amberdata.io

SVI TrueLine

Techniques for creating consistent, stable and robust real time implied volatility calibrations

in the nascent cryptocurrency markets where the distribution of returns and liquidity vary

JULY 2024

COLLECTIVELY WRITTEN BY

GRÉGOIRE MAGADINI

EUAN SINCLAIR

SAMNEET CHEPAL

Abstract

This paper presents a robust methodology for calibrating the Stochastic Volatility Inspired (SVI) model to the unique volatility surfaces observed in cryptocurrency markets. Cryptocurrency markets are characterized by high volatility, distinctive trading behaviors, low liquidity, and the nascent nature of the asset class, all of which pose significant challenges for traditional volatility modeling techniques. Our approach leverages the inherent flexibility and efficiency of the SVI model to address these challenges effectively.

We introduce a proprietary calibration process specifically designed to manage the dynamic volatility shapes and low liquidity prevalent in cryptocurrency markets. This process includes advanced optimization techniques and liquidity-adjusted pricing mechanisms to ensure accurate and stable model calibration. Our findings demonstrate that the SVI model, when calibrated using our proprietary process, provides reliable volatility surface estimations, offering significant improvements over traditional methods.

The results highlight the potential of the SVI model in managing the unique risks associated with cryptocurrency trading and contribute to the broader understanding of volatility modeling in emerging financial markets. This paper aims to provide practitioners and researchers with valuable insights into the application of SVI in the rapidly evolving field of cryptocurrency trading.

Keywords: Implied volatility, stochastic volatility, calibration, rough calibration, matrix, SLSQP

Table of Contents

Abstract **2**

Introduction **4**

A brief history of the Stochastic Volatility Inspired Model 4

Adoption in Financial Markets 4

Application to Cryptocurrencies 4

Insights into SVI calibration **5**

SVI Parameterization 5

Calibration Process 5

Using the SVI Calibration Output 6

Describing the SVI Parameters and Constraints 7

SVI Calibration in Practice **11**

Liquidity Considerations 11

Calibrating to Prices vs Implied Volatilities 12

Using Mid, Mark or Bid-Ask 12

Sequential Least Square Quadratic Program (SLSQP) 13

Initial Value Matrix 17

Conclusion of Calibration 17

Real Time Calibration of the SVI_{TrueLine} **24**

Data Quality Evaluation 24

Data Fail-Safe 24

Calibration Evaluation 25

Calibration Fail-Safe 25

Data Signatures 26

References Page **27**

Introduction

A brief history of the Stochastic Volatility Inspired Model

The Stochastic Volatility Inspired (SVI) model was introduced by Merill Lynch in 1999 and popularized by Jim Gatheral when he introduced the SVI model in his book “The Volatility Surface: A Practitioner’s Guide” published in 2006, as a practical approach to modeling implied volatility in options markets. SVI is a parametric model designed for calibration of a time-specific slice of the implied volatility surface, capturing the dynamics of the skew/smile across strikes, which can be challenging to model using other traditional approaches.

Historically, global calibration models, such as the Heston model (1993), try to calibrate both across strikes and maturities simultaneously. Although this complete surface calibration is attractive at first glance, in practice the large calibration surface reduces the flexibility of the model and typically fails to properly calibrate wingy (far out-of-the-money) options and short-dated maturities, areas of the volatility surface subject to the most “jump-risk”.

At present time, many extensions to the SVI model exist, such as SSVI, which, like the Heston model, attempts to calibrate the global volatility surface across both strikes and maturities simultaneously.

Another popular extension to the SVI model is the SVI-JW (Jump Wings), which transforms the raw SVI parameter values, along with a time to maturity component, into parameter values that have concrete interpretations with respect to the volatility surface.

Adoption in Financial Markets

The SVI model gained traction among practitioners for its ability to fit market data more accurately and efficiently than other models. It became a standard tool for traders and risk managers in equity and equity index options markets.

Other asset classes coalesced around different parametric volatility models such as SABR in the rates market and Vanna-Volga in the FX markets.

Application to Cryptocurrencies

Cryptocurrency markets, known for their high volatility and distinctive trading behaviors, present challenges for traditional volatility modeling techniques. The SVI model, renowned for its flexibility and efficiency, excels in calibrating both large-cap equity indices and small-cap single name stocks. This makes it particularly well-suited for the dynamic volatility surfaces encountered in cryptocurrency markets.

Insights into SVI Calibration

SVI Parameterization

The SVI model describes an implied volatility surface slice as a function of strike price (K) and time to maturity (T). The SVI parameterization typically takes the form:

$$\delta^2(k) = a + b\left(p(k\text{-}m) + \sqrt{(k\text{-}m)^2 + \sigma^2}\right)$$

Where $\delta\,(k)$ is a function of the log-forward moneyness $k = \log\left(\frac{K}{F}\right)$ and $a, b, p\ m$ and σ are the SVI parameters.

Calibration Process

1. **Initial Guess:** Start with an initial guess for the parameters a, b, p, m and σ
2. **Objective Function:** Define an objective function that measures the difference between the model-implied volatilities and the market-implied volatilities, such as minimizing the mean-squared error (MSE) or the sum squared residual errors (SSR).
3. **Optimization:** Use an optimization algorithm (e.g. Sequential Least Squares Quadratic Program) to minimize the objective function. The optimization should respect any parameter constraints.
4. **Iterate and Converge:** Iterate the optimization process until the parameters converge to values that minimize the objective function.

SVI has proven to be an exceptional valuation tool; however, calibration processes often suffer problems of local minima on their way to minimizing the objective function and finding the desired global minima solution.

Finding a consistent calibration process can require substantial development, due to the nonlinear nature of the problem, with added complexity from the parameter constraints.

This often requires accurate initial parameter values in order to gently guide the optimization process to a successful solution, as opposed to getting lost and confused along the way, ending up at a dead-end local minima solution.

This problem is further complicated by liquidity and data collections issues for a nascent asset class like cryptocurrency.

Asset return dynamics can cause the option market to price various shapes due to varying regimes based on changes in second, third and fourth order moments of the distribution.

Unlike equity indices which have more stable distribution moments, cryptocurrencies moments are nascent, noisy and subject to a variety of shapes rarely seen in more efficient markets.

To illustrate the point of inefficiency further, even basic spot-price arbitrages have persisted in crypto markets in a manner thought theoretically improbable.

Using the SVI Calibration Output

Once a suitable SVI calibration has been performed, market practitioners are able to use the parameter values for interpolation and extrapolation of the market.

For example, when building a delta based implied volatility surface, practitioners may want to have consistent delta values, such as $\Delta 5$, $\Delta 15$, $\Delta 25$, $\Delta 35$, $\Delta 45$, etc, as reference points. Often, given the limited subset of listed option strikes, the target delta value isn't continuously available.

SVI parameters enable a continuous volatility segment to be reconstructed, despite the limited subset of option strikes used for initial calibration. This enables practitioners to always have a price or volatility for their desired target delta, for example.

Similarly, many practitioners will want to price exotic options with bespoke pay-offs both for path-dependent and path-independent exotics. Having the ability to price continuous strikes (e.g. 0.10¢ wide strikes in Ethereum) enables efficient theoretical replication for practitioners.

Furthermore, arbitrage-free volatility smiles can be ensured by having only positive values for the implied density of the distribution (no butterfly arbitrage).

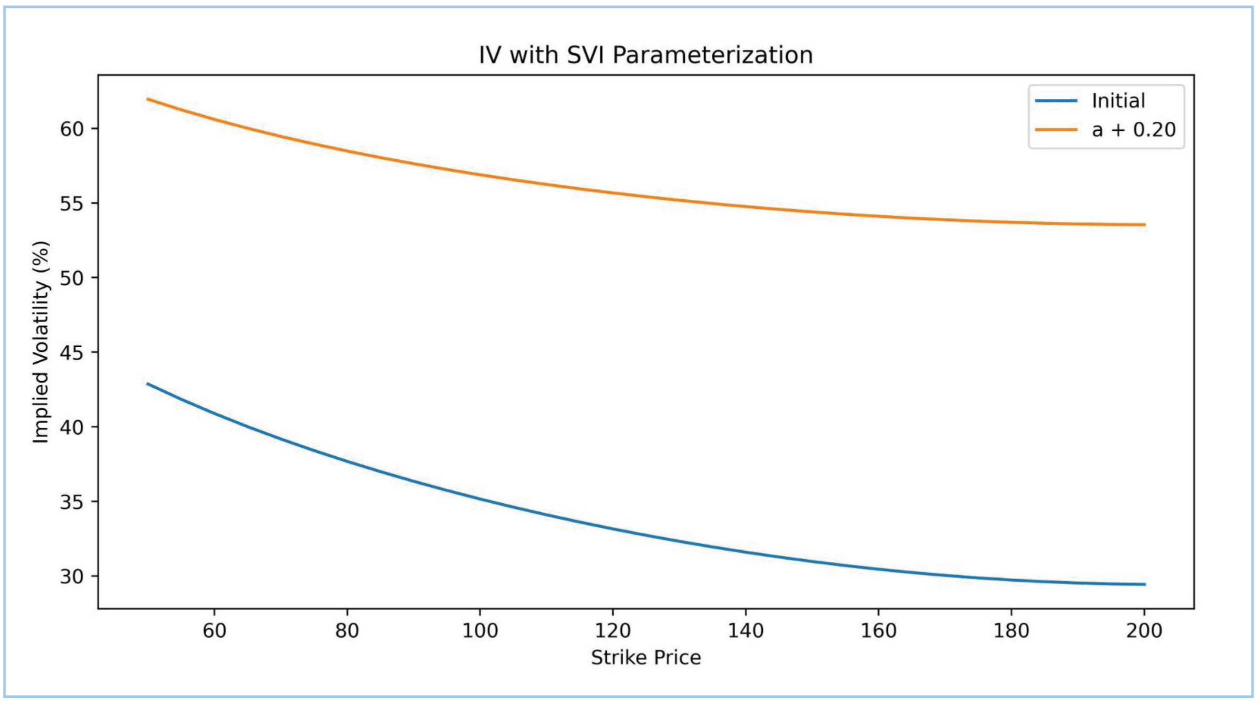
No Butterfly Arbitrage Bound: $$\frac{\partial^2 c}{\partial k^2} \geq 0 \quad (1)$$

Given that the SVI calibration occurs to a specific volatility time-slice, practitioners can ensure that no calendar arbitrage occurs by ensuring total variance increases with time as they perform interpolations across different maturities.

