NAG Toolbox

nag_specfun_opt_lookback_fls_greeks (s30bb)

1 Purpose

nag_specfun_opt_lookback_fls_greeks (s30bb) computes the price of a floating-strike lookback option together with its sensitivities (Greeks).

2 Syntax

```
[p, delta, gamma, vega, theta, rho, crho, vanna, charm, speed, colour, zomma,
vomma, ifail] = nag_specfun_opt_lookback_fls_greeks(calput, sm, s, t, sigma, r,
q, 'm', m, 'n', n)
```

```
[p, delta, gamma, vega, theta, rho, crho, vanna, charm, speed, colour, zomma,
vomma, ifail] = s30bb(calput, sm, s, t, sigma, r, q, 'm', m, 'n', n)
```

3 Description

nag_specfun_opt_lookback_fls_greeks (s30bb) computes the price of a floating-strike lookback call or put option, together with the Greeks or sensitivities, which are the partial derivatives of the option price with respect to certain of the other input parameters. A call option of this type confers the right to buy the underlying asset at the lowest price, S_{min}, observed during the lifetime of the contract. A put option gives the holder the right to sell the underlying asset at the maximum price, S_{max}, observed during the lifetime of the contract. Thus, at expiry, the payoff for a call option is $S - S_{\mathrm{min}}$, and for a put, $S_{\mathrm{max}} - S$.

For a given minimum value the price of a floating-strike lookback call with underlying asset price, S, and time to expiry, T, is

$$P_{\mathrm{call}} = Se^{-qT}\Phi(a_1) - S_{\mathrm{min}}\,e^{-rT}\Phi(a_2) + Se^{-rT}\,\frac{\sigma^2}{2b}\left[\left(\frac{S}{S_{\mathrm{min}}}\right)^{-2b/\sigma^2}\Phi\left(-a_1+\frac{2b}{\sigma}\sqrt{T}\right) - e^{bT}\Phi(-a_1)\right],$$

where $b = r - q \neq 0$. The volatility, σ, risk-free interest rate, r, and annualised dividend yield, q, are constants.

The corresponding put price is

$$P_{\mathrm{put}} = S_{\mathrm{max}}\,e^{-rT}\Phi(-a_2) - Se^{-qT}\Phi(-a_1) + Se^{-rT}\,\frac{\sigma^2}{2b}\left[-\left(\frac{S}{S_{\mathrm{max}}}\right)^{-2b/\sigma^2}\Phi\left(a_1-\frac{2b}{\sigma}\sqrt{T}\right) + e^{bT}\Phi(a_1)\right].$$

In the above, Φ denotes the cumulative Normal distribution function,

$$\Phi(x) = \int_{-\infty}^{x}\phi(y)dy$$

where ϕ denotes the standard Normal probability density function

$$\phi(y) = \frac{1}{\sqrt{2\pi}}\exp\left(-y^2/2\right)$$

and

$$a_1 = \frac{\ln(S/S_{\mathrm{m}}) + \left(b+\sigma^2/2\right)T}{\sigma\sqrt{T}}$$

$$a_2 = a_1 - \sigma\sqrt{T}$$

where S_m is taken to be the minimum price attained by the underlying asset, S_{min}, for a call and the maximum price, S_{max}, for a put.

The option price $P_{ij} = P(X = X_i, T = T_j)$ is computed for each minimum or maximum observed price in a set $S_{\text{min}}(i)$ or $S_{\text{max}}(i)$, $i = 1, 2, \ldots, m$, and for each expiry time in a set T_j, $j = 1, 2, \ldots, n$.

4 References

Goldman B M, Sosin H B and Gatto M A (1979) Path dependent options: buy at the low, sell at the high *Journal of Finance* **34** 1111–1127

5 Parameters

5.1 Compulsory Input Parameters

1: **calput** – CHARACTER(1)

Determines whether the option is a call or a put.

calput = 'C'
A call; the holder has a right to buy.

calput = 'P'
A put; the holder has a right to sell.

Constraint: **calput** = 'C' or 'P'.

2: **sm**(**m**) – REAL (KIND=nag_wp) array

sm(i) must contain $S_{\text{min}}(i)$, the ith minimum observed price of the underlying asset when **calput** = 'C', or $S_{\text{max}}(i)$, the maximum observed price when **calput** = 'P', for $i = 1, 2, \ldots,$ **m**.

