Quant Workbook

Content

1. Greeks

1.1. Delta

Def 1.1.1 Delta Strike. $\Delta_j = \sum_i \frac{dP}{d\sigma_i} \frac{d\sigma_i}{dK_i} \frac{dK_i}{dS_j}$

- $\frac{dP}{d\sigma_i} = \text{vega}_i$
- $\frac{dK_i}{dS_j} = \frac{K_i(S_j + dS_j) - K_i(S_j)}{dS_j}$
- $dS = S * 0.01, d\sigma = 0.01, dK = K * 0.01$

Motivation is when spot varies, $\frac{\text{strike}}{\text{spot}}$ varies (for each underlying), so volatility and prices varies too.

Def 1.1.2 Delta and Hedge. Delta and Hedge difference comes from settlement offset of the underlying, when you buy a stock you don't receive it immediately, but 3 days later. If repo = 0, then

$$S = S'ZC(t, t + \text{settlement offset})$$

- Delta $= \frac{\text{Price}}{\text{S}}$
- Hedge $= \frac{\text{Price}}{\text{S}'}$
- S = quoted level of spot, a forward contract starting now and maturing at settlement offset date 3 days later.
- S' = real price of the stock now.

1.2. Theta

Def 1.2.1. Theta 2. $\text{Theta } 2 = \text{Theta} - \text{rPdt} + \text{S(r} - \text{q)dt}$

- dt = time between pricing date and theta date time (1 day).
- r = interest rate
- P = option price
- q = repo rate
- $-rPdt + \Delta S(r - q)dt$ = premium you get by carrying your hedging from now until theta date.

Def 1.2.2. Theta Repo. $\Theta_{Repo} = \Delta\text{Spot}(1 - \frac{1}{\text{RF}(s_0, s_1)})$

- s_0 = pricing date
- s_1 = theta date
- $RF(s_0, s_1)$ = repo capitalization factor between pricing date and theta date.

Def 1.2.3. Theta Div. $\Theta_{Div} = -\text{Div}$

- Div = amount of dividend to be paid after theta date.

1.3. Vega

Vega is obatined by shocking the matrix of volatility uniformly of 1 point of volatility.

Def 1.3.1. Vega Corrected/Vega Weighted. $\text{Shock} = \frac{\text{Vol}}{\text{VolRef}} \frac{1}{\sqrt{\text{T}}}$

- VolRef = 25 (const).

Def 1.3.2. Vega Rotation Maturity. $\text{Shock} = \frac{\text{T} - 1}{\sqrt{\text{T}}}$

Def 1.3.3. Vega Rotation Smile/Vega Strike. $\text{Shock} = \frac{1}{\sqrt{\text{T}}} \frac{\log \text{K}\%}{\log 0.9}$

- $K\%$ = Strike (Reference Spot).

1.4. Voma

Voma is second order derivative of price w.r.t. volatility.

Def 1.4.1. Voma. $\text{Voma} = \frac{\partial^2 \text{Price}}{\partial^2 \text{Vol}}$

Ex 1.4.1. Baker Pricing Model Voma Approximation.

$$\text{Voma} = \frac{\text{Prix}(\text{Vol Pricing}^+) - 2\text{Prix}(\text{Vol Pricing}) + \text{Prix}(\text{Vol Pricing}^-)}{(\text{Vol Pricing}^+ - \text{Vol Pricing})^2}$$

- Vol Pricing$^+$ = 1.01Vol Pricing
- Vol Pricing$^-$ = 0.99Vol Pricing

1.5. Sticky Strike Greeks

When model of pricing is ST or STSK, Sticky Strike Greeks are computed by repricing, where calibration of the local volatility is calculated.

Def 1.5.1. Delta Sticky Price. $\Delta = \frac{\text{Price}(S_t + S, \text{LocalVol}(S_t + S)) - \text{Price}(S_t, \text{LocalVol}(S_t))}{\Delta S}$

There's no repricing in ST model for standard Greeks, to compute standard Delta, we use points of the PDE mesh for which the price is calculated. Sticky Strike Greeks are in general longer to compute since they imply a repricing and local vol recalibration, but more accurate.

1.6. Gamma

Def 1.6.1 Gamma in VarSwap. Gamma of the Log Swap, which is a financial payoff to compute Variance Swap volatility, it corresponds to the gamme of replication of variance swap.

Def 1.6.2 Gamma 2 in VarSwap. Second derivative of Variance Swap price w.r.t spot.

Def 1.6.3 Cross Gamma. $\Gamma_{12} = \frac{P_{12}^+ - P_1^+ - P_2^+ + P}{\epsilon^2 S_1 S_2}$

- $\epsilon = 1\%$
- P_1^-, P, P_1^+ is parralel to x-axis (S_1)
- P_2^-, P, P_2^+ is parralel to y-axis (S_2)
- $P_{12}+ = P_1^+ - P + P_2^+$

Def 1.6.3 Gamma corrected. $\Gamma_1^{corr} = \frac{P_{12}^+ + P_{12}^- - 2P}{4\epsilon^2 S_1^2}$

- $\epsilon = 5\%$
- width $= 2\epsilon$, parralel to x-axis (S_1)
- height $= \epsilon \rho_{12} \frac{\sigma_2}{\sigma_1}$, parralel to y-axis (S_2)
- $P_{12}^- = P - \text{width} - \text{height}$
- $P_1^+ = P + \text{width} + \text{height}$

Def 1.6.4 Gamma projected. $\Gamma_1^{proj} = \frac{P_{12}^+ + P_{12}^- - 2P}{\epsilon^2 S_1}$

- $\epsilon = 10\%$
- width $= \epsilon\sigma_1$, parralel to x-axis (S_1)
- height $= \epsilon\rho_{12}\sigma_2$, parralel to y-axis (S_2)
- $P_{12}^- = P-$ width $-$ height
- $P_1^+ = P +$ width $+$ height

Def 1.6.5 Translation Gamma.

