Goldman Sachs

India Hackathon 2025

Quant

Presented by : **Arjun Verma**

Indian Institute of Technology Guwahati

Goldman Sachs

Optimal Hedging Strategy

Automated Market Making

Exotic Option Pricing

Quant

1. Optimal Hedging Strategy

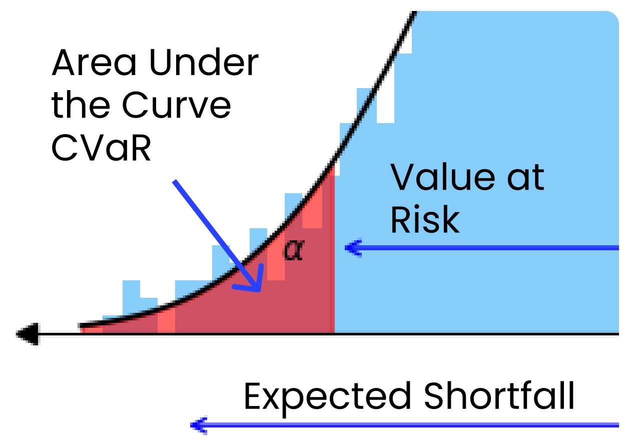
Problem Formulation

Minimize:
Value at Risk at 95% confidence, of the hedged portfolio while keeping the **total capital cost of the hedge low**. Given :

- PnL of last 250 days of the Portfolio.
- Returns and Metadata of **309** Stocks to select from.

Assumptions

- Risk Adjusted return
- Stocks are liquid
- Hedging cost is linear
- PnL is linear in returns



Current Approach Linear Programming

We will Optimize CVaR over VaR !!,
CVaR is convex, unlike VaR and **CVaR captures tail risk**, gives worst-case losses beyond the VaR threshold.

Objective :

$$\text{Minimize } \underbrace{\lambda \sum_{j=1}^{n} C_j (v_j^+ + v_j^-)}_{\text{Total Hedging Cost}} + \underbrace{t + \frac{1}{(1-\alpha)T} \sum_{i=1}^{T} u_i}_{\text{CVaR at level } \alpha}$$

Hedged PnL : $$p_i^{hedged} = p_i + \sum_j (v_j^+ - v_j^-) \cdot R_{ij}$$

Implementation and Observations

Constraint "$Ax \le b$" , Objective $\min_x c^\top x$

$$\underbrace{t - \left[p_i + \sum_j (v_j^+ - v_j^-) R_{i,j}\right]}_{\text{"how far below } t \text{ is day } i\text{?"}} \le u_i \;\&\; x = \begin{bmatrix} v^+ \\ v^- \\ u \\ t \end{bmatrix}$$

We initialise matrices **A, b, x**, and **C,** and use SciPy's linear solver to obtain the optimal x

λ: a trade-off hyperparameter that balances risk vs. cost

Higher Costs

Better CVaR

Future Improvements Using Lasso CV

LassoCV offers a simpler, faster alternative controlling PnL and cost through **L2 and L1-penalty, outperforming previous method.**

$$\min_{\beta} \underbrace{\frac{1}{2T} \left\| \mathbf{p} - R\beta \right\|_2^2}_{\substack{\text{(Hedging error)} \\ \text{unhedged PnL loss}}} + \underbrace{\alpha \left\| \beta \right\|_1}_{\substack{\text{(Hedging cost)} \\ \text{sparse positions}}}$$

Goldman Sachs

Optimal Hedging Strategy

Automated Market Making

Exotic Option Pricing

Quant

2. Automated Market Making

Problem Formulation

1. At each t_n observe bid/ask price p_n^b p_n^a and sizes.
2. Choose quotes (b_n, a_n)
3. If a trade hits your quote, buy or sell at quote price and update inventory