No Calendar Arbitrage Bound: $$\frac{\partial c}{\partial t} \geq 0 \quad (2)$$

Describing the SVI Parameters and Constraints

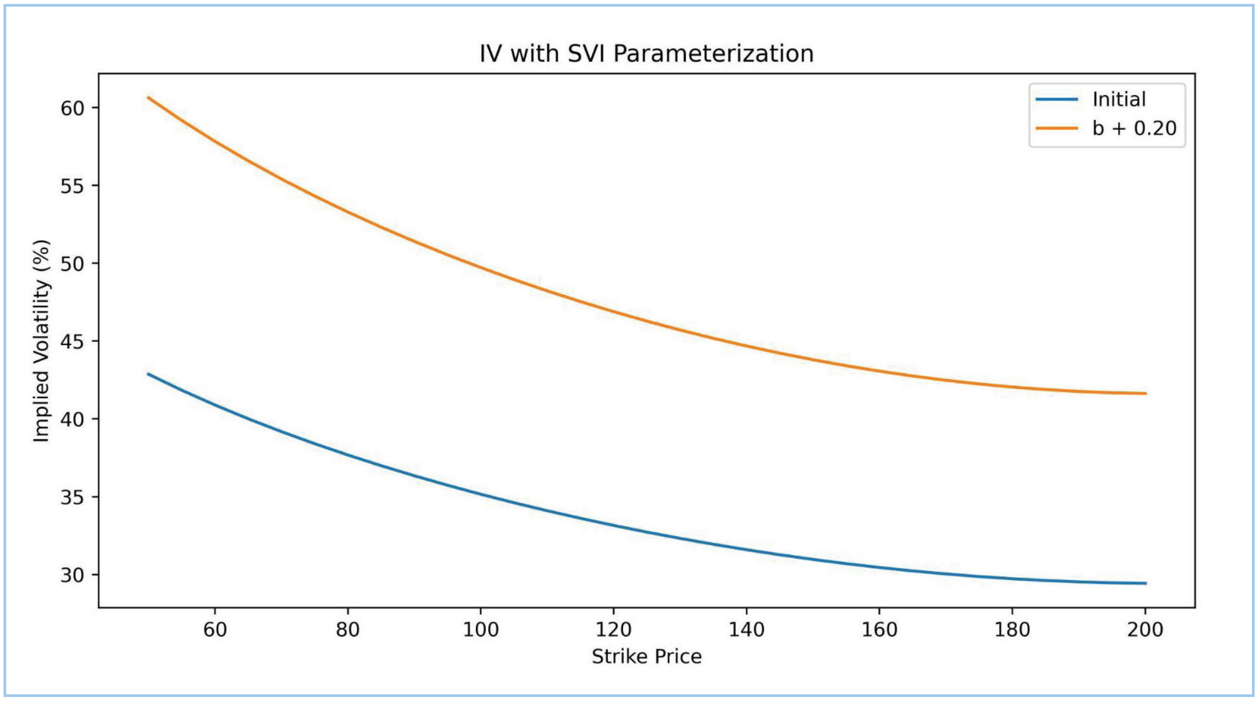
1. a : Vertical translation of the smile. Increasing or decreasing this parameter will cause a parallel shift in the level of volatility.



(Figure 1)

Logical parameter constraint ($a \in \mathbb{R}$) : This parameter can be any real number, as the parameterization process can tune the alpha parameter to shift the volatility level up/down.

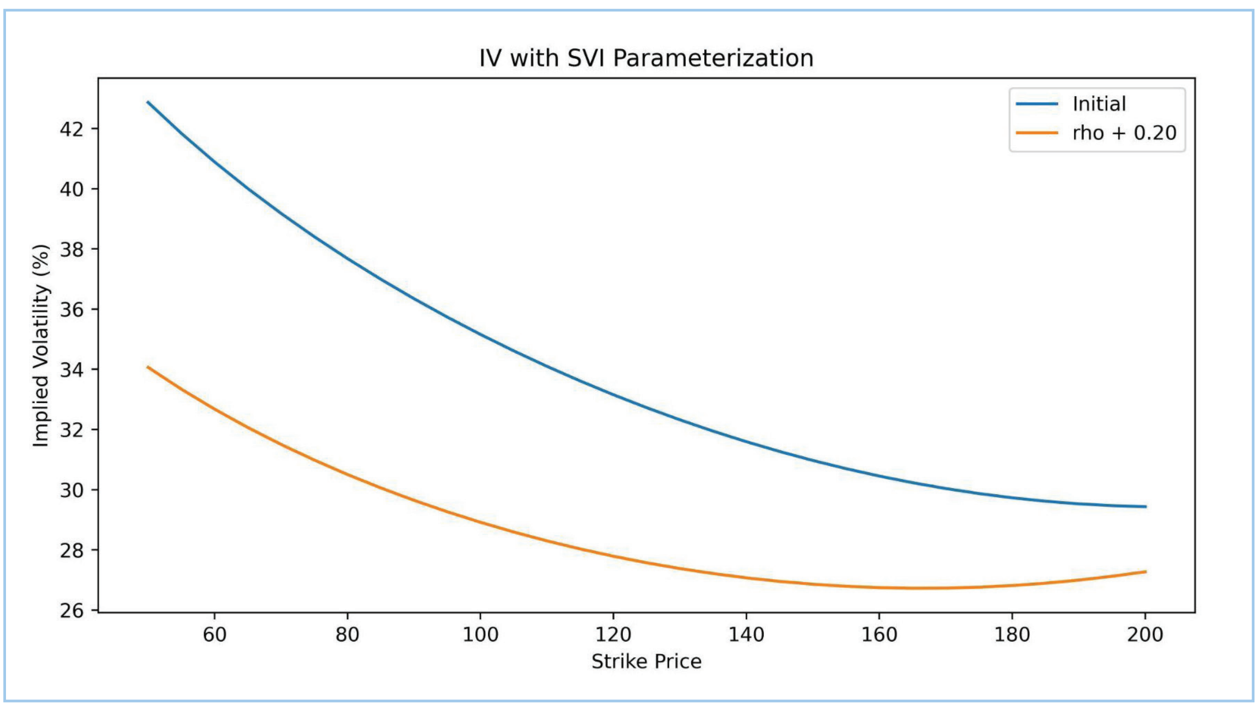
2. b : Controls the tightness of the volatility smile wings. Increasing or decreasing this parameter will cause a curvature shift in amplitude.



(Figure 2)

Logical parameter constraint ($b \geq 0$): This parameter must be non-negative, as all options should have a positive convexity to the curvature of the smile wings.

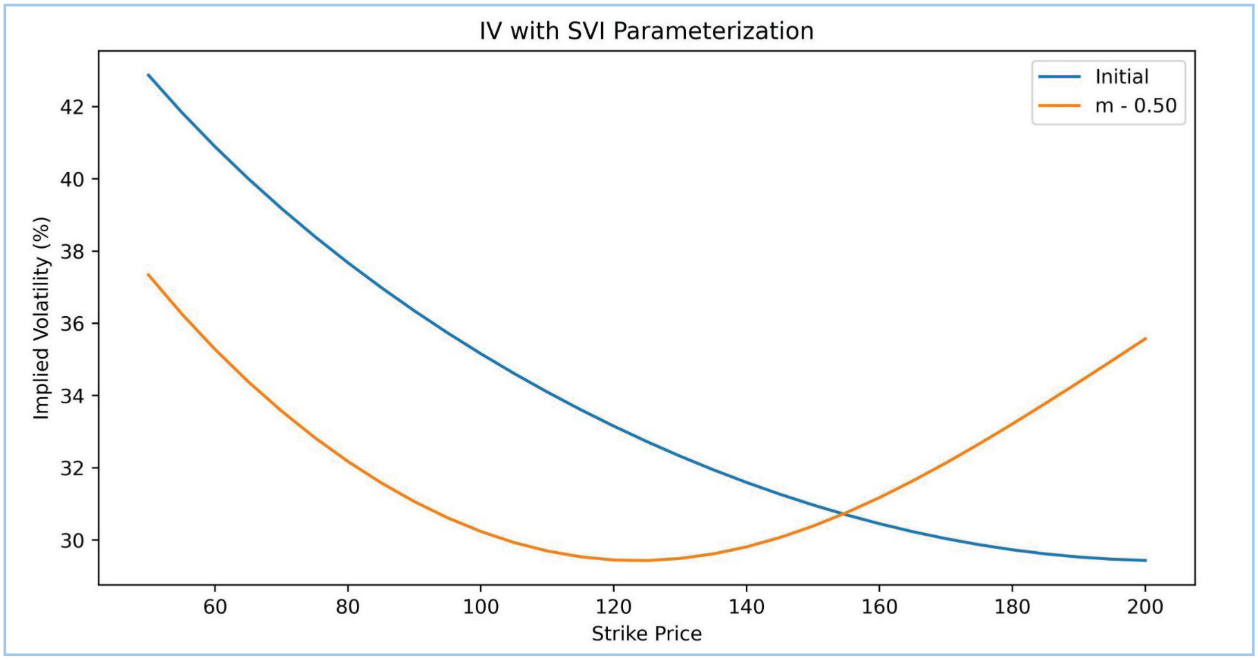
3. p : Controls the slope of the left wing.



(Figure 3)

Logical parameter constraint ($|p| < 1$) : This parameter is associated with the correlation dynamics of volatility and asset returns. The absolute value of correlation must be bounded by one.

