross penalty to be $p = \max_i\{c_i\}$. Then we have the following cost function for the current slice:

$$f(\chi) = \frac{R(\chi)}{R(\chi_0)} + \gamma \cdot p,$$

where χ is the collection of parameters, χ_0 is the initial guess, γ is the penalty factor we can choose and $R(\chi)$ is the RMSE under χ.

Figure 15 shows the volatility surface using quasi-explicit method with cross-penalty. When we plot this, it might not be as smooth as the previous one because here we do not have interpolations over all t. Next, we want to compare the SSVI square root parametrization and quasi-explicit method with cross-penalty.

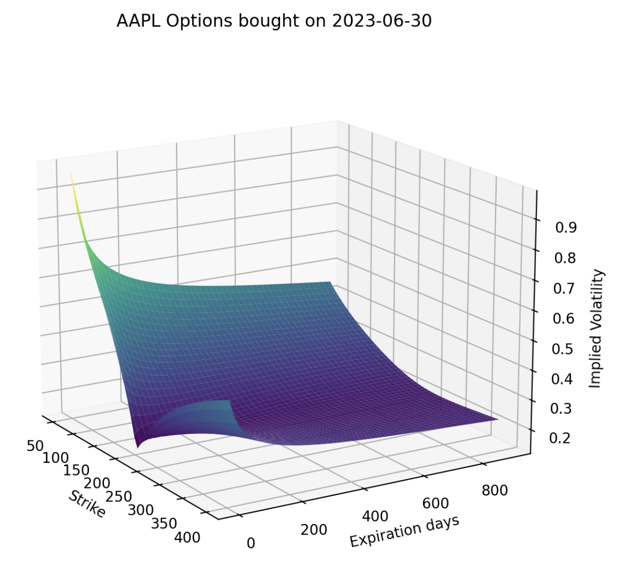

Figure 14: SSVI square root parametrization

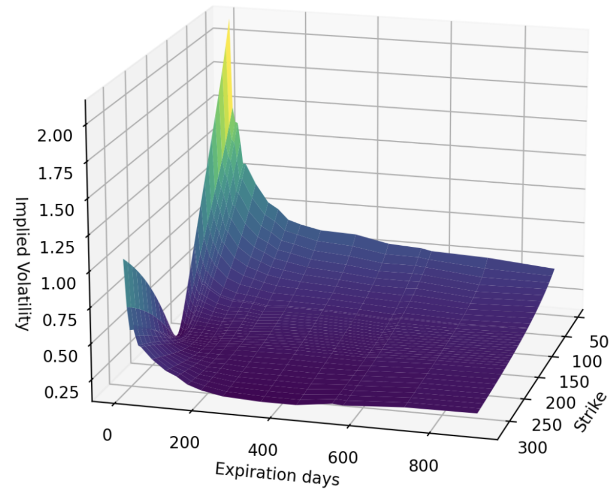

Figure 15: Quasi-Explicit with cross-penalty

Figure 16 shows how good the fit is using the two different methods. As we can see that the quasi method with cross-penalty performs much better than the SSVI sqaure root parametrization. One reason is that for the SSVI method we are only fitting two parameters, while for the quasi method we are fitting five paramters which could produce more accuarate results. Figure 17 shows that there is indeed no calendar arbitrage using our implementation of the quasi-explicit method with cross penalty.