Constraints:

sm(i) $\geq z$ and **sm**(i) $\leq 1/z$, where $z =$ x02am(), the safe range parameter, for $i = 1, 2, \ldots,$ **m**;
if **calput** = 'C', **sm**(i) $\leq S$, for $i = 1, 2, \ldots,$ **m**;
if **calput** = 'P', **sm**(i) $\geq S$, for $i = 1, 2, \ldots,$ **m**.

3: **s** – REAL (KIND=nag_wp)

S, the price of the underlying asset.

Constraint: **s** $\geq z$ and **s** $\leq 1.0/z$, where $z =$ x02am(), the safe range parameter.

4: **t**(**n**) – REAL (KIND=nag_wp) array

t(i) must contain T_i, the ith time, in years, to expiry, for $i = 1, 2, \ldots,$ **n**.

Constraint: **t**(i) $\geq z$, where $z =$ x02am(), the safe range parameter, for $i = 1, 2, \ldots,$ **n**.

5: **sigma** – REAL (KIND=nag_wp)

σ, the volatility of the underlying asset. Note that a rate of 15% should be entered as 0.15.

Constraint: **sigma** > 0.0.

6: **r** – REAL (KIND=nag_wp)

The annual risk-free interest rate, r, continuously compounded. Note that a rate of 5% should be entered as 0.05.

Constraint: **r** ≥ 0.0 and abs(**r** − **q**) $> 10 \times$ eps $\times$ max(abs(**r**), 1), where eps = x02aj(), the ***machine precision***.

7: **q** – REAL (KIND=nag_wp)

The annual continuous yield rate. Note that a rate of 8% should be entered as 0.08.

Constraint: $\mathbf{q} \geq 0.0$ and $\mathrm{abs}(\mathbf{r} - \mathbf{q}) > 10 \times \mathrm{eps} \times \max(\mathrm{abs}(\mathbf{r}), 1)$, where $\mathrm{eps} = \mathrm{x02aj}()$, the ***machine precision***.

5.2 Optional Input Parameters

1: **m** – INTEGER

Default: the dimension of the array **sm**.

The number of minimum or maximum prices to be used.

Constraint: $\mathbf{m} \geq 1$.

2: **n** – INTEGER

Default: the dimension of the array **t**.

The number of times to expiry to be used.

Constraint: $\mathbf{n} \geq 1$.

5.3 Output Parameters

1: $\mathbf{p}(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

$\mathbf{p}(i,j)$ contains P_{ij}, the option price evaluated for the minimum or maximum observed price $S_{\min}(i)$ or $S_{\max}(i)$ at expiry $\mathbf{t}_j$ for $i = 1, 2, \ldots, \mathbf{m}$ and $j = 1, 2, \ldots, \mathbf{n}$.

2: $\mathbf{delta}(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

The leading $\mathbf{m} \times \mathbf{n}$ part of the array **delta** contains the sensitivity, $\frac{\partial P}{\partial S}$, of the option price to change in the price of the underlying asset.

3: $\mathbf{gamma}(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

The leading $\mathbf{m} \times \mathbf{n}$ part of the array **gamma** contains the sensitivity, $\frac{\partial^2 P}{\partial S^2}$, of **delta** to change in the price of the underlying asset.

4: $\mathbf{vega}(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

$\mathbf{vega}(i,j)$, contains the first-order Greek measuring the sensitivity of the option price P_{ij} to change in the volatility of the underlying asset, i.e., $\frac{\partial P_{ij}}{\partial \sigma}$, for $i = 1, 2, \ldots, \mathbf{m}$ and $j = 1, 2, \ldots, \mathbf{n}$.

5: $\mathbf{theta}(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

$\mathbf{theta}(i,j)$, contains the first-order Greek measuring the sensitivity of the option price P_{ij} to change in time, i.e., $-\frac{\partial P_{ij}}{\partial T}$, for $i = 1, 2, \ldots, \mathbf{m}$ and $j = 1, 2, \ldots, \mathbf{n}$, where $b = r - q$.

6: $\mathbf{rho}(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

rho(i,j), contains the first-order Greek measuring the sensitivity of the option price P_{ij} to change in the annual risk-free interest rate, i.e., $-\frac{\partial P_{ij}}{\partial r}$, for $i = 1, 2, \ldots, \mathbf{m}$ and $j = 1, 2, \ldots, \mathbf{n}$.