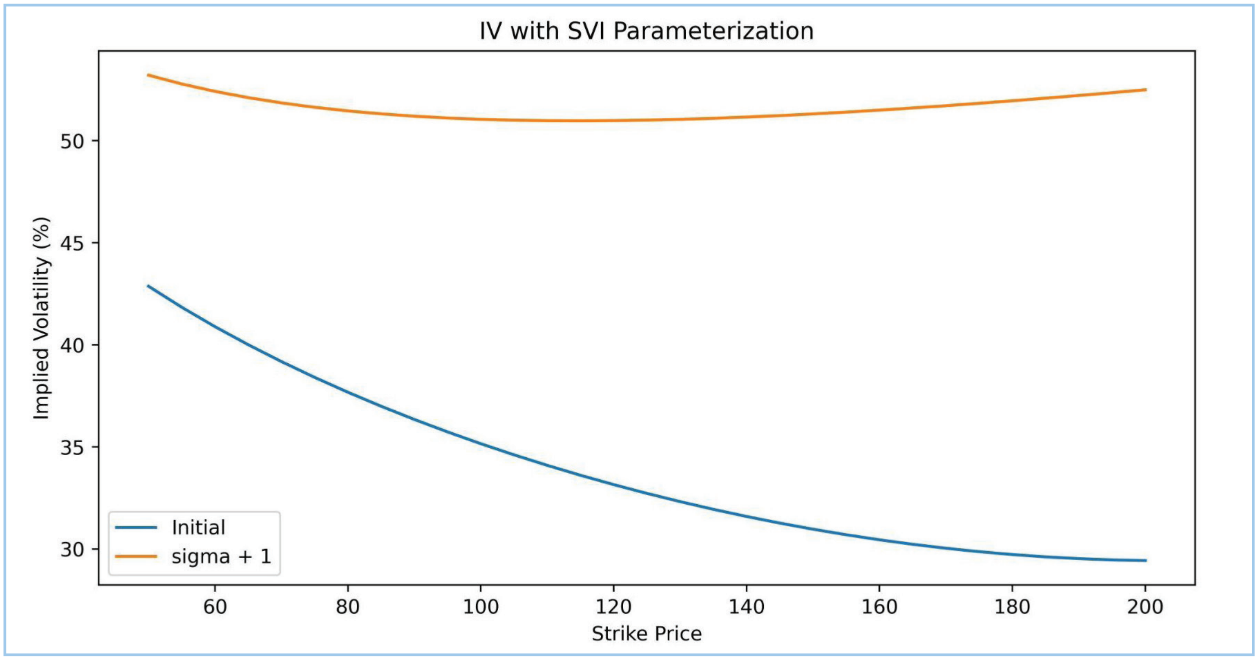
4. m : Represents the location of the volatility smile; an increase will shift the smile to the right.



(Figure 4)

Logical parameter constraint ($m \in \mathbb{R}$) : This parameter can be any real number, as the central location of the parameter can be shifted freely between lower (left) and higher (right) values in the log-moneyness space.

5. σ : Controls the curvature of the at-the-money volatility smile.



(Figure 5)

Logical parameter constraint ($\sigma > 0$) : This parameter represents a level of variance which flattens the at-the-money curvature as variance increases. This parameter is constrained to positive values for practicality (although negative convexity isn't a theoretical impossibility).

The final constraint is that total variance must be positive $a + b\sigma\sqrt{1-p^2} \geq 0$

Again, options naturally provide value to their owners ("longs") and therefore, by definition, the total variance must be positive.

SVI Calibration in Practice

Liquidity Considerations

Any options practitioner will know that option implied volatility is a byproduct of the extrinsic value of the option price. With respect to European options, intrinsic value increases the cost of an option but carries the same extrinsic value, or implied volatility price, as an out-of-the-money option with the same strike. We can prove this using an FX like Put-Call parity formula.

Put-Call Parity

$$C - P = S_0 e^{-r_f T} - K e^{-r_d T} \quad (3)$$

Where

- C is the price of the European call option.
- P is the price of the European put option.
- S is the current price of the underlying asset.
- K is the strike price of the options.
- r is the risk-free interest rate: foreign (f) and domestic (d).
- T is the time to expiration of the options.
- t is the current time.

Knowing this consideration, it's naturally observed that given the choice between using in-the-money (ITM) options or out-of-the-money (OTM) options, using the most liquid subset will enable better calibration without the loss of valuable information regarding implied volatility.

Using this judgment, we begin by excluding all ITM options as a first step in the SVI calibration process.

Calibrating to Prices or Implied Volatilities

Practitioners can choose to calibrate the SVI model by either minimizing errors in the objective function with respect to option prices or option implied volatilities.

The trade-off between the two provides a slightly different calibration weight and outcome.

Option prices have the largest extrinsic value when they are exactly at-the-money, essentially providing the most optionality on the underlying asset. Therefore, when calibrating to option prices expensive options will typically have larger price based errors, hence becoming a heavier weight for the objective minimization function.

All else being equal, calibration to option prices will weigh at-the-money options more heavily. On the other hand, using implied volatility for the SVI calibration process will have a more uniform weight across option strikes, ensuring to balance the calibration minimization between both at-the-money strikes and those out-of-the-money wings.

To capture better uniformity we've used implied volatility in order to smoothly calibrate across the entire strike surface.

Using Mid, Mark or Bid-Ask

Our methodology also uses a mix of exchange marks, mids and bid-ask volatilities.

Exchange marks are often an exponentially-weighted-average value of the mid-price volatility and remain present despite bids/offers flashing-away in times of low liquidity.

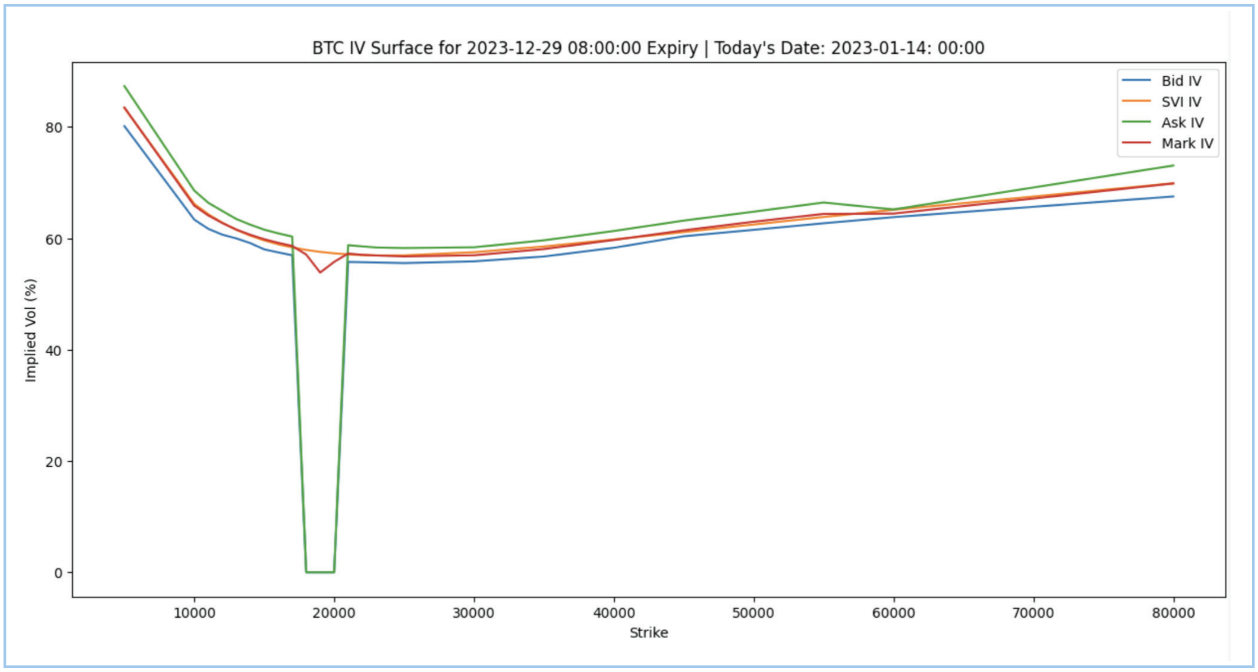
However, exchange marks are not perfect and a cleaning process must be applied in order to select when to calibrate to exchange marks and when to calibrate to mids or bids-ask.

For example, when a trader attempts to execute a trade, they may cross the bid-ask spread in part but not the complete spread (for example trying to buy a call slightly below the asking price). In these situations option price marks, a slower moving average of price mids, will be crossed, meaning the mark price will be below the bid prices. Here, we adjust the volatility to reflect the aggressive market, and then calibrate to the more appropriate bid.

Another important example will be when the market has multiple pulled quotes, meaning neither a bid nor an ask is present in the market.

This can often create a mark-kink, as the exponential weighting scheme of mark prices gets pulled away from the fair market value and slopes down to zero. (See Figure 6)

In the figure below, using our data cleaning methodology, the calibration is able to adjust itself around the mark "kink" and provide a reliable volatility despite the lack of liquidity.



(Figure 6)

Sequential Least Square Quadratic Program (SLSQP)

Once the initial dataset has been prepared for proper calibration, we must guide the calibration process in order to provide consistent, robust and stable calibrations.

Our methodology uses Sequential Least Squares Quadratic Programming (SLSQP) in order to converge the calibration efficiently.

SLSQP is an iterative optimization algorithm that solves nonlinear programming problems with equality and inequality constraints. It is particularly effective for problems where the objective function and constraints can be approximated by quadratic and linear functions, respectively.

Using SLSQP, the calibration can be solved in order to achieve the minimization of errors between the market implied volatility and the model implied volatility.

$$\sum_{i=1}^{n} \left(\sigma_{SVI}\left(k_i\right) - \sigma_{\text{market}}\left(k_i\right)\right)^2 \quad (4)$$

Where:

$\sigma_{\text{market}}\left(k_i\right)$ is the market observed implied volatility f or log moneyness k_i
$\sigma_{SVI}\left(k_i\right)$ is the SVI model implied volatility f or log moneyness k_i

The SLSQP algorithm solves this nonlinear optimization problem by iteratively solving quadratic programming subproblems.