$$\Delta_i^+ = \frac{P_i^{++} - P_i^{+-}}{2\epsilon_\Delta S_i(1 + \epsilon_{TR})}, \Delta_i^- = \frac{P_i^{0+} - P_i^{0-}}{2\epsilon_\Delta S_i}$$

$$\Gamma_i^{trans} = \frac{\Delta_i^+ - \Delta_i^-}{S_i\epsilon_{TR}} X^{P \to i}$$

$$P_i^{+-} = \text{Prix avec S} = \begin{pmatrix} S_1(1 + \epsilon_{TR}) \\ \vdots \\ S_i(1 + \epsilon_{TR})(1 - \epsilon_\Delta) \\ \vdots \\ S_N(1 + \epsilon_{TR}) \end{pmatrix}$$

$$P_i^{0-} = \text{Prix avec S} = \begin{pmatrix} S_1 \\ \vdots \\ S_i(1 - \epsilon_\Delta) \\ \vdots \\ S_N \end{pmatrix}$$

Def 1.6.6. Factorial Gamma. $\Gamma^{GF} = (\Gamma_{i,j}^{GF})_{i,j \in \{1,\dots,n\}^2}$

Given n underlyings S_i, variation of delta w.r.t. S_i is

$$\delta\Delta_i = \sum_{j=1}^n \Gamma_{i,j}\delta S_j$$

- $\Gamma_{i,j} =$ cross-gamma w.r.t. spots S_i and S_j.

Decompose each δS_i on state factors $(\delta F_k)_k$:

$$\delta S_i = \sigma_i(\sum_{k=1}^K \alpha_{i,k}\delta F_k + e_i)$$

Take $K = 3, \sigma_i, \alpha_{i,k}$ as scalar coefficients, e_i error term, then

$$\delta S_i = \sum_{j=1}^n \Gamma_{i,j}^{GF}\delta S_j$$

- $\Gamma_{i,j}^{GF}$ = linear combination of $(\delta F_k)_k$

Resulting matrix $\Gamma^{GF} = (\Gamma_{i,j}^{GF})_{i,j \in \{1,\dots,n\}^2}$ is factorial gamma.

1.7. Delta/Gamma and Vega/Voma NPrices

Use several prices with polynomial regression of second degree, with first and second derivative (Delta/Gamma and Vega/Voma).

Two parameters

- NbPoints = # spots
- Step Size = relative spot shock between each spot

Default setting

- Large: NbPoints = 11 / Step Size = 0.02
- Small: NbPoints = 11 / Step Size = 0.01
- Fine and large: NbPoints = 21 / Step Size = 0.01

2. Financial Markets

2.1. Commodities

2.1.1. Forward/Future markets

Physical nature of underlying make no-arbitrage formula too complex, there's a need of hedging future production/consumption.

- Base contract is future.
    - Price of the contracts for some maturities build up the forward curve.
    - There's no general rule, can be contango, backwardation, seasonality, etc.
    - Interval between quoted maturieis depends on the exchange, and may not be regular.
- Physical delivery of cash settlement
- Forward markets also quote a spot
- Future markets quotes only futures, with notion of nearby and rolling.

Implied parameters

- Initial future/forward curve of prices
- Each future is driftless
    - can be seen as driftless stock, but simulating only until its expiry
- Volatilities of each future
    - ATM terminal vol (for the option corresponding to the future expiry)
    - Terminal smile
    - Time structure of vol for each future (from vol swaption: option maturing in T_1 on a future of expiry T_2)
    - Smile of swaption

- Correlations between futures on a single curve: correl intra
- Correlations between futures of two different curves: correl inter
- Correlations between commodity futures and other assets (stock, forex)

2.1.2. Pricing Model

2.1.2.1. Black Model

- Use moneyness to use ref strike when reading smile, else only ATM vol is used.
- Use σ_t to have a time structure on vol, else vol of each future is const.

$$dF_t^T = F_t^T \sigma(t,T) dW_t^{F^T}$$

- F_t^T = value of future expiry T in $t < T$.
- No drift to ensure expectation of future = spot value of future
- Diffused until T
- $\sigma(t,T)$ = const vol or time-vol
- Reference strike taken into account for smile effect
- Correlations assumed const

2.1.2.2. Two-Factor Model

- Use moneyness to use ref strike when reading smile, else only ATM vol is used.
- Calibrate 2F params from implied vol/swaption/correl
- Use full smile to have a local vol model on 2F

$$dF_t^T = F_t^T (w_1(t,T) dU_t + w_2(t,T) dW_t)$$

- Brownians U_t, W_t shared by all the futures of a curve
- Choosing a good form for w_1, w_2, simulation is easy.
- Futures are log-normal
- Implicitly, each future is simulated, without drift (forward curve expectation is OK)
- $w(t,T)$ controls the time structure of volatilities and correlations, important to calibration.
- Improvement of Monte Carlo performance in case of products on many futures.

2.2. Repo

$$\text{drift}^{\text{ovn}}_{S_A,B} = (1-x)(OVN_B + DUF_A - DUF_B - q) + x(DFE_A - \mathbb{1}_b(DUF_A - BOR_A)) - r - a$$

- XXX = average rate y s.t. $ZC^{XXX}(t) = e^{-yt}$.
- S_A = underlying in currency A with instantaneous repo $q(t)$.
- B = financing currency

- BOR_A = stripping BOR 3M flat, local refinancing, no spread. Technical curve used historically for equity drift (no financial reality), e.g. EUREIB, USDLIB.
- DUF_A = stripping BOR 3M flat, EUR refinancing, no spread, e.g. USD-DUF.
- DFE_A = stripping BOR 3M flat, EUR refinancing, SG treasury spread. Used to discount flows of transactions without CSA.
- $DUC_{C,D}$ = C cash flow currency, D collateral payment currency. If C = D, it's an OVN curve. Used to price cash flows of fully collateralized trades (i.e. under CSA) with symmetric collateralization in cash. Used for forex drift.
- $F_x(A \to B)$ = forex from A to B, i.e. value of 1 unit of A expressed in B.
- $\delta^X_{A \to B}$ = continuous cross-currency (xccy) basis between currencies A and B. It's computed from ZC curves stripped on xccy swaps (usually collateralized in \$).
- $\delta^\phi(t) = DFE_\$(t) - DUF_\$(t)$ = funding spread.
- $\delta^{ois/bor}_A = BOR_A - OVN_A$ = ois/bor basis in ccy A, when OVN is defined in currency A.
- r = underlying retrocession, a rate homogeneous to repo, const and additive.
- x = funding weight. It's the % of rate drift under DFE.
- a_t = underlying drift add-on.
- b = whether funding without xccy, false in most cases. If true, remove xccy from the funding part: this was added for SGI indices which contractually drift at DFE without xccy.

