Interactive Visualization of Volatility Surfaces using the Black-Scholes & Merton Model

Bourdeix Corentin,
Martin William,
Kassi Franck Atte Aka

Outline

1. Implied volatility in the Black-Scholes & Merton model
2. Data fetching
3. Data normalization
4. User interface implementation
5. Deployment
6. Results

The role of implied volatility in financial markets

- Efficient volatility forecast
- Reflect investors beliefs
- Key in option pricing
- Volatility as an asset class

Implied volatility in the Black-Scholes & Merton model

Call price: $\mathrm{c}(t,s) = e^{-\delta\tau} s\,\Phi(d_+(\tau,s)\; - e^{-r\tau} sk\Phi(-\mathrm{d}_-(\tau,s))$

Put price $p(t,s) = e^{-r\tau} sk\,\Phi(d_+(\tau,s))\; - e^{-\delta\tau}\; s\Phi\left(-\mathrm{d}_-(\tau,s)\right)$

$s = stock\ price, t = time, \tau = time\ to\ maturity, k = strike, \delta = dividend\ yield, \sigma = volatility\ of\ the\ underlying,$

$\Phi = cumulative\ standard\ normal\ distribution$ and $\mathrm{d}_\pm = \frac{1}{\sigma\sqrt{\tau}}(log\,\frac{e^{r-\delta}s}{k} \pm \frac{\sigma^2}{2}\tau$

Implied volatility σ_{imp} solves : $O\left(\tau, s, \sigma_{imp}\right) - MP = 0$

Implied volatility in the Black-Scholes & Merton model

Bounds on the call implied volatility:

$$-2\,\Phi^{-1}\left(\frac{1-c}{2}\right) \le \sigma_{imp} \le -2\,\Phi^{-1}\left(\frac{1-c}{1+e^{k}}\right)$$

where c is the price of a call and k its strike, Φ^{-1} *standard normal distribution quantile*

Put/call parity : $c^* = p + e^{-\delta\tau}s - e^{-r\tau}k$

Data fetching

Automate process to obtain data required for computing implied volatility

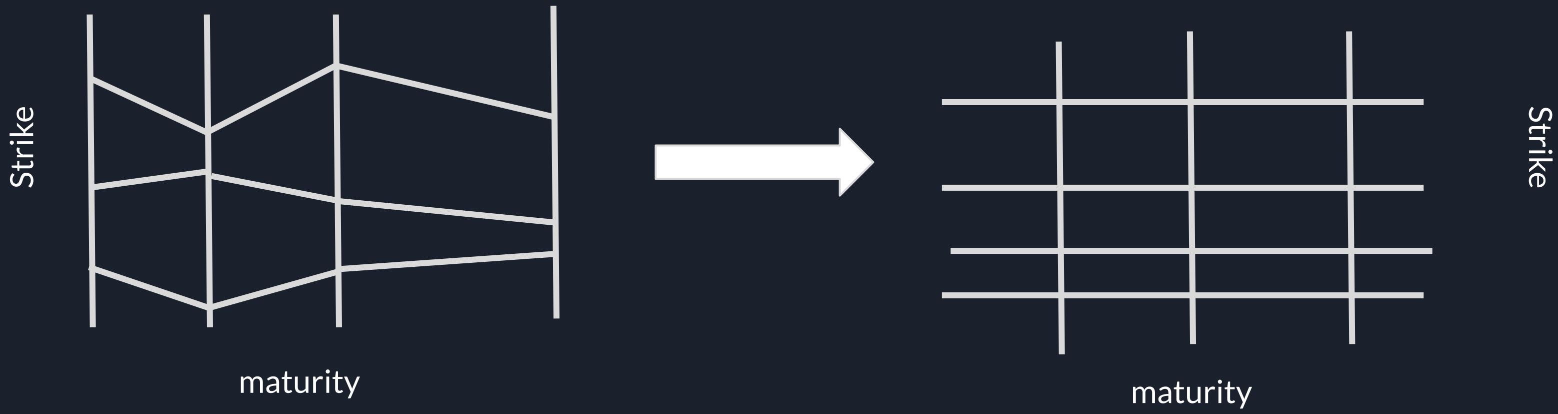
- Options (price, strike, maturity) → *Yahoo Finance* API
- Dividend yield → user input
- Yield curve → web-scraping

Bundled into a *Python* class for replicability and automation of tasks

Data Normalisation

The grid of the maturity strike is :

And we need something like :