1. **Objective Function:** Defined as the sum of squared residual errors (SSR) between the market and model implied volatilities

$$f(\theta) = \sum_{i=1}^{n} \left(\sigma_{SVI}\left(k_i\right) - \sigma_{\text{market}}\left(k_i\right)\right)^2 \quad (5)$$

2. **Constraints:** Incorporating the constraints to ensure valid SVI parameters, as previously described
   - $a \in \mathbb{R}$
   - $b \geq 0$
   - $|p| < 1$
   - $m \in \mathbb{R}$
   - $\sigma > 0$
   - $a + b\sigma\sqrt{1 - p^2} \geq 0$

3. **Quadratic Programming Subproblem:** At each iteration k i SLSQP solves the following Quadratic Programming subproblem

minimize $\nabla f(\theta)^T d + \frac{1}{2} d^T H_k d$ (6)

subject to
$$\nabla c_i(\theta_k)^T d + c_i(\theta_k) = 0, i = 1, \ldots, m$$
$$\nabla e_j(\theta_k)^T d + e_j(\theta_k) \leq 0, j = 1, \ldots, p$$

Where $c_i(\theta_k)$ and $e_j(\theta_k)$ represent the equality and inequality constraints, respectively.

4. **Broad Algorithm Overview**

a. Initialization: Start with an initial guess and set $k = 0$

b. Step calculation: Solve the subproblem to find the step direction d_k

c. Line Search: Perform a line search to determine the optimal step size.

d. Update: Update the parameter and the Hessian approximation H_{k+1}

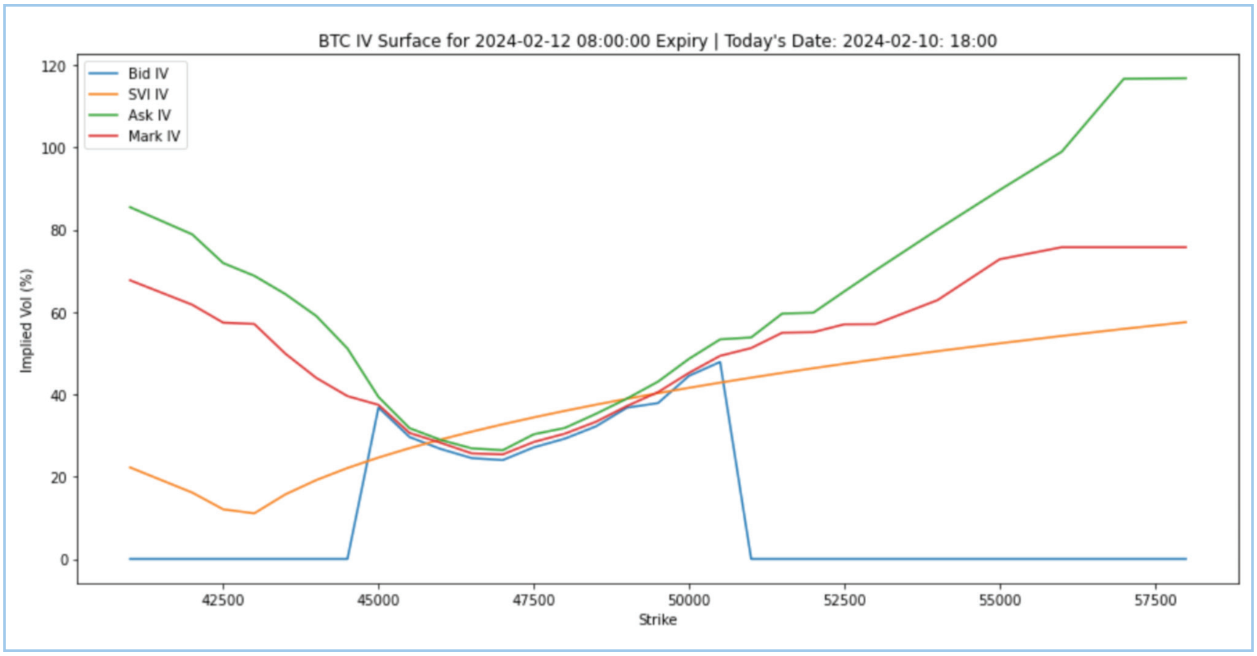
e. Convergence Check: based on predefined criteria

5. **The algorithm terminates** when the gradient norm of the objective function is below a threshold. This threshold can be set as a step size or something else.

It's important to understand what is happening during the convergence process, because we can clearly see how the SLSQP can obtain a nonsensical convergence if the initial starting point is poorly chosen.

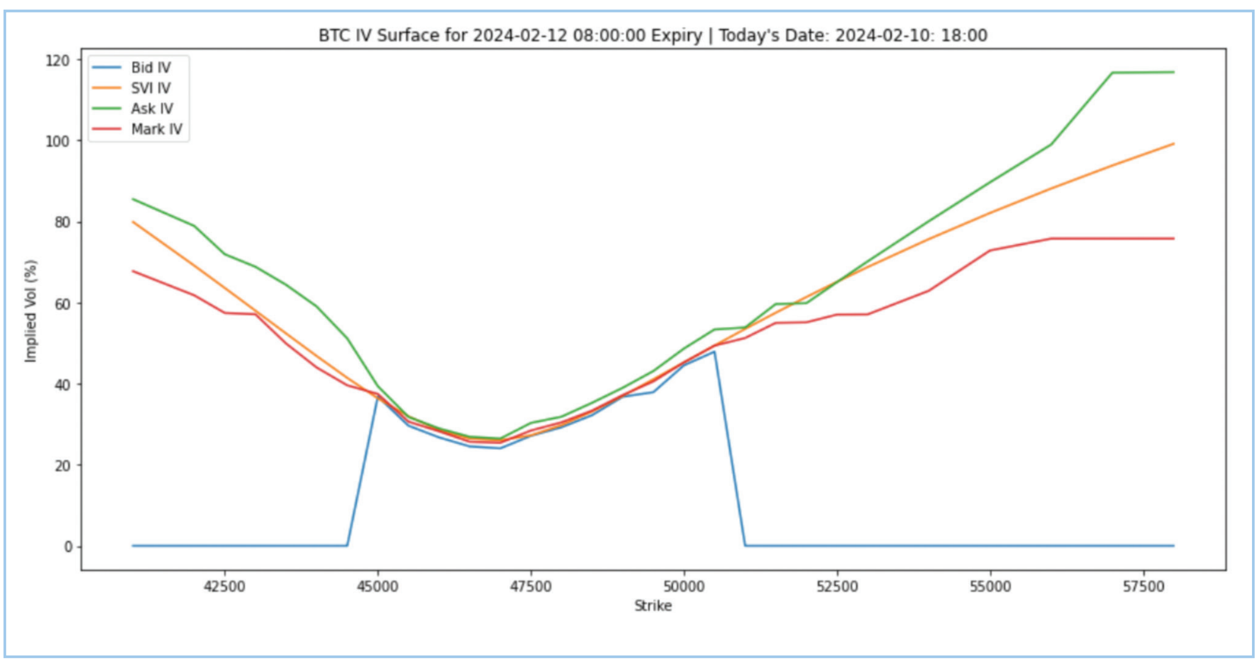
Once SLSQP attains the objective function threshold during its convergence process, it believes that it has attained the objective function minima.

This explains the local minima problem and it is often the source of complexity when performing SVI calibration. We will explore how our real-time solution addresses this problem later.



(Figure 7)

The above example in (Figure 7) displays the classic SVI calibration problem when using SLSQP with bad initial parameter guesses. As the SLSQP minimizes the objective function of reducing errors between the observed market implied volatility and the SVI model implied volatility, it has found a local minima, as opposed to the global minima found in the figure below.



(Figure 8)

Being able to do a rough volatility categorization enables us to find which initial parameter values are most sensical from our proprietary matrix of initial values, which then is used to guide SLSQP into a successful calibration as shown in (Figure 8).

Initial Value Matrix

Having explored the parameters and their individual impacts on the shape of the surface calibration and the convergence process using SLSQP, we can now put everything together for a consistent, stable, and robust SVI calibration.

Our methodology uses high quality historical data in order to categorize the various regimes, for selected cryptocurrencies, to establish the multitude of shapes of the volatility surface.

Each cryptocurrency surface is divided into many distinct rough regime types, resulting in a plethora of different initial parameter subgroups. From these, we create a matrix of initial values for each surface type. As a first step, this 'rough' calibration categorization provides a logical initial parameter values, taking into account how the different parameters impact the slices.

This matrix is a key feature of our methodology and is downstream of the initial rough calibration, another aspect of the process.