2.2.1. Curves

Let $P_{DUF}(C_1, C_2) = \frac{DUF_{C1,\$}}{DUF_{C2,\$}}$ be pivot ratio, target is to pivot with $P_{DUC}(C_1, C_2) = \frac{DUC_{C1,\$}}{DUC_{C2,\$}}$.

- $DUC_{C1,C2} = DUC_{C2,C2} P(C_1, C_2)$
- $DUF_{C1,C2} = DUF_{C2,C2} P(C_1, C_2)$
- $DUF_\$ = BOR_\$$, BOR curve is flagged as estimation curve (underlying a forward curve), DUF curve is flagged as discount curve, leading to differences in the way they're stripped.
- $DUF_A = BOR_A + \delta^X_{A \to \$}$
- $DUC_{A,A} = OVN_A$
- $DUC_{A,\$} = OVN_A + \delta^X_{A \to \$}$
- $DFE_A = DUF_A + \delta^\phi = BOR_A + \delta^X_{A \to \$} + \delta^\phi = OVN_A + \delta^{ois/bor}_A + \delta^X_{A \to \$} + \delta^\phi$
- $OVN_A - OVN_B = BOR_A - BOR_B - (\delta^{ois/bor}_A - \delta^{ois/bor}_B)$
- $DFE_A - DFE_B = BOR_A - BOR_B + (\delta^X_{A \to \$} - \delta^X_{A \to \$}) = BOR_A - BOR_B + \delta^X_{A \to B}$
- $\text{drift}_{FX_{A \to B}} = DUC_{B,\$} - DUC_{A,\$} = OVN_B - OVN_A + {}^X_{B \to A}$
- $\delta^X_{A \to B} \sim (DUC_{A,\$} - OVN_A) - (DUC_{B,\$} - OVN_B) \sim (DUF_A - BOR_A) -$

$$(DUF_B - BOR_B)$$

2.2.2. Instantaneous Drift

$A = B$, no overnight ref

$$\text{drift}_{S_A}^{no\_ovn} = (1-x)(BOR_A - q) + x(DFE_A - \mathbb{1}_b(DUF_A - BOR_A)) - r - a$$

$A = B$, with overnight ref

$$\text{drift}_{S_A}^{ovn} = (1-x)(OVN_A - q) + x(DFE_A - \mathbb{1}_b(DUF_A - BOR_A)) - r - a = \text{drift}_{S_A}^{no\_ovn} - (1-x)\delta_A^{ois/bor}$$

$A \neq B$, no overnight ref

$$\text{drift}_{S_A}^{no\_ovn} = (1-x)(BOR_B + DUF_A - DUF_B - q) + x(DFE_A - \mathbb{1}_b(DUF_A - BOR_A)) - r - a = \text{drift}_{S_A}^{no\_ovn} - (1-x)\delta$$

$A \neq B$, with overnight ref

$$\text{drift}_{S_A}^{ovn} = (1-x)(OVN_B + DUF_A - DUF_B - q) + x(DFE_A - \mathbb{1}_b(DUF_A - BOR_A)) - r - a = \text{drift}_{S_A,B}^{no\_ovn} - (1-x)\delta_B^{ois/}$$

3. Stochastic Volatility Model

3.1. P1 Volatility Model

$$\frac{dS_t}{S_t} = r dt + \sigma(t, S_t)\sqrt{\epsilon_t^T} dZ_t$$

$$\epsilon_t^T = (1 - \gamma_T)\epsilon_{1,t}^T + \gamma_T \epsilon_{2,t}^T$$

$$\frac{d\epsilon_{1,t}^T}{\epsilon_{1,t}^T} = \omega_T((1-\theta)e^{-k_1(T-t)}dU_t + \theta e^{-k_2(T-t)}dW_t)$$

$$\frac{d\epsilon_{2,t}^T}{\epsilon_{2,t}^T} = \beta_T \omega_T((1-\theta)e^{-k_1(T-t)}dU_t + \theta e^{-k_2(T-t)}dW_t)$$

$$\epsilon_{1,0}^T = \epsilon_{2,0}^T = \epsilon_0^T, \omega_T = 2\nu \frac{\zeta_T}{1 - \gamma_T + \gamma_T \beta_T}$$

$$< dU_t, dW_t >= \rho_{XY} dt, < dZ_t, dU_t >= \rho_{SX} dt, < dZ_t, dW_t >= \rho_{SY} dt$$

Pricing settings

- Activate decoration
  - Add a multiplicative local volatility to the instantaneous vol. This permits to fit the market smile.
- StockStock Correl Calibration
- Calibrate Skew ATMF
- Vol of Vol, Implied Vol
- Input Variance Swap in Market Data (Used only when calibration is on Variance Swaps)
- Calib. steps
  - Number of Monte Carlo path used when calibration is on ATMF vanillas.

3.1.1. Local Volatility

Add local volatility to stochastic volatility term, to better fit vanilla smile and implied vol.

