

The Volatility Curve Estimation Algorithm

1 General information

Volatility curve (implied *volatility curve*) is calculated for the call and put options on the same underlying asset with the same maturity date.

Curves freeze 10 minutes before both evening and intraday clearing sessions.

Hereinafter settings are written in **bold**.

1.1 Data

The data for curve calculating are:

- bids and asks, their volumes and lifetimes for each option series
- T - a time to maturity in years
- F - a current futures quotation

Orders are filtered with regard to lifetime and volume. Then best bid and ask is defined.

Best bids and asks are recalculated in to the implied volatilities σ by applying a numerical algorithm. For this purpose we use Black formula for options with futures-style settlement method on futures:

$$C = F \cdot N(d_1) - Strike \cdot N(d_2)$$

$$P = C - F + Strike$$

Where C , P - call and put option price respectively,

$$d_1 = \frac{\ln\left(\frac{F}{Strike}\right) + 0.5\sigma^2 T}{\sigma\sqrt{T}} \quad d_2 = \frac{\ln\left(\frac{F}{Strike}\right) - 0.5\sigma^2 T}{\sigma\sqrt{T}}$$

$$N(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{y^2}{2}} dy \text{ -the cumulative distribution function of the standard normal distribution}$$

The implied volatilities are multiplied by 100 for further steps. If there are no bids or asks for options with particular strike and maturity date then implied volatility equals zero.

Finally, there are 4 implied volatilities: *call_bid*, *call_ask*, *put_bid*, *put_ask*.

For each strike we calculate two variables: *bid*, *ask*.

First of all two auxiliary variables are calculated *max_bid*, *min_ask*.

$$\max_bid = \begin{cases} \max(\textit{put_bid}, \textit{call_bid}), & \textit{if there are both orders} \\ \textit{order price}, & \textit{if there is only one order} \\ 0, & \textit{else} \end{cases}$$

$$\min_ask = \begin{cases} \min(\textit{put_ask}, \textit{call_ask}), & \textit{if there are both orders} \\ \textit{order price}, & \textit{if there is only one order} \\ 0, & \textit{else} \end{cases}$$

Then *bid*, *ask* are calculated:

$$bid = \begin{cases} \min(\max_bid, \min_ask), & \textit{if both values don't equal 0} \\ \max_bid, & \textit{If } \max_bid > 0, \min_ask = 0 \\ 0, & \textit{else} \end{cases}$$

$$ask = \begin{cases} \max(\max_bid, \min_ask), & \text{if both values don't equal 0} \\ \min_ask, & \text{if } \min_ask > 0, \max_bid = 0 \\ 0, & \text{else} \end{cases}$$

Remark: If intervals $(call_bid, call_ask)$ and (put_bid, put_ask) don't intersect, i.e. $call_bid > put_ask$ or $put_bid > call_ask$, then according to the formulas above, the interval $bid - ask$ will be the span between these intervals, i.e. $(put_ask, call_bid)$ in first case and $(call_ask, put_bid)$ in second one.

1.2 Curve formula

Volatility curve defined by an equation of the form:

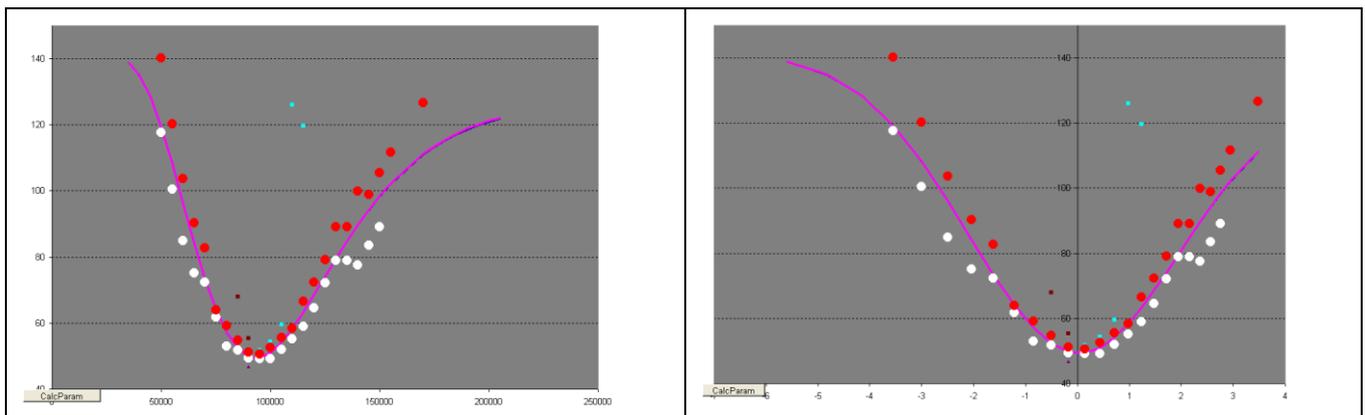
$$\sigma = f(x, s, a, b, c, d, e) = a + b * \left(-e^{-cy^2} \right) \left[\frac{d * \arctg(e * y)}{e} \right],$$

where:

- s, a, b, c, d, e - adjusting parameters (*param(1÷6) hereinafter*)
- $x = \frac{1}{\sqrt{T}} \ln\left(\frac{Strike}{F}\right)$ - horizontal coordinate of curve, which is actually a modified Strike, because T and F are fixed.
- $y = x - s$.

σ is bounded by **curve_min** and **curve_max**.

The volatility curve for RTS is shown on the plots below, with respect to Strike at the left one, and with respect to x at the right one. White and red dots are the bids and asks respectively.



1.3 Boundary conditions

The curve's parameters *param(1÷6)* are bounded by upper-bound constraints **boundary_up(1÷6)** and lower-bound constraints **boundary_down(1÷6)**. All or some constraints can be omitted.

There are four types of curve binding:

If **sample_variant** = 0, then current curve is not bound

If **sample_variant** = 1, then base curve is set manually

If **sample_variant** = 2, then current curve is a base curve for itself

If **sample_variant** = 3, then volatility curve for other options is set as base curve. For instance, volatility curve for the options on RTS-3.10 with expiration in March can be base curve for the options on RTS-6.10 with expiration in April.

The main idea of curve binding is that the parameters of the current curve should meet the restrictions, which are built upon base curve.

Absolute: **sample_deviation_absolute(1÷6)**.
Relative: **sample_deviation_relative(1÷6)**.

The additional parameter **relative_threshold** is also set for relative constraints. If the parameter *param(i)* is less than **relative_threshold** by absolute value, then relative constraints aren't applied.

All data and constraints are checked not to contradict each other.

2 Volatility curve adjusting

The idea of volatility curve adjusting is criterion function Cr minimization.

The optimization can be divided into two stages:

1. Rough adjusting – quasirandom Monte Carlo
2. Fine adjusting – coordinate-wise descent method

2.1 Rough adjusting

This stage is used only if current curve fit current market very poor, i.e. if the value of criterion function is large.

$Cr \geq Cr_{rough}$

The main idea of this method is to shift current parameters on $\xi \cdot 100\%$,

where ξ is 6-dimensional quasirandom variable with uniform distribution on cube $[-1.5; 1.5]$, which are generated from quasirandom 6×16383 Sobol sequence (uniformly distributed on $[0, 1]$ variables)

If the iteration of rough adjusting is successful, i.e. it leads criterion function decrease, then option prices monotonicity is tested.

If prices are monotone then shifted parameters are accepted.

2.2 Fine adjustment

Fine adjustment is performed by coordinate-wise descent method. One cycle of optimization is a step-by-step adjustment at 6 coordinates.

Adjusting algorithm for each coordinate:

1. Set initial step: $Step = dparam(j) \cdot param(j)$
2. Define the direction of movement:
Calculate the criterion functions for the shifted upward and shifted downward parameter. The direction is defined as direction toward lowest criterion function.
3. If the shift on the Step in the defined direction leads criterion function decrease then check the monotonicity.
4. If prices are monotone then go to 2)
5. If prices are not monotone or the criterion function doesn't decrease then divide Step by 2 and go to 2)
6. Repeat cycle while $step > 0.0001 \cdot dparam(j) \cdot param(j)$

2.3 Monotonicity

Two types of monotonicity are tested:

1. General

Prices of put and call options for all strikes, including all that are traded today and will be probably traded next trading day, are calculated using curve under test. Then put prices monotone increase and call prices monotone decrease are tested.

2. Differential

Derivatives of put and call prices for all strikes, including all that are traded today and will be probably traded next trading day, with respect to strike are calculated using curve under test.

$$\frac{\partial C}{\partial Strike} = N'(d2) \cdot \frac{\partial \sigma}{\partial y} - N(-d2) \quad \frac{\partial P}{\partial Strike} = \frac{\partial C}{\partial Strike} + 1$$

$$\frac{\partial \sigma}{\partial y} = 0.01 \cdot (2 \cdot b \cdot c \cdot y \cdot \text{Exp}(-c \cdot y^2)) + \frac{d}{1 + e^2 y^2}$$

Then non-negativeness of all put price derivatives and non-positiveness of all call price derivatives are tested.

2.4 Criterion function

Criterion function is calculated as a sum of exponents of error for all strikes. Errors are weighted with regard to remoteness from central strike by means of descending function.