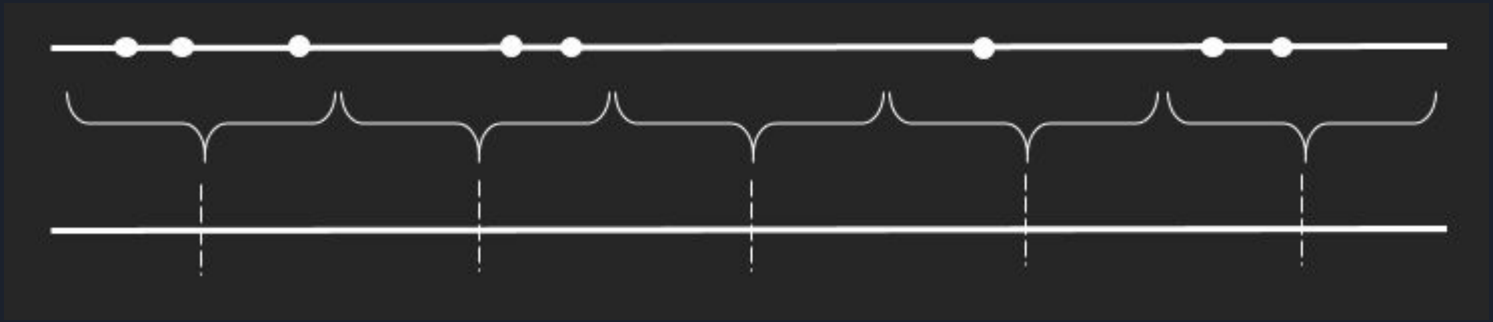


We need to normalize the grid, so the data to be able to plot the 3D surface

Data Normalisation

- We create Strike intervals where we compute the average of implied volatility

Strike data

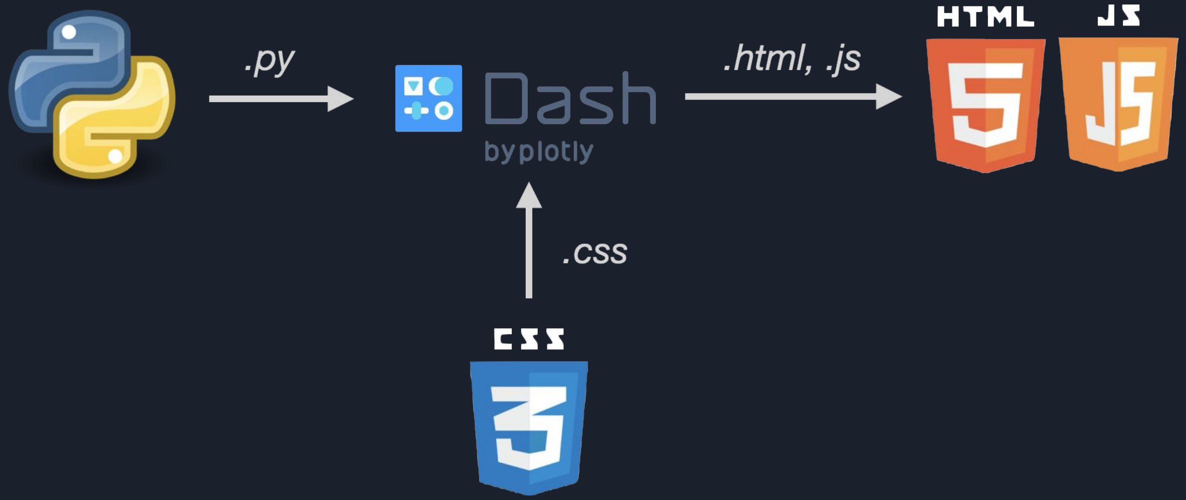


Strike intervals

If the intervals are empty we approximate the volatility by doing a linear regression