Classic equation (stock without dividends)

$$dS_t = (r_t - q_t)dt + \sqrt{\epsilon_t^t}dW_t$$

Local Vol equation

$$dS_t = (r_t - q_t)dt + \sigma(t, S_t)\sqrt{\epsilon_t^t}dW_t$$

3.1.2. Forward Variance Case

Dynamic for forward VS variance instead of instant variance (MIV). Forward variance is driven by 2 factors and is auto-calibrated on the VS curve.

$$\epsilon_t^T = \epsilon_0^T f^T(t, X_t, Y_t)$$

- ϵ_t^T = T-forward instant variance at t, derived from vol surface
- f^T = mapping function
- $(k_1, k_2, \theta, \sigma)$ = Market data inputs

Properties

- Auto-calibration on vol surface
- 2 factors give a control on the vol of vol term structure as well as correlation between forward volatility.
- It gives a control on the smile of vol of vol.
- Options on vol surfaces are priced consistently in the same manner as options on standard underlyings.

Correlations for multi-underlying pricing

- Intra correlation: between underlying and its two factors.
- Correlation between different underlying.
- Correlation between underlying i and vol $j = (X_j, Y_j)$.
- Correlation between vol i and vol j.

3.2. Calibration

3.2.1. Calibration on Stock/Stock collreations

To fit the worst-of call option (strike date at pricing date) by iterative Monte Carlo calibration.

- MCSteps = 10000 to choose number of path used to compute each price.
- Strike of the call
- Max number of iterations. Calibration stops once one reaches 1bps precision or max iterations.

3.2.2. Calibration on Variance Swaps

Initial forward variances are calibrated on Variance Swaps taken from implied vol surface through the replication formula. Cash vol shift is added to vol VarSwaps.

3.2.3. Calibration on ATMF puts

This calibration fits the implied vol surface at point $(T, K = \alpha * F_0^T)$, where F_0^T = forward with maturity T.

Difference between calibration on put options and call options

- Calibration on calls is slower
- Vega of the forward is very low, but not null when there're dividends. (slight difference)

3.2.4. Calibration on ATMF calls

P1 model uses a set a parameters (smile, θ, volvol, long mean rev, short mean rev).

Variance swap curve is a variable of the model.

Analogy

- varaince - rate
- P1 - HJM: variance curve plays around bond curve
- M4 - instantaneous rate model: $\text{var}_t \sim r_t$.

So the variance swap curve must be stripped to fit either the varswaps curve or prices curve of ATMF calls. If true, calib on calls, if false, variance swaps.

Def. Calibration on ATMF calls. Given $T_1 < ... < T_N$ and call prices $C(T_i, K_i)$ where K_i are forwards at time T_i, find the variance curve so that the P1 model fits the call prices.

3.2.4.1. Monte Carlo algorithm, exact and iterative We sample M states of the universe $\omega_k \in \Omega$ and calibrate the model s.t.

$$\frac{1}{M}\sum_{k=1}^{M}(S_{T_i}^{Mod}(\omega_k) - K_i)^+ = C^{Mkt}(T_i, K_i)$$

- Exact
  - $=$ is strict equality on the paths ω_k considered.
- Iterative
  - Variance curve is piecewise constant on $[T_{i-1}, T_i]$
  - For $t \in [T_0, T_1]$, variance is constant and chosen as (1) with $i = 1$ be satisfied.
  - For $t \in [T_{N-1}, T_N]$, variance is constant and chosen as (1) with $i = N$ be satisfied.

3.2.4.2. Vegas Settings No dates, no weights, $K_i = F_t^{T_i}$.

- The calibration is exact (vs. Best-fit) hence no weights.
- The maturities T_i are not inputed directly.
  - Take the maturity of vol matrix of the instrument.
  - Add extra maturities if needed to have less than 30 dividends in each interval $[T_i, T_{i+1}]$.

2 Calibration parameters in Vega settings.

- Calibration steps
  - Number M of Monte Carlo paths used for calibration.
- Time step
  - dt used in Euler scheme in the calibration discretization.
  - NB: step is necessarily 1 day for pricing, take $dt = 1$Day.

3.3. Usage

3.3.1. Calendar Effect vol VS

VS volatilities are as calendar effect volatilities, calendar effect has weights for weekends and closed days. When calibration is on ATMF vanillas, the calibrated vols VS are logged in Vegas as calendar effect volatilities.

3.3.2. Weights for Correlation Matrix Projection

Projection algorithm is a minimization routine on the weighted distance. Starting point (first guess) is the result of Higham algorithm, where all correlations with a weight strictly higher than 0.5 are fixed.

Three matrices are logged in vegas:

- input matrix
- weight matrix
- projected matrix

3.3.3. Init Variance FWD(Und, t_1, Δt)

$$\sqrt{\frac{1}{\Delta t} \int_{t_1}^{t_1+\Delta t} \epsilon_t^u du}$$

To get VIX, set Und = S&P 500, $t_1 = t$, $\Delta t = 1M$.

3.3.4. Future on Vol Index

Define underlying und as a future on a vol index (e.g. VIX)

- und: underlying to mark as a future on a vol index.
- volIndex: stock or index owning the vol (S&P for VIX index)
- maturity of future
- offset: tenor of vol index (30 days for VIX index).

$$\mathbb{E}[\sqrt{\frac{1}{\text{offset}} \int_{\text{maturity}}^{\text{maturity+offset}} \frac{u}{\text{maturity}} du | \mathcal{F}_{\text{date}}]$$

- $\epsilon_t^T = \mathbb{E}[\sigma_T^2 | \mathcal{F}_t] = $ forward instantanuous variance

If it's American products on Vol Index, simply set future maturity always = fixing date.

$$\mathbb{E}[\sqrt{\frac{1}{\text{offset}} \int_{\text{date}}^{\text{date+offset}} \frac{u}{\text{date}} du | \mathcal{F}_{\text{date}}]$$

3.3.5. Greeks

Sticky Dupire delta is delta computed from a shocked price with no re-calibration.

3.4. Smile Checker

It's post-calibration smile and vanilla prices dumper. It dumps

- Model implied volatility
- Error between market and model implied volatility
- Vanilla prices with all equity models and forex.

Pricing Methodologies

- PDE
  - Stepsize depends on maturity, 0.088 for $T \leq 1 <$, 1.0 for $1M \leq T \leq 2Y$, 5.0 for $T \geq 2Y$.
- Monte Carlo
  - Generator: Mersenne-Twister with Ninomiya pillars.
  - Split strike-wise: put $K < S_0$, call $K > S_0$.
  - Number of steps: 10000 by default.