Constraint

- Tick Size : $a_n - b_n \geq 0.1$
- | Inventory | ≤ 20

1. Signal Extraction

- **Mid Price** $= \dfrac{p_t^b + p_t^a}{4} + \dfrac{\mu_t}{2}$
- **Volatility** (std of mid prices)
- **Liquidity** $Q_t = s_t^b + s_t^a.$
- **Momentum** EMA slope
- **Inventory Ratio** $= \dfrac{I_{t-1}}{I_{\max}}$
- **Order Book Imbalance**

$$\mathrm{OI}_t = \frac{s_t^b - s_t^a}{s_t^b + s_t^a + \varepsilon},$$

2. Adjustments

- Calculate Spread Δ
- ↑ **Volatility** - ↑ Widen Δ
- ↑ **Liquidity** - ↓Tighten Δ
- **Momentum** ↑/↓ → Quotes shift in trend direction
- **Inventory** ↑ → Quotes pulled toward mid.
- **Imbalance** ↑/↓ → Quotes shift to match orderbook pressure

3. Quote Formation Parameter Tuning

- **bid price** = mid - Δ ± δ
- **ask price** = mid + Δ ± δ
- Execute Trades and Update Inventory
- Strategy includes **tunable parameters** .
- Find the most effective balance between fill rate, PnL, and risk

Reinforcement Learning

DRL replaces fixed parameters by **learning** from orderbook and trade patterns.

- **State:** Prices and sizes & inventory size.
- **Action :** Deciding the spread.
- **Reward Function :** Rewards PnL, Punishes large Inventory

Observations and Results

Price Downtrend Detected → shift bid/ask downward using trend & inventory adjustments to avoid accumulating long positions.
Large Ask-Size Spikes → Interpreted as sell-side walls; strategy tilts quotes downward using the order book imbalance adjustment.

For all tested parameters PnL ranges

70–120

−3
mean inventory

Inventory remained low carrying short positions.

Goldman Sachs

Optimal Hedging Strategy

Automated Market Making

Exotic Options Pricing

Quant

3. Exotic Options Pricing

Problem Formulation

Pricing **36** European options with:

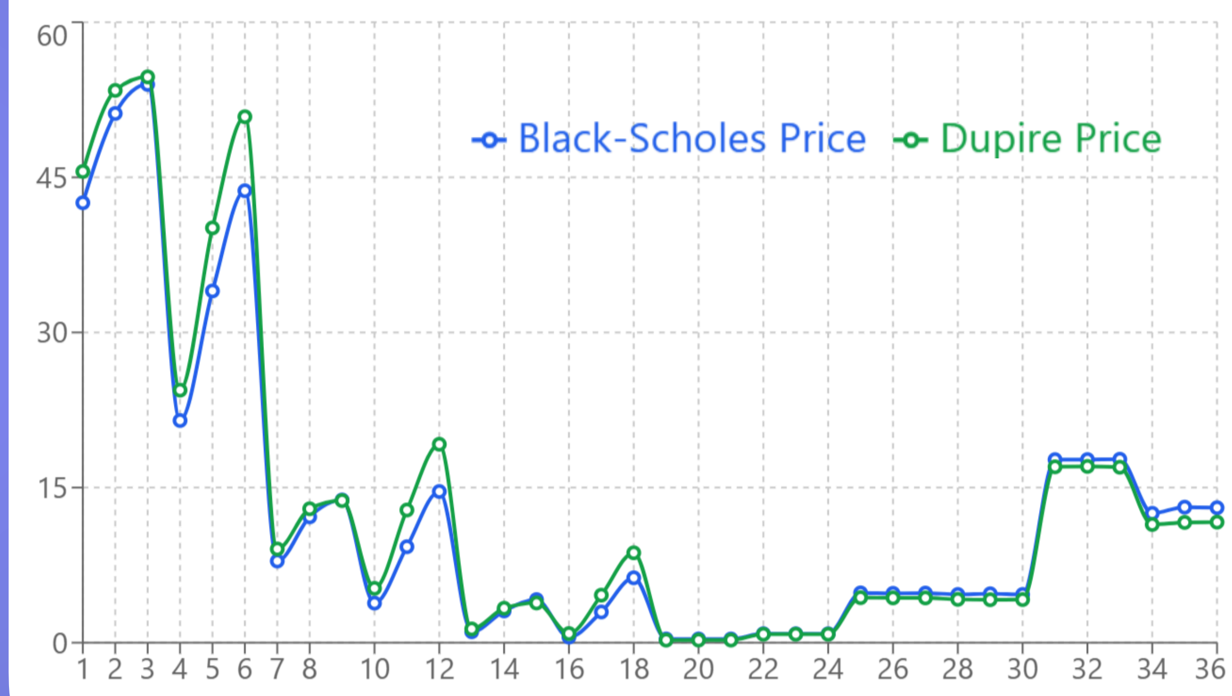
- Knock-out (up and out).
- Equally weighted basket of three stocks **(DTC, DFC, DEC).**
- Correlated risk-neutral model.