Conclusion of Calibration

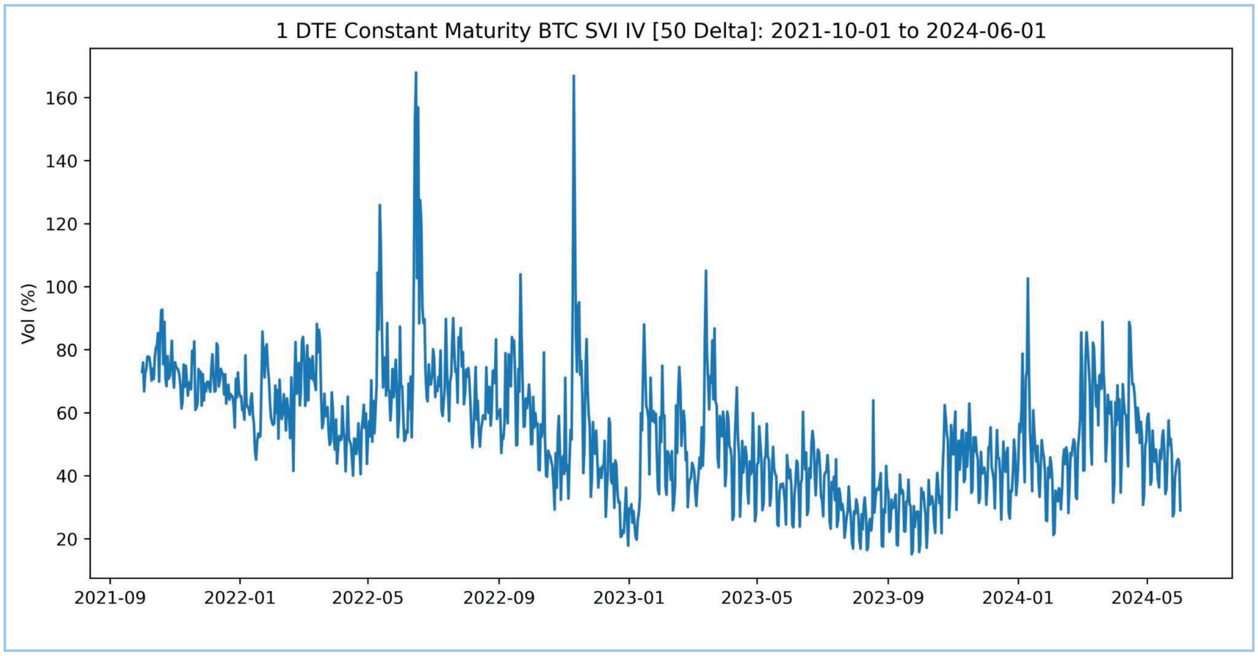
The end result is that our methodology can fit a wide variety of maturities in many different environments in a consistent, stable and robust manner.

In the figures below we can see our Bitcoin volatility SVI calibrations from September 1st, 2021 to June 1st, 2024. This three year period includes notable volatility events such as; the Terra Luna crash, 3AC crash and the FTX bankruptcy.

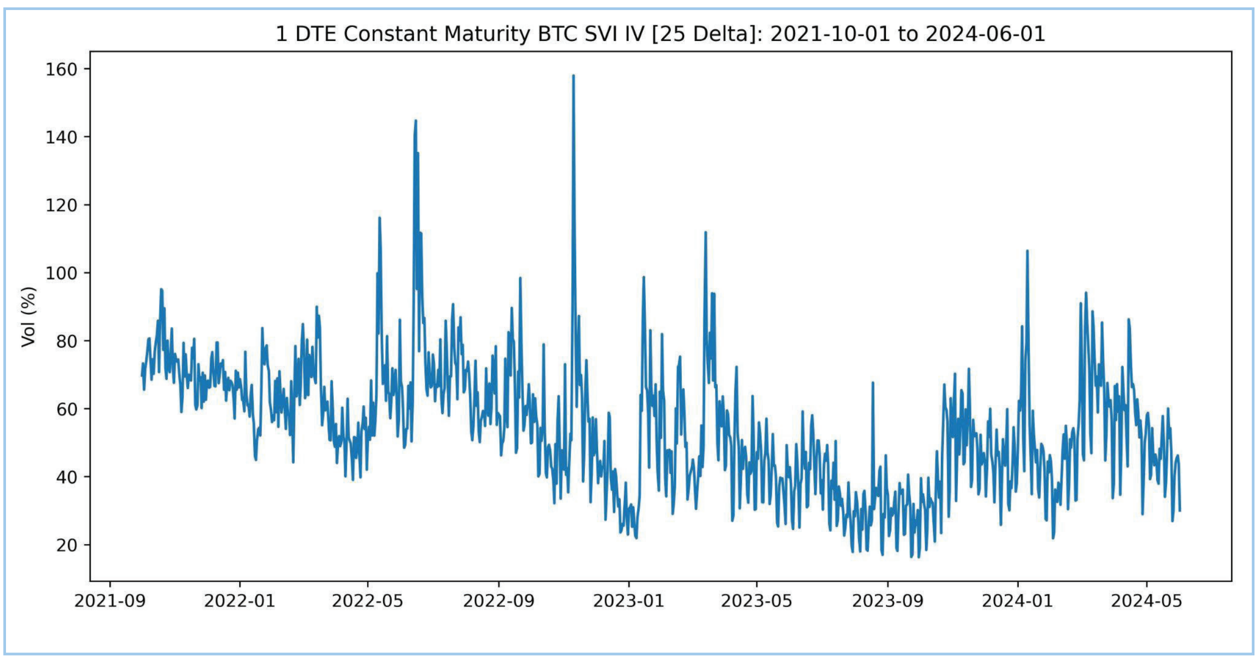
It also includes bullish moments such as the SVB banking crisis, the SEC's spot ETF approval and breakouts to new all-time highs.

One of the most exceptional aspects of our methodology is that we see consistent calibration results for a wide range of maturities ranging from 1-day to 180-days.

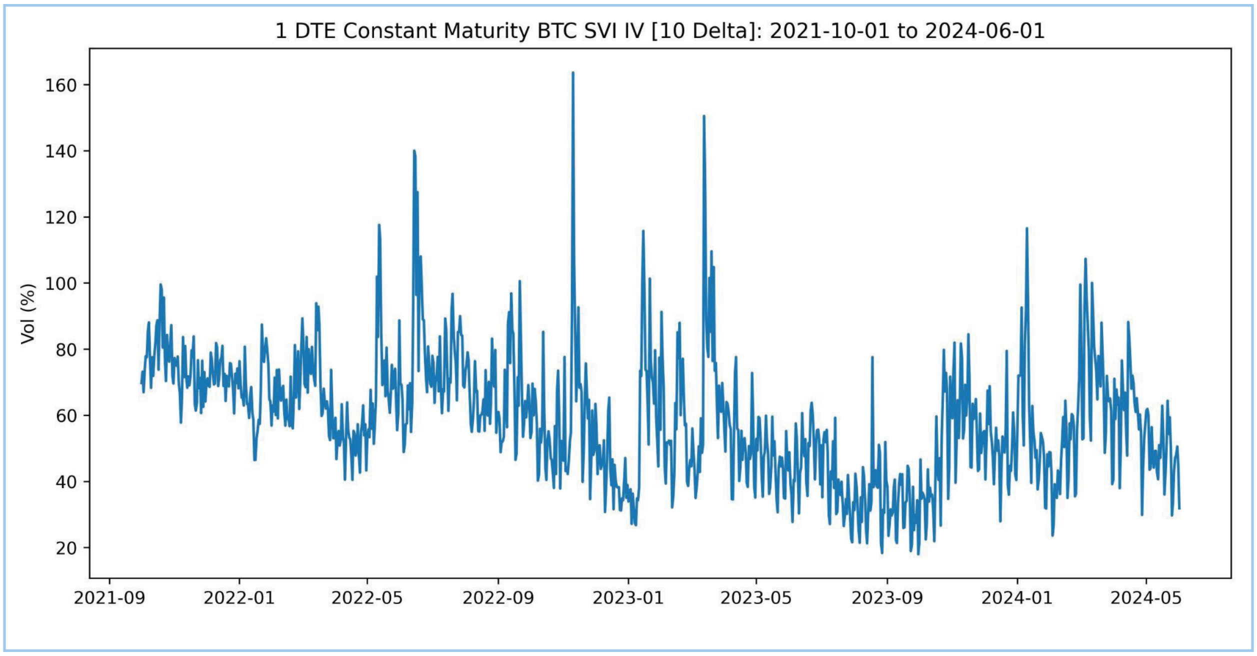
The figures below display the calibration values for various maturities and three distinct target deltas; Δ50, Δ25, and Δ10.



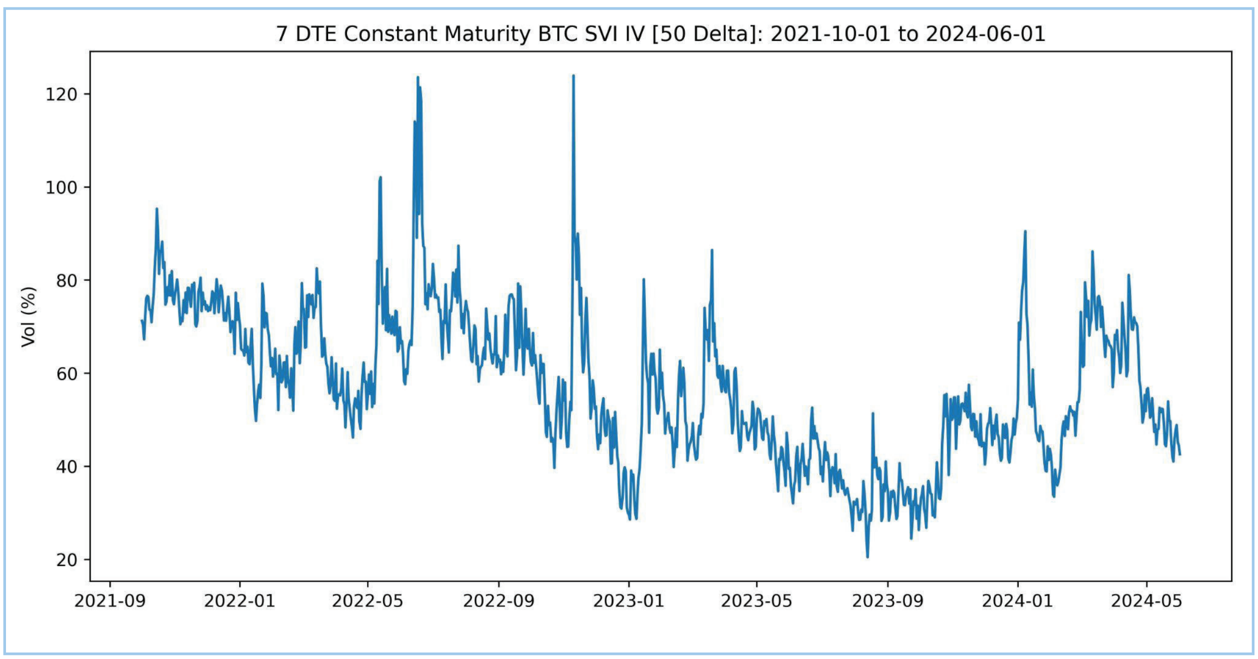
(Figure 9a)