Model smile is computed by PDE/MC CSA prices + Baker forward CSA bond $\rightarrow$ Black-Scholes with Newton-Brent.

- Market call prices computed with Baker pricer.
- Implied vol errors computed in this framework, don't expect to fit MSD implied volatilities.

3.5. Hybrid Smile Calibrator: Decorated P1 Stochastic Volatility Model

$$df_t = f_t \sigma(t, f_t) a_t dW_t$$

This model is calibrated to market smiles iff effective volatility = Dupire local volatility.

- $\sigma(t, f_t)^2 \mathbb{E}^{P_{f_t}}[a_t^2 | f_t = f] \equiv \sigma_{\text{Dupire}}(t, f_t)^2$
- $df_t = f_t \frac{\sigma_{\text{Dupire}}(t, f_t)}{\sqrt{\mathbb{E}^{P_{f_t}}[a_t^2 | f_t = f]}} a_t dW_t$

Nonlinear McKean equation

$$dX_t = b(t, X_t, P_{t,X})dt + \sigma(t, X_t, P_{X_t})dW_t$$

- Nonlinear because volatility and drift depend not only on t, X_t, but also on distribution of X_t.
- Can be simulated replacing this density by the empirical distribution given by N simulation of interacting process, which becomes particle algorithm.
  - $df_t = f_t \sigma(t, f_t) a_t dW_t = f_t \sigma_{t_k}(f_t) a_t dW_t$
  - $X_t^{(i)} = (f_t^{(i)}, a_t^{(i)})$
  - $\sigma_{t_k} = \frac{\sigma_{\text{Dupire}}(t_k, f)}{\mathbb{E}^{P_{f_t}}[a_t^2 | f_t = f]}$
  - $\mathbb{E}^{P_{f_t}}[a_t^2 | f_t = f] = \frac{\sum (a_t^{(i)})^2 \delta(f_t^{(i)} - f))}{\sum \delta(f_t^{(i)} - f)}$

4. Longstaff Schwartz

4.1. Longstaff Schwartz Monte Carlo Regression Functions

We can construct scatter plot representing 10000 realizations of future cash flows of an Euro call (if 2 underlyings then 2D) that mature 6 months later, 0 rate 0 dividends 0 repos, const vol = 30%.

- Const regression
  - Regression on averaging all scatter plot (all backwarded future cash flows).
- Polynomial regression
  - Multi-polynomial function of N regressors by least square, each monom of function is a line of matrix of regressors.
- Moving Average 1D (Gaussian Kernel)
  - Regression function is an interpolator on `nbPointInterp` points (X, Y)
    * $Y = \dfrac{\sum_i e^{-\left(\frac{X - x_i}{aFactor \times std(x)}\right)^2} y_i}{\sum_i e^{-\left(\frac{X - x_i}{aFactor \times std(x)}\right)^2}}$
    * $(x_i, y_i) =$ points of scatter plot
    * $x_{\max} - x_{\min} = normFactor$
- Moving Average 2D (Gaussian Kernel)
  - 2D interpolation on 2 regressors `nbPointInterp[1]` x `nbPointInterp[2]` points.

4.2. Application on Clause/Chooser

Callable clause: exit after exercise

$$V_{t_0} = \inf_{\tau \in T} \mathbb{E}^Q_{t_0}[\sum_{t_i < \tau} DF_{t_0, t_i} CF^{noEX}_{t_i} + DF_{t_0, \tau} CF^{EX}_{\tau}]$$

Callable chooser: one/two exercises and continue. Example

$$V_{t_0} = \inf_{\tau \in T} \mathbb{E}^Q_{t_0}[\sum_{t_i < \tau} DF_{t_0, t_i} CF^{noEX}_{t_i} + \sum_{t_i > \tau} DF_{t_0, t_1} CF^{EX}_{t_i}] \quad (\text{putable} \to \sup_{\tau \in T})$$

Choice strategy based on Longstaff Schwartz.

4.2.1. Payoff Decision Tree

- One decision date = one node = one decision + 2 (or more) decisions.
- One direction = one branch from parent decision to the next decision node
- Execution and calibration are automatically fitted to the payoff's geometry.

4.2.2. Node's Longstaff Schwartz calibration

- First build the cloud of points (coupons value) by computing two nodes generated in $t + 1$: $B^t_{CF} = B^{t+1}_{CF} + CF^{t+1}, {B'}^t_{CF} = {B'}^{t+1}_{CF} + CF^{t+1}$
- Then solve the regression for $B^t_{CF}, {B'}^t_{CF}$

- Finally compute the choice for each path $B_{CF}^t = [B, B', B, B, B', ...]$, choice = max (expected value if continuation branch 1, expected value if continuation branch 1).
- For one chooser node at t_i, we choose between 2 directions for t_{i+1}, define a default one and an exercise one, we can exercise several times, but the dual algorithm to computer other bound only works for one exercise.
- The better the regressors, the smaller the interval, the faster the convergence of the other bound.
- Can price margining calls, where a function would return expected value, taking subpayoffs as an argument and using Longstaff Schwartz regression factors, result used in main payoff.

5. Steepner Closed Formula: Price Vanilla Options on a Spread of CMS.

5.1. Displaced log-normal Spread

Compute the price of rate spread options on $S_T = R_T^1 - R_T^2$, where the spread is a reverse displaced log normal diffusion:

$$S_T = d - (d - S_{t,T})e^{\sigma\sqrt{T-t}Z - \frac{\sigma^2}{2}(T-t)} \quad Z \sim \mathcal{N}(0,1)$$

- $S_{t,T}$ = spread convexified forward, computed by static replication.
- d = displacement (shift) which produces a negative Gaussian smile slope.