Steps Monte Carlo

- Simulate correlated asset paths with local volatility.
- Compute basket values.
- Apply knock-out.
- Calculate discounted payoffs.
- Average over all paths to price the option.

Results Comparison

- Black-Scholes gave better results.
- Error in calculating derivative.



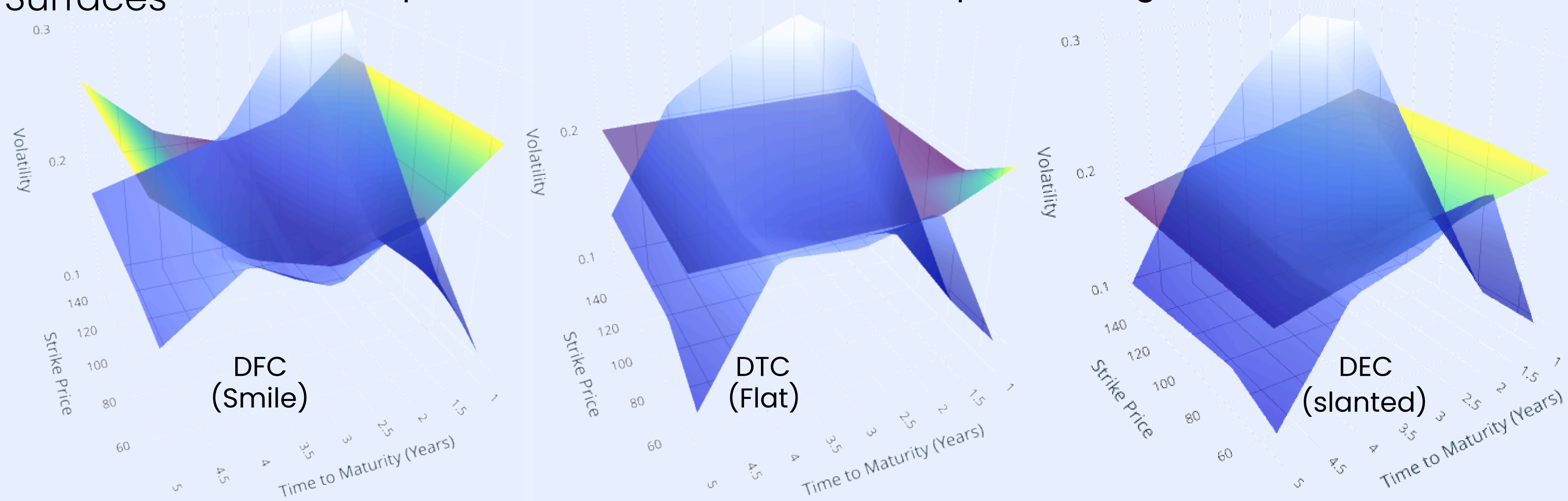
Assumptions

$$dS_t = r\,S_t\,dt + \sigma(S_t, t)\,S_t\,dW_t$$

1. Random Variable $Z \sim \mathcal{N}(0, I)$ and Cholesky factor L
2. Correlated Brownian increments $dW = Z \cdot L^\top \cdot \sqrt{\Delta t}$
3. Discretization $S_{t+\Delta t} = S_t + rS_t\Delta t + \sigma_{\text{loc}}(S_t, t) \cdot S_t \cdot dW$

Calibration Local Volatility Surfaces

1. From market price, invert **Black–Scholes** to get the **Implied Volatility**.
2. or Using **Dupire's formula** calculate **Local Volatility** as a function of spot and time and derivatives of spot - time grid.



95% accuracy in pricing the Vanilla Basket options with knock-out using **Black-Scholes model**.

Local Volatility Surface plays important role in pricing **accurately**.

With a bigger real life dataset we could use **Dupire's Formula.**

Thank You