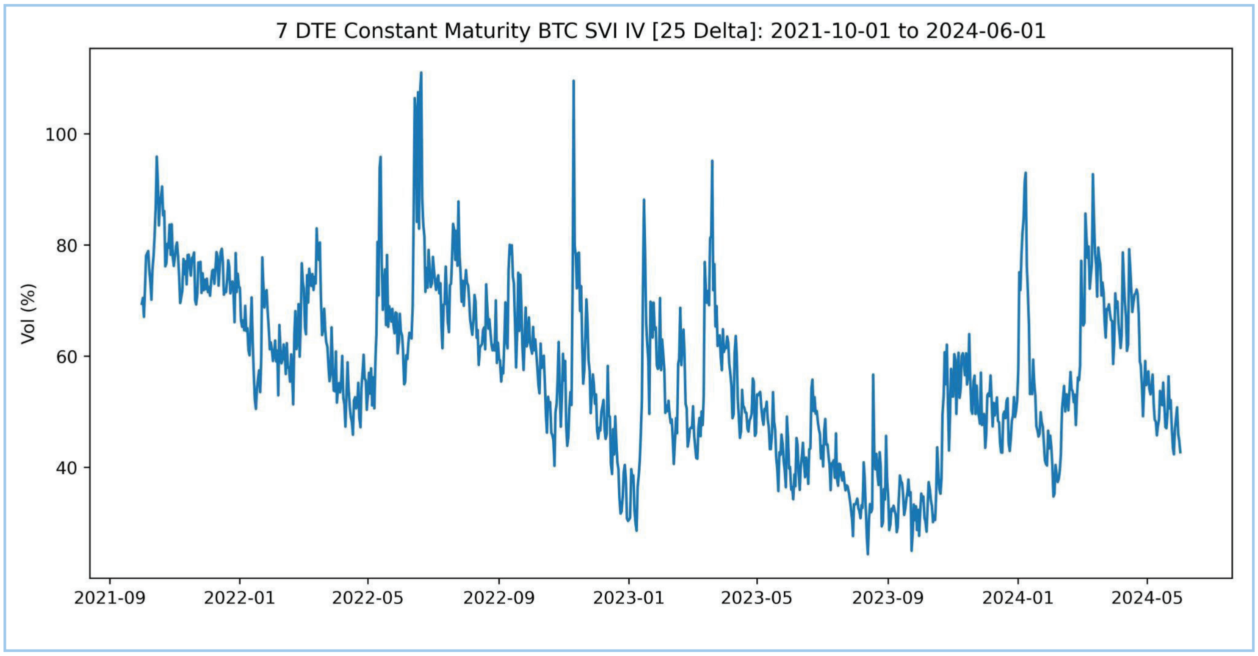
(Figure 9b)



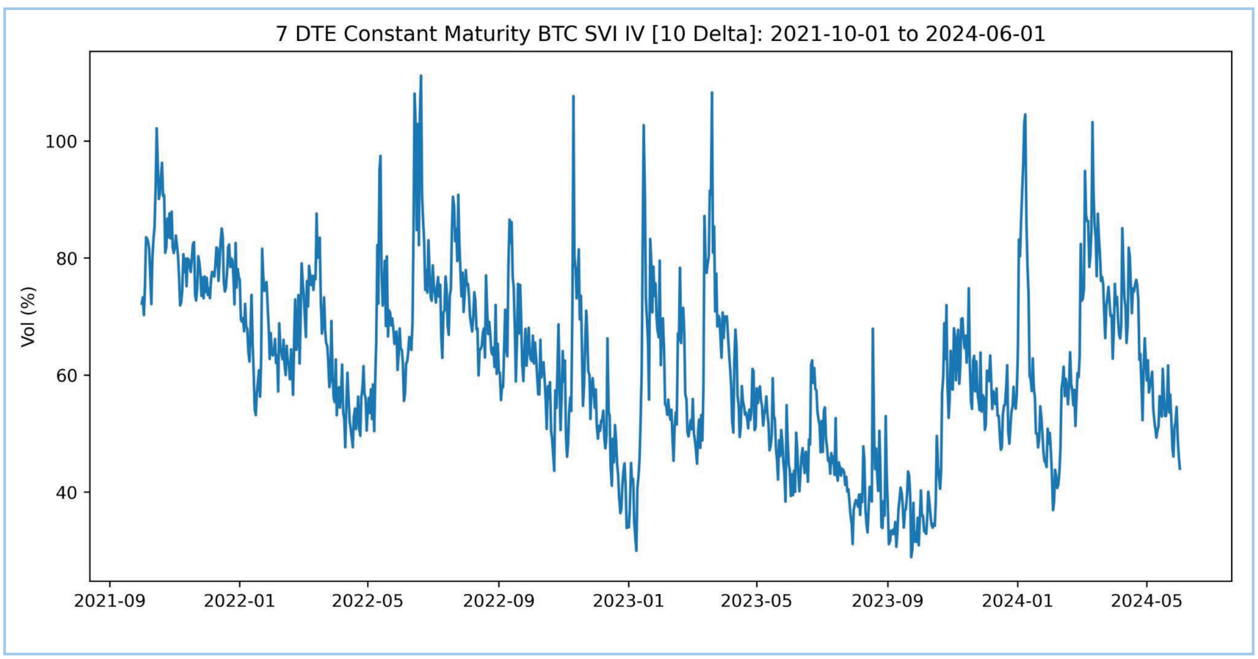
(Figure 9c)



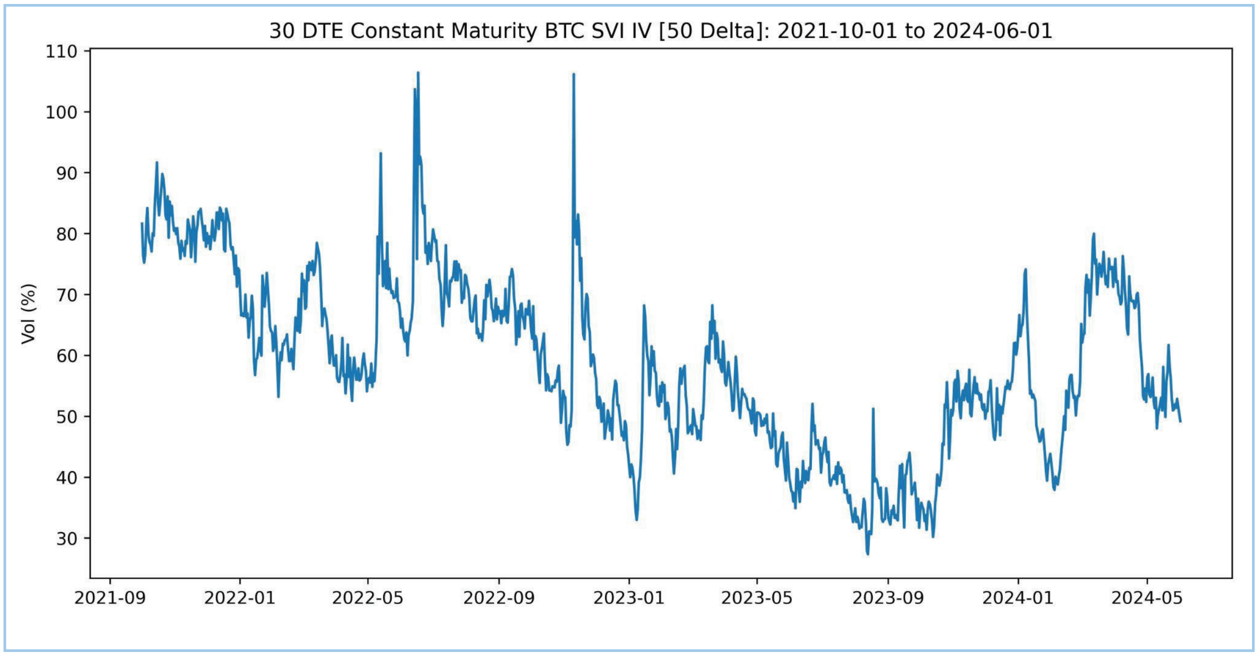
(Figure 10a)



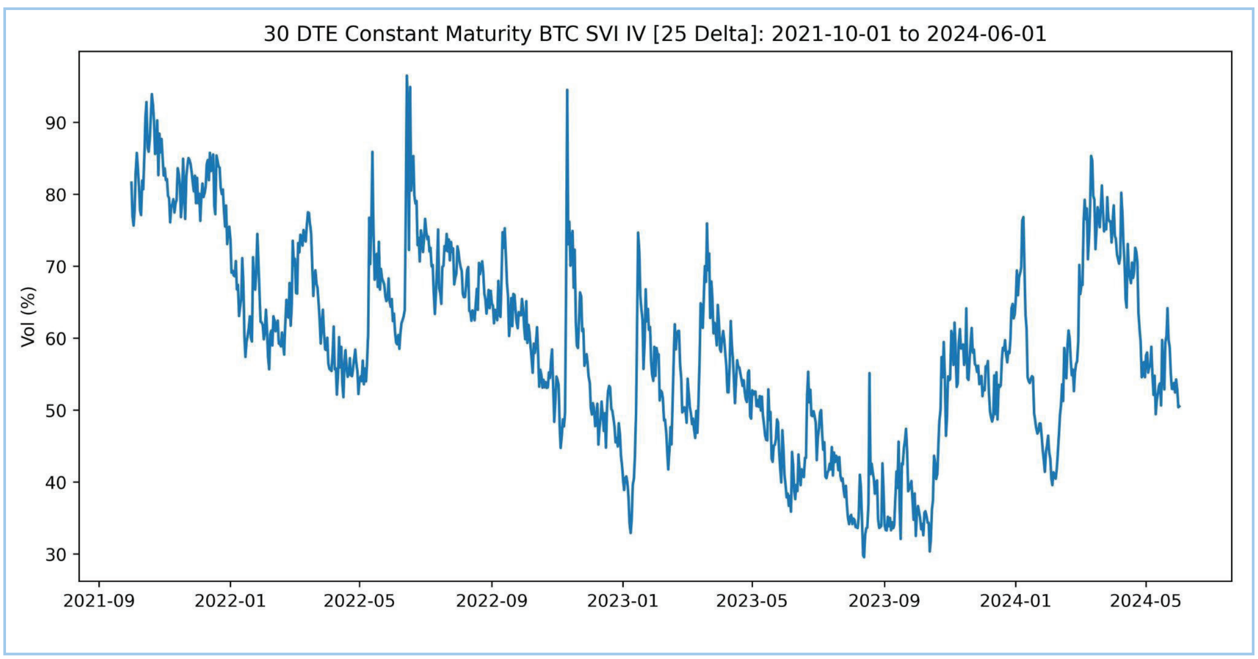
(Figure 10b)