Price of call on spread maturing at T with strike K in this displaced model is Black-Scholes price s.t.

$$Call_{LND}(t,T,K) = B(t,T)\{(d-K)\mathcal{N}(d_1) - (d - S_{t,T}\mathcal{N}(d_2))\}$$

- $d_1 = \frac{1}{2}\sigma + \ln(\frac{d-K}{d-S_{t,T}})$
- $d_2 = d_1 - \sigma$

Value of log-shifted volatility is obtained by equalizing the price in this log-normal displaced model with the price in a normal model (where swap rate $\sim$ normal diffusion) with assumptions that the strike = 0, convexified spread = 0.

This Gaussian price is obtained by using volatilities of the two CMS, calculated by implicitating the Gaussian Swaption prices at the money forward.

$$Call_N(t,T,K) = B(t,T)\{(S_{t,T} - K)\mathcal{N}(d) + \sigma_{spread}\sqrt{T-t}\frac{e^{-\frac{d^2}{2}}}{\sqrt{2\pi}}\}$$

- $d = \frac{S_{t,T}-K}{\sigma_{spread}\sqrt{T-t}}$
- $\sigma_{spread} = \sqrt{\sigma_1^2 + \sigma_2^2 - 2\rho\sigma_1\sigma_2}$

To use closed formula in Vegas, inputs are ccy, tenor1 (e.g. 2023), tenor 2, mean reversion, correlation between 2 underlyings constituting the spread.

5.2. Stochastic Volatility Spread

$$S_T = d - (d - S_{t,T})H_T, \quad d\nu_t = \lambda(1 - \nu_t)dt + \gamma\sqrt{\nu_t}dW_t^2$$

$$\frac{dH_t}{H_t} = \sigma\sqrt{\nu_t}dW_t^1, \quad < dW_t^1, dW_t^2 >= \rho dt$$

- Spread Vol Calib Strike = 0 by default
- Spread Vol Calib Method
  - Calibration method of σ at strike K given by Spread Vol Calib Strike.
    * Standard: fits call price in a gauss model. Rates are centered on convexified forwards, they have gauss ATM vol from swaption matrix, and they're correlated with input Correl.
    * Frozen: fits call price when spread convexified forward is assumed = 0. When moves, remains the same. It allows to control smile dynamic when computing delta.
- Spread Vol Shift = shift applied to the gauss vol before calibration of σ is set, whatever the maturity is, to 0.
- Mean Reversion of Heston process is used to model stochastic volatility.
- Spread Vol Correl = correlation between stochastic volatility and spread.

6. Gap: Discontinuous Price Smoothing

Def 2.1. Gap. Gap is a hypotenuse of triangle, where horizontal edge is gap size, vertical edge is the price jump, where jump = ExitFees - h_{T_i}.

Use case

- Security cushion/hedging provision.
- Regularization tool for all risk indicators for
  - Smoothing the greeks
  - Capping the delta exposure in a conservative way, s.t. $\Delta \leq \Delta_{\max}$, gapSize $\sim \frac{\text{Jump}}{\Delta_{\max}}$.

6.1. Gap in Probability, the best way to apply Gap

6.1.1. Old WithGap method, directly use Estimated Jump

Successive knock-outs: need for LSMC nested gapping.

$$V_t = \mathbb{E}[\mathbb{1}_{S_{T_i} > L}\{\text{ExitFees}(T_1) - h_T\} + h_T | \mathcal{F}_t]$$

$$\text{gap}(S_{T_i}, L, \hat{h}_{T_i}) = \text{PositiveJumpSign} \times (\text{ExitFees}(T_i) - \hat{h}_{T_i})_+ + \text{NegativeJumpSign} \times (\hat{h}_{T_i} - \text{ExitFees}(T_i))_+$$

- $h_T = \mathbb{1}_{S_{T_2} > L} \{ \text{ExitFees}(T_2) - H_T \} + H_T$
- $T_0 < T_1 < T_2 < T$

6.1.2. Gap in Probability is applied Path-by-Path

If positive jump, we exercise more; if negative jump, we exercise less.

$$\overline{V_t} = \mathbb{E}[\theta(S_{T_i}, L, \hat{h}_{T_i})\{\text{ExitFees}(T_1) - h_T\} + h_T | \mathcal{F}_t]$$

Advantages

- Pathwise continuity implies smoother greeks
- Built-in gap
- Regression is only required to determine the jump sign, hence much less sensitive to Longstaff-Schwartz noise.
  - No more instabilities
  - Better convergence speed for both price and Greeks
  - Optimal gap automatically calculated for products which used over-conservative jump proxy

6.2. Optimal Time to Apply Gap

6.2.1. Autocall Glider

Def. Glider. Knock-out barrier throughout a time interval, $S_{T_i} = \min_{s \le u \le T_i} \{\text{Worst}_u\}$.

$$\text{ExitProb} = \mathbb{1}_{\min_{s \le u \le T_i}\{\text{Worst}_u\} > L} = \mathbb{1}_{\text{AllDatesMustHit} > L} = \mathbb{1} - \mathbb{1}_{\text{AtLeastOneDateMustHit} < L}$$

Trader can gap the continuation probability at $\tau := \min\{u \in [s, T_i], S_u < L\} =$ the first time when underlying goes below limit. An over-conservative gap was chosen because:

- h_τ is not accessible (stochastic stopping time) only h_{T_i} is computable.
- Mistaken belief that gapping at τ or gapping daily is the right way.
  - Hence gap over-conservatively at T_i with a "double gap".

Gap should be applied at **effective discontinuity event date** T_i.

- Optimal gap cost if we gap in probability (computes optimal jump automatically)
  - Exit probability of Autocall glider $= \mathbb{1}_{\min_{s \le u \le T_i}\{\text{Worst}_u\} > L} = \min\{\mathbb{1}_{\text{Worst}_s > L}, ..., \mathbb{1}_{\text{Worst}_u > L}, ..., \mathbb{1}_{\text{Worst}_{T_i} > L}\}$

- Any number of periods is possible: optimal nested gapping (e.g. 2 Glider periods)
- Any look-back path-depth observations is possibly (not only min).

