Data: volatility curve obtained by linear interpolation of variance

Inputs:

$(t_m)_{m=1,\dots,M}$: maturities

$(\sigma_m)_{m=1,\dots,M}$: volatilities

t_0 : initial time

Output: volatility on $[t_0, t_M]$ obtained by linear interpolation of variance curve.

Domain: $[t_0, t_M]$

Algorithm:

Step 1 : arguments & values

$$\bar{x} = (t_0, t_1, \dots, t_M)$$

$$\bar{y} = (0, y_1, \dots, y_M)$$

$$y_m = (t_m - t_0)\,\sigma_m^2 \,, \quad m = 1, \dots, M$$

<u>Step 2</u> : linear interpolation

$$(\bar{x}, \bar{y}) \implies V(t)\,, \quad t \in [t_0, t_M]$$

<u>Step 3</u> :

If $t < t_0 + \overset{\approx 10^{-8}}{\varepsilon}$, then

$$t = t_0 + \varepsilon$$

$$\sigma(t) = \sqrt{V(t)/(t - t_0)}\,, \quad t \in (t_1, t_M]$$

return $\big(\sigma(t)\big)_{t \in [t_0, T]}$ $\blacksquare$

Asset Standard : compound call on american put

Inputs

T : maturity of call

C : strike of call

$(t_m)_{m=1,\dots,M}$: exercise times of put

P : strike of put

t_0 : initial time

Event times : $\{t_0, T, t_1, \dots, t_M\}$

Algorithm :

1. Boundary condition

$$X_{t_M} = 0$$

2. Loop $\underbrace{T}_{\text{end}} \longleftarrow \underbrace{t_M}_{\text{begin}}$

$$X_{t_m} \;(?) \longleftarrow X_{t_{m+1}} \;(\text{known})$$

$X_{t_{m+1}}$: value to continue (no exercises $\leq t_{m+1}$)

$$\underset{(\leq t_m)}{X_{t_{m+1}}} = \max\big(\underset{(\leq t_{m+1})}{X_{t_{m+1}}},\ P - S_{t_{m+1}} \big)$$

$$\underset{(\leq t_m)}{X_{t_m}} = \mathcal{R}_{t_m}\big(\underset{(\leq t_m)}{X_{t_{m+1}}} \big)$$

③ After loop

$$\underset{(\text{call})}{X_T} = \max\big(\underset{(\text{put})}{X_T - C},\ 0 \big)$$

$$X_{t_0} = \mathcal{R}_{t_0}(X_T)$$

return X_{t_0} ■

Interest Rate Standard:

callable capped floater

Inputs:

N : notional

C : cap rate

δt : interval between payments

M : # of payments

δh : spread over LIBOR

t_0 : initial time

Event times: $\{t_0, t_1, \dots, t_{M-1}\}$

$$t_m = t_0 + m\,\delta t \quad , \quad m = 1, \dots, M-1$$

① Boundary condition

$$X_{t_{M-1}} = B(t_{M-1}, t_{M-1} + \delta t) * N + \text{Value Next Coupon}(t_{M-1})$$

below

② loop $t_0 \longleftarrow t_{M-1}$

end $\quad\quad\quad\quad$ begin

X_{t_m} (?) $\longleftarrow X_{t_{m+1}}$ (known)

$X_{t_{m+1}}$: value to continue

$$X_{t_{m+1}} = \min\left(X_{t_{m+1}}, N\right)$$

$$X_{t_m} = \mathcal{R}_{t_m}\left(X_{t_{m+1}}\right)$$

$$X_{t_m} \mathrel{+}= \text{Value Next Coupon}\,(t_m)$$

$$\text{Value Next Coupon}\,(t_m) = N * B(t_m, t_m+\delta t)\,\min\left(\left(L(t_m, t_m+\delta t) + \delta L\right)\delta t,\ c\,\delta t\right) = N *$$

$$\min\left(1 + B(t_m, t_m+\delta t)(\delta L\,\delta t - 1),\ B(t_m, t_m+\delta t)\,c\,\delta t\right)$$

③ After loop

return X_{t_0} ∎

Interest Rate Path Dependent:

index amortizing swap

Inputs:

N: notional

R: fixed rate

δt: period between payments

M: # of payments

bPayFloat: side of swap

$f = f(r)$, $r \in (0,1)$: amortizing function for the notional

q: threshold for the cleanup call

Event times: $\underbrace{t_0, t_1, \ldots, t_{M-1}}_{\text{when next payment is computed}}$

Path-dependent state process:

Y_{t_m}: notional for next payment

(a) Origin :

$$Y_{t_0} = 1$$

(b) Reset indices : $\{1, \dots, M-1\}$

(c) One-step evolution:

$$Y_{t_m} \text{ (known)} \longrightarrow Y_{t_{m+1}} \text{ (?)}$$

$$Y_{t_{m+1}} = Y_{t_m} \, f\left(r_{t_{m+1}} \right)$$

$$r_{t_{m+1}} = \left(\frac{1}{\underbrace{B(t_{m+1}, t_{m+1}+\delta t)}_{\text{discount factor}}} - 1 \right) / \delta t$$

Algorithm :

① Boundary condition

$$X_{t_{M-1}} = \underbrace{\text{Value Next Payment}}_{\text{see below}}\,(t_{M-1})$$

② Loop $t_0 \longleftarrow t_{M-1}$

end begin

X_{t_m} (?) $\longleftarrow$ $X_{t_{m+1}}$ (known)

// $X_{t_{m+1}}$: value to continue (notional is above threshold).

// We multiply on notional at the end.

// We pay fixed and receive float.

$$X_{t_{m+1}} \mathrel{*}= 1_{\{ Y_{t_{m+1}} \geq q \}}$$

$$X_{t_m} = R_{t_m} \left(X_{t_{m+1}} \right)$$

$$X_{t_m} \mathrel{+}= \text{Value Next Payment} \, (t_m)$$

where

$$\text{ValueNextPayment}(t_m) =$$

$$Y_{t_m}\left(1 - B(t_m, t_m+\delta t)\,(1 + R\,\delta t)\right)$$

(3) After loop.

If bPayFloat = false, then

$$X_{t_0} \mathrel{*}= -1$$

$$X_{t_0} \mathrel{*}= N$$

return X_{t_0} $\blacksquare$