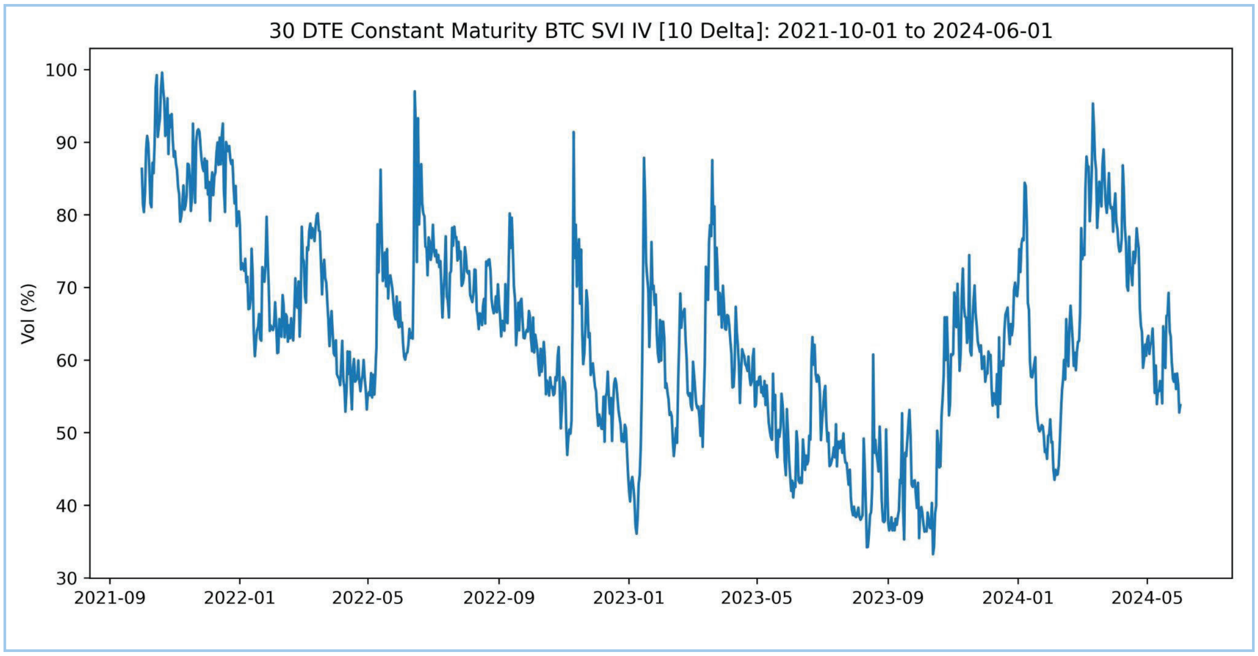
(Figure 10c)



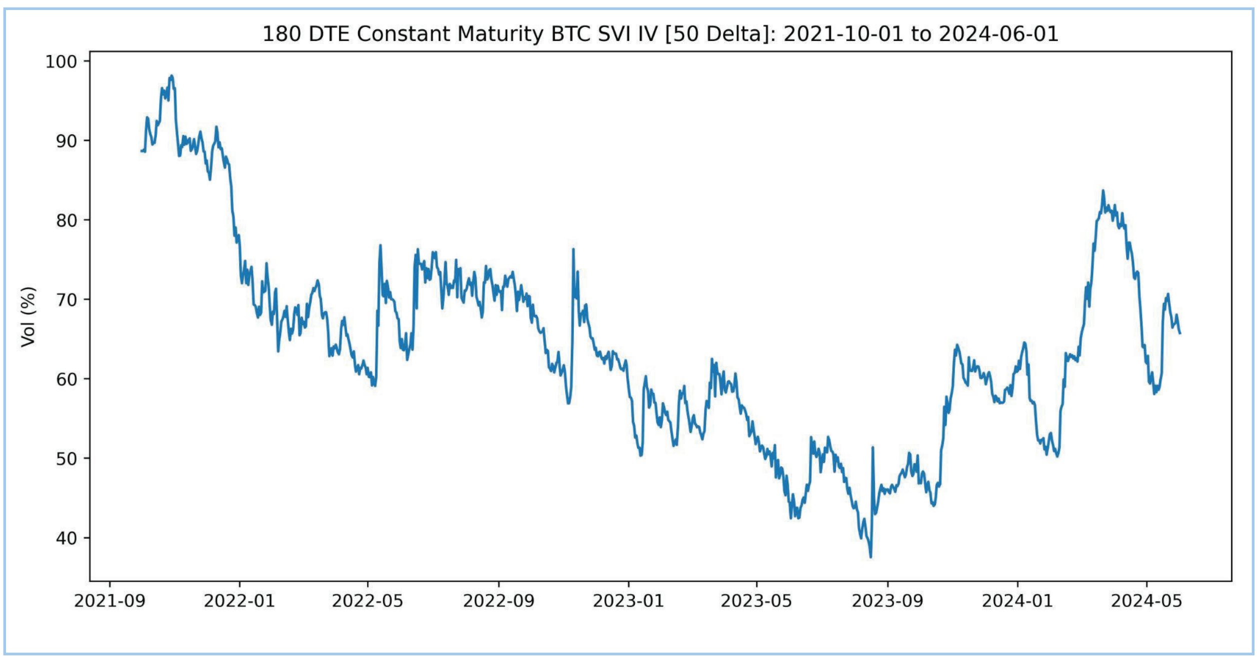
(Figure 11a)



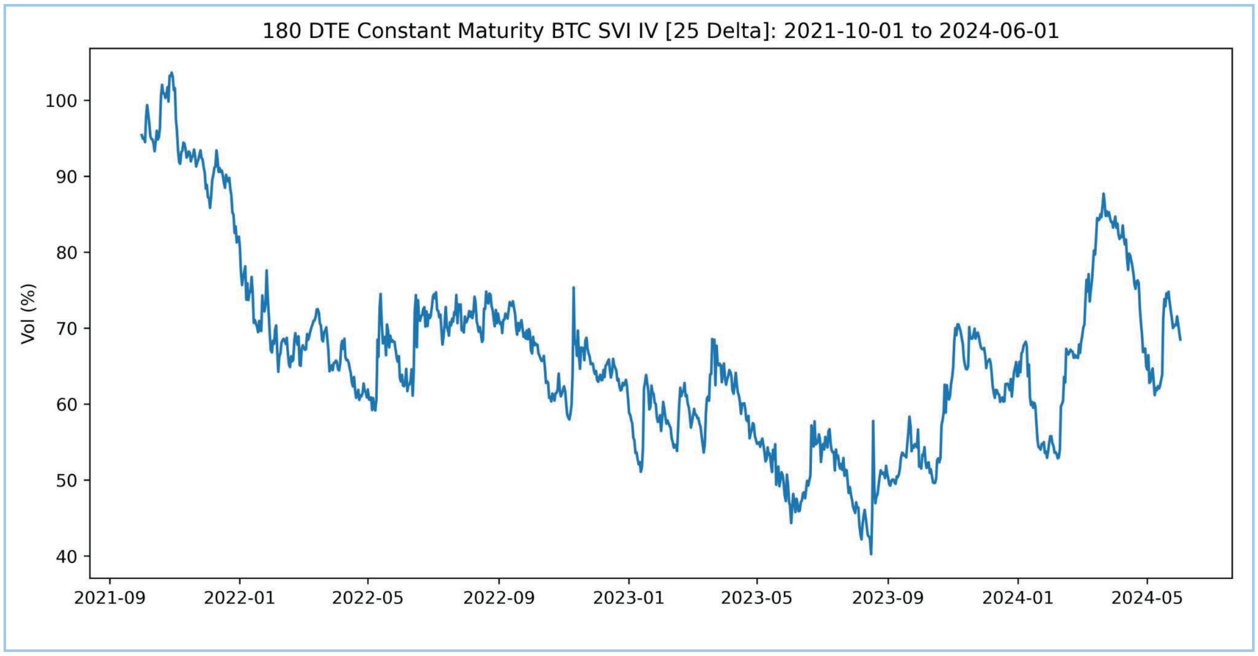
(Figure 11b)