6.2.2. Forward Volatility Agreement (FVA) with daily knock-out

Daily knock-out is difficult to manage.

- Performance problems
- Accumulation of nested regression errors
- Daily hedging is painful

Use artificial look-back underlying to avoid daily knock-out.

$$V_t = \mathbb{E}[\mathbb{1}_{S_{T_0}<L} \times ... \times \mathbb{1}_{S_u<L} \times ... \times \mathbb{1}_{S_{T_1}<L}\}h_T|\mathcal{F}_t] = \mathbb{E}[\mathbb{1}_{\max_{T_0 \leq u \leq T_1}\{S_u\}<L}h_T|\mathcal{F}_t]$$

- $S_u > L, \in [T_0, T_1]$, exit with 0.
- $h_T \in \{T_1, T_2\}$, exit with CF_{T_1}, CF_{T_2}.

6.2.3. Gap an Indicate with look-back Underlying

A bit difference from Exit probability of Autocall glider.

$$\text{gapped}\{\mathbb{1}_{\min_{s \leq u \leq T_i}\{\text{Worst}_u\}>L}\} = \min\{\mathbf{Slope}_{\text{Worst}_s>L}, ..., \mathbf{Slope}_{\text{Worst}_u>L}, ..., \mathbf{Slope}_{\text{Worst}_{T_i}>L}\}$$

- $\min_{OnDates}$, all dates must hit condition
- $\max_{OnDates} = \mathbf{Slope}$, at least one date must hit condition.

6.3. How to Gap classical Autocall Worst Underlying

6.3.1. Classical Autocall Trigger Clause

Indicate to gap: $\mathbb{1}_{\text{Worst}_{T_i}>L} = \mathbb{1}_{\min\{S_{T_i}^{(1)},...,S_{T_i}^{(n)}\}>L}$

But we only have inputs $\text{gapSize}^{(i)}$ for underlying $S^{(i)}$. A naive but over-conservative gapping is to use:

- A slope on the worst underlying
- $\text{gapSize} = \max\{\text{gapSize}^{(1)}, ..., \text{gapSize}^{(n)}\}$

6.3.2. Old method: Worst Inversion

Complex with side effects and possible hidden discontinuities.

- Use $\text{gapSize}^{(\text{WorstIndex})}$ of the worst
- Make a smooth transition if other underlyings get close to the Worst. $\text{gapSize} = \max\{\text{localGapSize}^{(1)}, ..., \text{localGapSize}^{(n)}\}$

$$- \ \text{localGapSize}^{(k)} = (1-_k)\text{gapSize}^{(\text{WorstIndex})} +_k \max\{\text{gapSize}^{(k)}, \text{gapSize}^{(\text{WorstIndex})}\}$$
$$- \ \lambda_k = \textbf{Slope}^{\text{up}}_{\text{S}^{(k)}>\text{Worst}}(\max\{\text{gapSize}^{(k)}, \text{gapSize}^{(\text{WorstIndex})}\}))$$

6.3.3. New method: Min Slope

More robust and optimal, obviously continuous, still conservative, simpler smarter and cheaper. We gap not only depending on worst position, but if all underlying are in their danger zone.

$$\mathbb{1}_{\text{Worst}_{\text{T}_\text{i}}>\text{L}} = \mathbb{1}_{\min\{S^{(1)}_{T_i},...,S^{(n)}_{T_i}\}>L} = \min\{\mathbb{1}_{S_{T_i}^{(1)}>L}, ..., \mathbb{1}_{S_{T_i}^{(n)}>L}\}$$

$$\text{gapped}\{\mathbb{1}_{\text{Worst}_{\text{T}_\text{i}}>\text{L}}\} = \min\{\textbf{Slope}_{S^{(1)}_{T_i}>L}, ..., \textbf{Slope}_{S^{(n)}_{T_i}>L}\}$$

$$\text{gapSize}^{(\text{k})} = \max\{\text{Floor}^{(\text{k})},$$

$$\min\{Cap^{(k)}, \frac{100}{2} \frac{QuantoFactor^{(k)}}{AsianDivisor} \frac{GapRef^{(k)}}{\min(1.2, Limit^{(k)})} \frac{\sqrt{Nominal \times DigitSize^{(k)} \times Jump}}{DeltaMax^{(k)}}\}\}$$

For a slope, effective gapSize is $\text{effectiveGapSize}^{(\text{k})} = 2 \times \text{Limit}^{(\text{k})} \times \text{gapSize}^{(\text{k})}$.

- $2 = $ Slope to be homogeneous to a Rect
- $\times$Limit, because gapSize is homogeneous to a 100% limit.

6.3.4. How to gap other features like in the OmniCall

- BestOf: $\mathbb{1}_{\text{Best}_{\text{T}_\text{i}}>\text{L}}$
  - Same idea, to gap use $\max\{\textbf{Slope}_{S^{(1)}_{T_i}>L}, ..., \textbf{Slope}_{S^{(n)}_{T_i}>L}\}$
- Basket: $\mathbb{1}_{\sum_k \omega_k S^{(k)}_{T_i}>L}$
  - Use slope on the basket underlying, $\text{gapSize} = \sum_k \omega_k \text{ gapSize}^{(k)}$
- Multiperf: $\mathbb{1}_{1-\sum_k\{\text{yield}^{(\text{k})}_{\text{T}_\text{i}}-1\}>\text{L}}$
  - Use slope on the multiperf underlying, $\text{gapSize} = \sum_k \text{gapSize}^{(k)}$

6.4. Regression

6.4.1. Autocall Regression

- Autocall is mono
  - Regressors: {Spot}
  - Method: Moving average
- Autocall is multi
  - Regressors: {Worst, Worst2}

– Method: Linear regression

Using Gap in Probability, mono/multi cases can share the same regression.