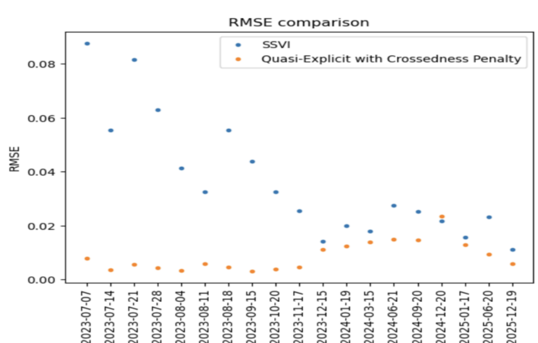

Figure 16: RMSE of SSVI vs. Quasi-Explicit

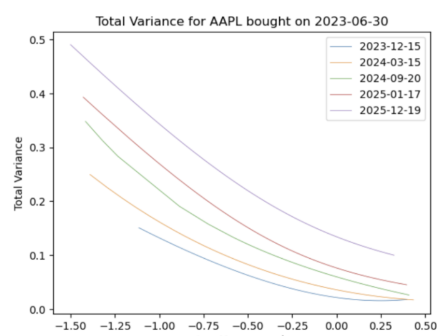

Figure 17: Arbitrage free with cross-penalty

7 Test on another dataset

Since the liquidity of AAPL option is good, we want to test our method in another data set which does not have high liquidity. We use iShares iBoxx $ High Yield Corporate Bond ETF (HYG) as the test set.

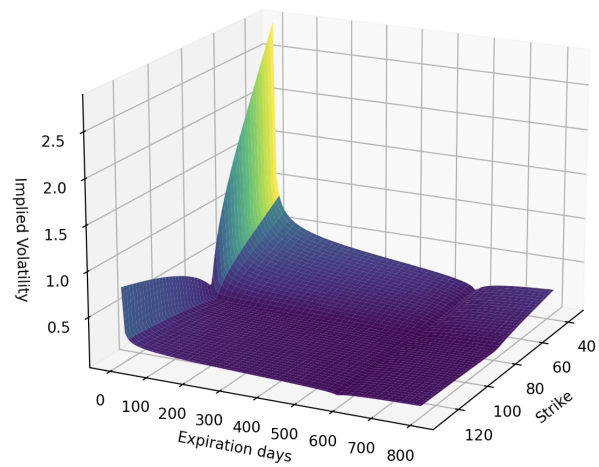

Figure 18: HYG options bought on 2023-06-30

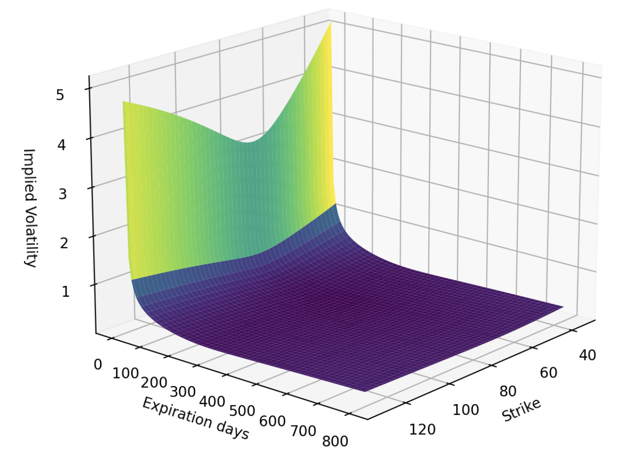

Figure 19: HYG options bought on 2022-12-30

Here we are using SSVI parameterization to interpolate the surface. The Quasi-Explicit method with cross-penalty does not work well in this situation. Because the bid-ask spread is quite large. This lets the total variance in the data not increase with respect to time in many cases, which makes it hard to apply cross penalty method.

A possible improvement is eliminating the data where total variance violates the increasing property. But this requires setting a proper metric to determine which data should be remained.

References

[1] Jim Gatheral & Antoine Jacquier (2014) Arbitrage-free SVI volatility surfaces, Quantitative Finance, 14:1, 59-71, DOI: 10.1080/14697688.2013.819986

[2] White, Zeliade. "Quasi-Explicit Calibration of Gatheral's SVI model." (2009).

[3] Gatheral, J. (2004). A parsimonious arbitrage-free implied volatility parameterization with application to the valuation of volatility derivatives. Presentation at Global Derivatives & Risk Management, Madrid, 0.