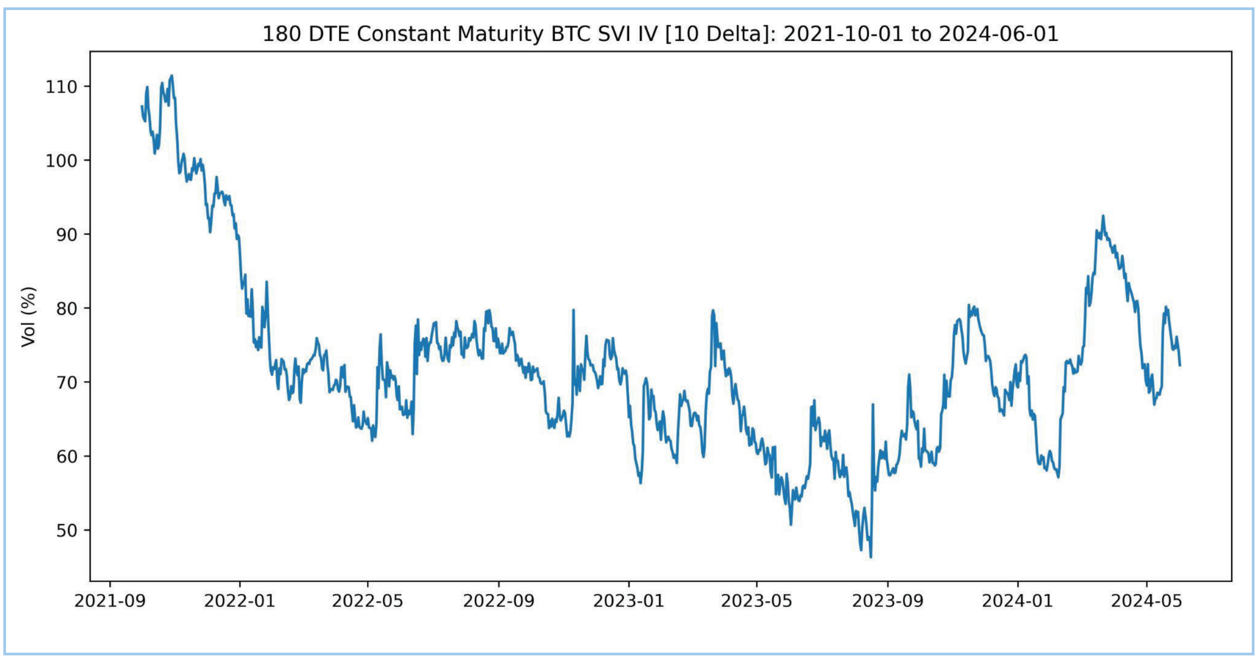
(Figure 11c)



(Figure 12a)



(Figure 12b)



(Figure 12c)

Real Time Calibration of the SVI_{TrueLine}

As demonstrated in our previous discussion on historical calibration processes, numerous challenges can arise when calibrating the volatility surface using SVI. These include data issues, local minima confusing the SLSQP algorithm, and unrealistic rates of change in the wings of the volatility surface.

To ensure that our high-frequency, real-time SVI solution consistently delivers accurate calibrations, we need to integrate a robust feedback evaluation mechanism into the calibration process.

Data Quality Evaluation

The first step in the calibration process is to ensure that we have a complete set of option quotes. It is crucial to verify that the most recent dataset provides a comprehensive option chain. This includes an automated check to ensure that the instrument count for selected expirations is complete and that all necessary options are present. Additionally, we need to confirm that each option has corresponding marks, bids, and asks for us to use in the fitting process.

In fast markets, option chains often "flash away" as market-makers widen their spreads or withdraw their quotes altogether. In the event of incomplete data or an outright outage, the Amberdata SVI_{TrueLine} process will review the next most liquid options exchange with similar instruments.

For example, Deribit is the most liquid options exchange, and their BTC and ETH instruments are inverse options. The next most similar exchange would be OKX, which also offers BTC and ETH inverse options. Therefore, if Deribit experiences an outage, Amberdata SVI_{TrueLine} will seamlessly switch to using OKX quotes. If OKX also suffers an outage, Amberdata will then retrieve data from the next most liquid exchange, Bybit, and continue down the list as needed.

Data Fail-Safe

In the worst-case scenario, when all exchanges experience simultaneous outages or market-makers withdraw their quotes from all exchanges, the Amberdata SVI_{TrueLine} will revert to the most recent complete data fit and hold that fit constant until the market returns to an active and quoted state.

The Amberdata SVI_{TrueLine} also includes a confidence function based on quality of data. This function adjusts its score with respect to age of data as well as the volatility of the environment.

If the data is outdated but the market hasn't moved significantly, the impact on the confidence score may be minimal. Conversely, in very volatile markets, even slight delays in data can quickly erode confidence in the data's relevance.

Calibration Evaluation

Once a sustainable set of quote data has been collected and passes the evaluation checks for the calibration process, we will use our proprietary parameter matrix. This matrix has been established by quantifying and categorizing the various volatility surface shapes for the underlying coin in question (e.g., BTC).

$SVI_{\mathbf{TrueLine}}$ starts every calibration process by estimating the current shape of the volatility surface, a rough estimate. This allows the process to quickly find suitable initial parameter values and proceed with the calibration.

After the initial calibration, we need to evaluate the "quality of fit" for the Amberdata $SVI_{\mathbf{TrueLine}}$. This involves various measurement techniques, for example, assessing the absolute bid-ask violations, a vega-weighted error rate, etc. to ensure the quality of fit.

If the initial calibration process produces a poor result, we then shift to an alternate initial parameter matrix. This matrix contains additional variables for enhanced guidance and precision. The Amberdata $SVI_{\mathbf{TrueLine}}$ real-time solution will then attempt a recalibration process using the alternate initial parameters, which will undergo the same quality of calibration evaluation.

Calibration Fail-Safe

Should the recalibrated process fail the evaluation process once again, the Amberdata $SVI_{\mathbf{TrueLine}}$ real-time solution will then serve the last known high quality calibration.

The confidence score will be penalized based on the time gap between now and the last quality fit. Volatility between the fallback calibration and the current timestamp will also degrade the confidence score, while a low volatility environments will keep the confidence interval high.

Data Signatures

EIP-712 is a standard for hashing and signing structured data in Ethereum, enabling more secure and user-friendly interactions with smart contracts. It defines a structured approach to create a digest of the data that users will sign, ensuring that the meaning of the data is clear and unambiguous. This standard helps prevent signature replay attacks by including a domain separator and a message schema, thus making signatures context-specific. By leveraging EIP-712, $SVI_{\mathbf{TrueLine}}$ can include cryptographic signatures alongside the data payloads, ensuring that the data integrity and origin can be verified.

ECDSA (Elliptic Curve Digital Signature Algorithm) is a cryptographic algorithm used to ensure the authenticity and integrity of data. It works by generating a unique signature for each data payload using the sender's private key, which can then be verified by anyone with the corresponding public key. When combined with EIP-712, ECDSA allows for the creation of secure, tamper-proof signatures for structured data, providing an additional layer of security for data transactions. By adopting these signature schemes, Amberdata can enhance the trustworthiness and security of the $SVI_{\mathbf{TrueLine}}$, giving users confidence in the data's authenticity and integrity.

References

1. Gatheral, J. (2004). “A parsimonious arbitrage-free implied volatility parameterization with application to the valuation of volatility derivatives.” SSRN Electronic Journal.

2. Gatheral, J., & Jacquier, A. (2011). “Convergence of Heston to SVI.” Quantitative Finance, 11(8), 1129-1132.

3. Gatheral, J., & Jacquier, A. (2014). “Arbitrage-free SVI volatility surfaces.” Quantitative Finance, 14(1), 59-71.

4. Boggs, P. T., & Tolle, J. W. (2000). “Sequential Quadratic Programming.” Acta Numerica*, 4, 1-51.

5. Nocedal, J., & Wright, S. J. (2006). “Numerical Optimization.” Springer.

6. Gatheral, J., Jaquier, A., & Hsu, E. P. (2018). “Volatility is rough.” Quantitative Finance, 18(6), 933-949.

7. Gatheral, J., & Jacquier, A. (2011). “An explicitly solvable stochastic volatility model.” Quantitative Finance Papers.

8. Fukasawa, M. (2011). “Asymptotic analysis of stochastic volatility models.” SIAM Journal on Financial Mathematics, 2(1), 439-468.

9. Heston, S. L. (1993). “A closed-form solution for options with stochastic volatility with applications to bond and currency options.” The Review of Financial Studies, 6(2), 327-343.

10. Andersen, L., & Piterbarg, V. (2007). “Moment explosions in stochastic volatility models.” Finance and Stochastics, 11, 29-50.

11. Gatheral, J. (2006). “The Volatility Surface: A Practitioner’s Guide.” Wiley Finance.

12. Kahl, C., & Jäckel, P. (2005). “Fast strong approximation Monte Carlo schemes for stochastic volatility models.” Quantitative Finance, 5(1), 1-12.

13. Hull, J., & White, A. (1987). “The Pricing of Options on Assets with Stochastic Volatilities.” The Journal of Finance, 42(2), 281-300.

14. Fouque, J. P., Papanicolaou, G., & Sircar, K. R. (2000). “Mean-Reverting Stochastic Volatility.” International Journal of Theoretical and Applied Finance, 3(1), 101-142.

15. Lewis, A. L. (2000). “Option Valuation under Stochastic Volatility.” Finance Press.

16. Broadie, M., & Kaya, O. (2006). “Exact simulation of stochastic volatility and other affine jump diffusion processes.” Operations Research, 54(2), 217-231.

17. Medvedev, A., & Scaillet, O. (2007). “Approximation and calibration of short-term implied volatilities under jump-diffusion stochastic volatility.” Review of Financial Studies, 20(2), 427-459.

18. Cont, R., & Tankov, P. (2004). “Financial Modeling with Jump Processes.” Chapman and Hall/CRC.

19. Jäckel, P. (2002). “Monte Carlo Methods in Finance.” Wiley.

20. C. Martini and S. De Marco “Quasi-Explicit Calibration of Gatheral’s SVI model” Zeliade Systems

21. Steven L. Heston “A Closed-Form Solution for Options with Stochastic Volatility with Applications to Bond and Currency Options.”

22. Amberdata Research “Developing and Backtesting DEX/CEX Arbitrage Trading Strategies”

23. QuantNext founder Raphaël DANDO for his excellent insights and outstanding product. Bio: QIS Structurer at BNP Paribas. Founder of Quant Next.

amberdata

amberdata.io
docs.amberdata.io
hello@amberdata.io