- Regressors: {Worst}
  – less regressors, more robust
  – Worst is main explanatory factor
- Method: Moving average
  – more flexible and adaptive
  – No side effects because of restrict domain

6.4.2. FVA Regression

Current regressor = {Spot} is not ideal, should be {Realized Fwd Vol} because FVA is vol product (not possible). So we see **oscillations** like noise because **Spot** is:

- Not main explanatory factor which is **Realized Fwd Vol**
- Orthogonal to **Realized Fwd Vol**

6.5. Application on Magnet Autocall

6.5.1. Magnet

Def. Magnet. If big decrease, increase recall probability in exchange for lower coupons.

When at least one of the underlyings go below L_{magnet} on any date of magnet schedule, i.e., $\mathbb{1}_{\min_{s \leq u \leq T_1}\{Worst_u\} < L_{magnet}}$. Then for all future standard autocall recall dates:

- Trigger barrier is lowered: $L_{New} \leq L_{Old}$
- Bonus coupon is reduced: $ExitFees_{New} \leq ExitFees_{Old}$

Business Interest

- Increase Autocall recycle workflow (make client exit and reinvest in other products)
- Reduce the long vega exposition

6.5.2. Trader gapping mistake

Mistaken belief that gapping by only **increasing the magnet probability** is conservative.

- Increase probability of receiving recall coupons
- But also decrease probability of receiving higher bonus coupons

Conservative only if $ExitFees_{New} = ExitFees_{Old}$. Indeed better recall sooner than later, due to discounting considerations.

6.5.3. How to gap? 2D Discontinuity only feasible with Gap in Prob.

At given recall date T_i,

$$\text{Payoff} = \mathbb{1}_{\min_{s \leq u \leq T_i}\{\text{Worst}_u\} < L_{\text{magnet}}} \{\mathbb{1}_{\text{Worst}_{T_i} < L_{\text{New}}} h_T + \mathbb{1}_{\text{Worst}_{T_i} > L_{\text{New}}} \text{ExitFees}_{\text{New}}\} +$$

$$\mathbb{1}_{\min_{s \leq u \leq T_i}\{\text{Worst}_u\} > L_{\text{magnet}}} \{\mathbb{1}_{\text{Worst}_{T_i} < L_{\text{Old}}} h_T + \mathbb{1}_{\text{Worst}_{T_i} > L_{\text{Old}}} \text{ExitFees}_{\text{Old}}\}.$$

Define exit probability

$$\mathbb{1}_{\text{Trigger}} = \mathbb{1}_{\min_{s \leq u \leq T_i}\{\text{Worst}_u\} < L_{\text{magnet}}} \mathbb{1}_{\text{Worst}_{T_i} > L_{\text{New}}} + \mathbb{1}_{\min_{s \leq u \leq T_i}\{\text{Worst}_u\} > L_{\text{magnet}}} \mathbb{1}_{\text{Worst}_{T_i} > L_{\text{Old}}}$$

Rewrite Payoff

$$\text{Payoff} = \mathbb{1}_{\overline{\text{Trigger}}} h_T + \mathbb{1}_{\text{Trigger}} \{\mathbb{1}_{\min_{s \leq u \leq T_i}\{\text{Worst}_u\} < L_{\text{magnet}}} \text{ExitFees}_{\text{New}} + \mathbb{1}_{\min_{s \leq u \leq T_i}\{\text{Worst}_u\} > L_{\text{magnet}}} \text{ExitFees}_{\text{Old}}\}$$

$$\text{ExitFees} = \mathbb{1}_{\min_{s \leq u \leq T_i}\{\text{Worst}_u\} < L_{\text{magnet}}} \{\text{ExitFees}_{\text{New}} - \text{ExitFees}_{\text{Old}}\} + \text{ExitFees}_{\text{Old}}$$

is gapped to guarantee conservative price:

$$\text{ExitFeesGapped} = \mathbf{SlopeAm}_{\text{Left}}^{\text{Down}} \times \{\text{ExitFees}_{\text{New}} - \text{ExitFees}_{\text{Old}}\} + \text{ExitFees}_{\text{Old}}$$

6.5.4. Autocall Magnet Regression

- Minmal regressors needed: $\{\mathbb{1}_{\min_{s \leq u \leq T_i}\{\text{Worst}_u\} < L_{\text{magnet}}}, \text{Worst}\}$
  - If magnet is triggered or not, it's like having 2 different Autocalls.
- Method: Moving Average / Linear regression + restrict domain.

6.5.5. Summary

Advantages

- Conservative
- Path-by-path continuous
- DeltaMax condition verified
- Extension easily feasible by Gap in Prob method

Improvements

- Have a real "Trigger Clause of Trigger Clauses"
- Use Worst_{T_i} and $\min_{s \leq u \leq t}\{\text{Worst}_u\}$ have correlated laws, to have cheaper but still conservative gap.

6.6. Tempo Call: Autochoose product priced with Longstaff Schwartz + Gap

Def. Tempo Call. Call on the accumulated performance of an underlying on some periods. At each period, performance will be accumulated under some conditions: if we had accumulated for last period, then we will accumulate for the next period in case the yiedl is above a low barrier (95% of high barrier). If we hadn't accumulated for last period, then we'll accumulate for next period in case the yield is above high barrier (100% of last max yield).

This triggerable accumulation generates a jump at accumulation dates, this jump is positive if there're no dividend and negative in case dividend falls during accumulation period. These jumps can be smoothed with Longstaff Schwartz algorithm.

6.6.1. Discontinuity of Tempo Call

- Pricing date at 10 minutes/a day at 23:59pm before a reexposure date
- 10% discret dividend is payed between next reexposure date and reexposure date after (and no div after).

6.6.2. Gap Application

`Gap(SmoothingType, LeftPayoff, RightPayoff, Value, Limit, GapSize, date, Und)` computes an extra payoff > 0 to smooth original payoff conservatively:

- gap = long gap if LeftPayoff < RightPayoff and Value <= Limit
- gap = short gap if RightPayoff < LeftPayoff and Value > Limit
- gap = 0 else, or if Value is outside limit $\pm$ gapSize.

Smoothing Type: long gap computed as up & out call, short gap computed as down & out put.

In Tempo call example, RightPayoff = 1 (Expo), LeftPayoff = 2 (No Expo).