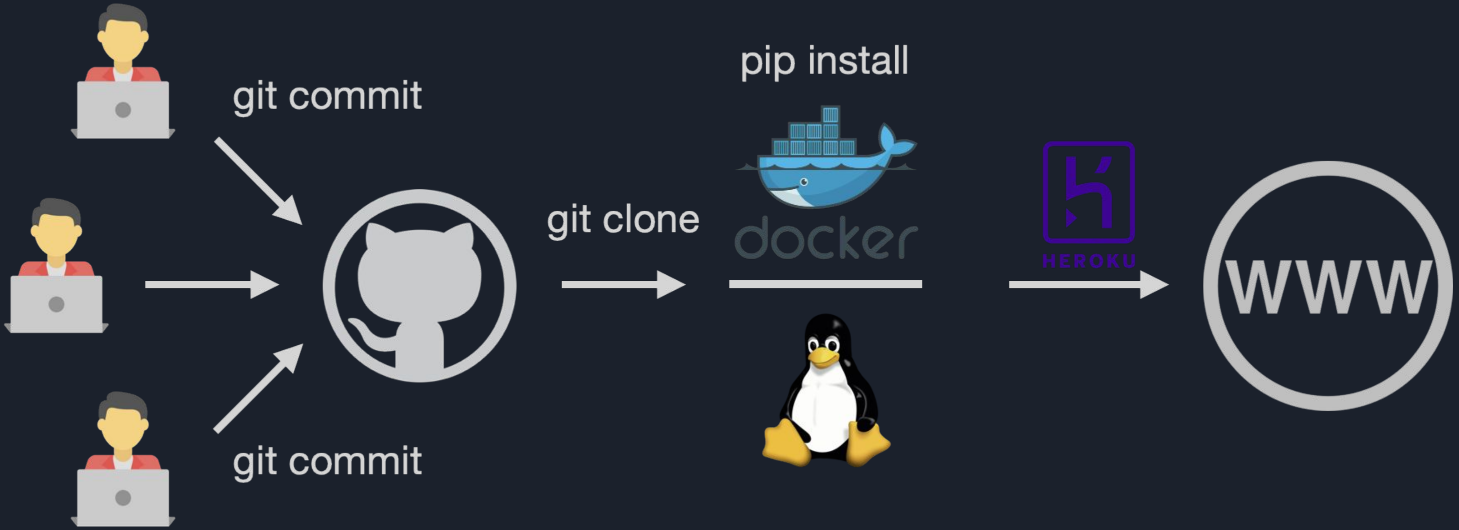
User interface implementation

- Interactive graphical user interface using *Dash* & *Plotly*
- Ability to create web-based interfaces without using *Javascript*

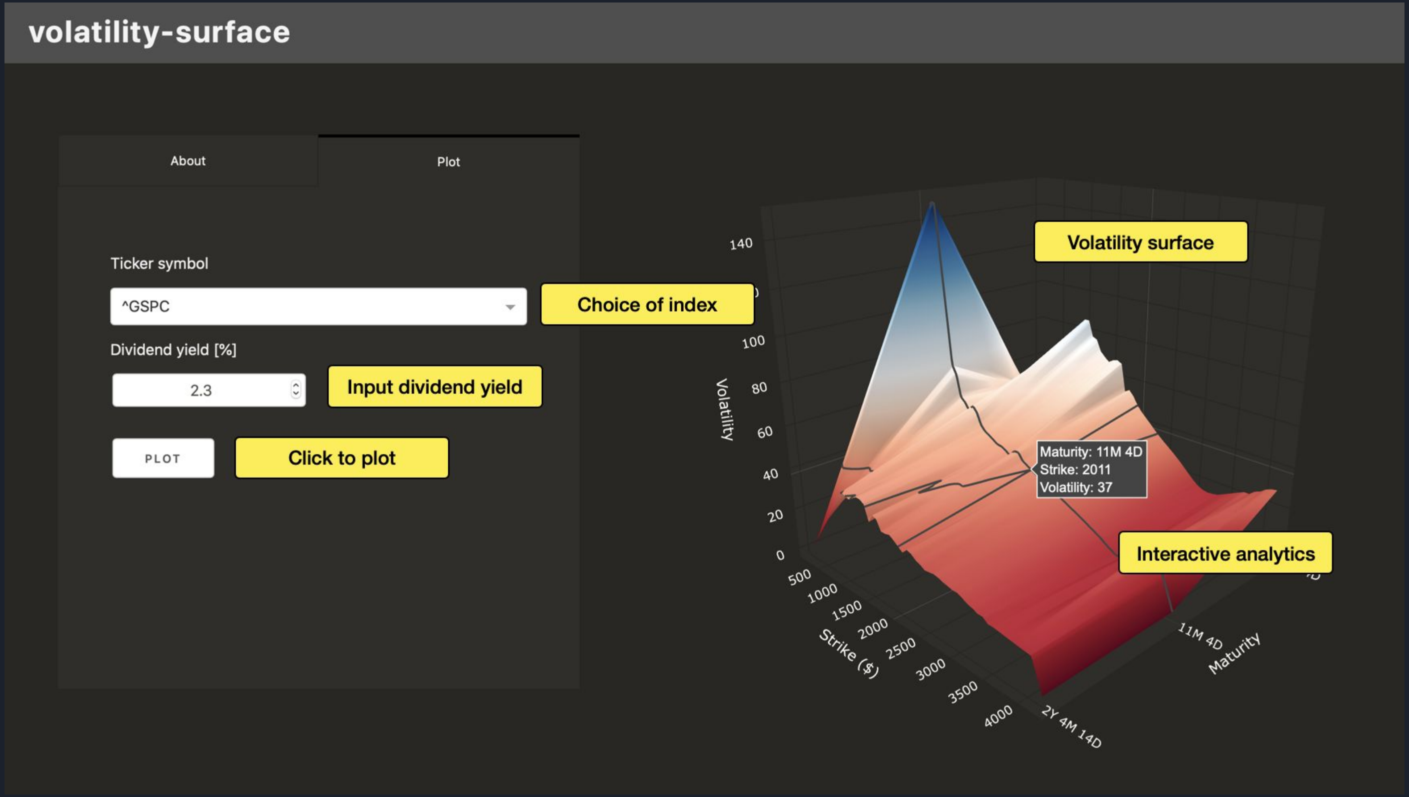


Deployment

- Served on localhost (http://127.0.0.1:8050) at execution
- Open-sourced, hosted and maintained on *Github*
- Packaged for *pip* install
- Ideally deploy on production Linux server (e.g. using *Docker* with *Heroku* or *Gunicorn*)



Results



volatility-surface

Ticker symbol: ^GSPC

Dividend yield [%]: 2.3

Choice of index

Input dividend yield

Click to plot

Volatility surface

Maturity: 11M 4D
Strike: 2011
Volatility: 37

Interactive analytics