7: **crho**$(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

crho(i,j), contains the first-order Greek measuring the sensitivity of the option price P_{ij} to change in the annual cost of carry rate, i.e., $-\frac{\partial P_{ij}}{\partial b}$, for $i = 1, 2, \ldots, \mathbf{m}$ and $j = 1, 2, \ldots, \mathbf{n}$, where $b = r - q$.

8: **vanna**$(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

vanna(i,j), contains the second-order Greek measuring the sensitivity of the first-order Greek Δ_{ij} to change in the volatility of the asset price, i.e., $-\frac{\partial \Delta_{ij}}{\partial T} = -\frac{\partial^2 P_{ij}}{\partial S \partial \sigma}$, for $i = 1, 2, \ldots, \mathbf{m}$ and $j = 1, 2, \ldots, \mathbf{n}$.

9: **charm**$(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

charm(i,j), contains the second-order Greek measuring the sensitivity of the first-order Greek Δ_{ij} to change in the time, i.e., $-\frac{\partial \Delta_{ij}}{\partial T} = -\frac{\partial^2 P_{ij}}{\partial S \partial T}$, for $i = 1, 2, \ldots, \mathbf{m}$ and $j = 1, 2, \ldots, \mathbf{n}$.

10: **speed**$(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

speed(i,j), contains the third-order Greek measuring the sensitivity of the second-order Greek Γ_{ij} to change in the price of the underlying asset, i.e., $-\frac{\partial \Gamma_{ij}}{\partial S} = -\frac{\partial^3 P_{ij}}{\partial S^3}$, for $i = 1, 2, \ldots, \mathbf{m}$ and $j = 1, 2, \ldots, \mathbf{n}$.

11: **colour**$(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

colour(i,j), contains the third-order Greek measuring the sensitivity of the second-order Greek Γ_{ij} to change in the time, i.e., $-\frac{\partial \Gamma_{ij}}{\partial T} = -\frac{\partial^3 P_{ij}}{\partial S \partial T}$, for $i = 1, 2, \ldots, \mathbf{m}$ and $j = 1, 2, \ldots, \mathbf{n}$.

12: **zomma**$(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

zomma(i,j), contains the third-order Greek measuring the sensitivity of the second-order Greek Γ_{ij} to change in the volatility of the underlying asset, i.e., $-\frac{\partial \Gamma_{ij}}{\partial \sigma} = -\frac{\partial^3 P_{ij}}{\partial S^2 \partial \sigma}$, for $i = 1, 2, \ldots, \mathbf{m}$ and $j = 1, 2, \ldots, \mathbf{n}$.

13: **vomma**$(ldp, \mathbf{n})$ – REAL (KIND=nag_wp) array

$ldp = \mathbf{m}$.

vomma(i,j), contains the second-order Greek measuring the sensitivity of the first-order Greek Δ_{ij} to change in the volatility of the underlying asset, i.e., $-\frac{\partial \Delta_{ij}}{\partial \sigma} = -\frac{\partial^2 P_{ij}}{\partial \sigma^2}$, for $i = 1, 2, \ldots, \mathbf{m}$ and $j = 1, 2, \ldots, \mathbf{n}$.

14: **ifail** – INTEGER

ifail $= 0$ unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

On entry, **calput** $= \langle value \rangle$ was an illegal value.

ifail $= 2$

Constraint: **m** ≥ 1.

ifail $= 3$

Constraint: **n** ≥ 1.

ifail $= 4$

Constraint: $\langle value \rangle \leq$ **sm**$(i) \leq \langle value \rangle$ for all i.

ifail $= 5$

Constraint: **s** $\geq \langle value \rangle$ and **s** $\leq \langle value \rangle$.

ifail $= 6$

Constraint: **t**$(i) \geq \langle value \rangle$ for all i.

ifail $= 7$

Constraint: **sigma** > 0.0.

ifail $= 8$

Constraint: **r** ≥ 0.0.

ifail $= 9$

Constraint: **q** ≥ 0.0.

ifail $= 11$

Constraint: $ldp \geq$ **m**.

ifail $= 12$

Constraint: $|\mathbf{r} - \mathbf{q}| > 10 \times \mathrm{eps} \times \max(|\mathbf{r}|, 1)$, where eps is the ***machine precision***.

ifail $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

ifail $= -399$

Your licence key may have expired or may not have been installed correctly.

ifail $= -999$

Dynamic memory allocation failed.

7 Accuracy

The accuracy of the output is dependent on the accuracy of the cumulative Normal distribution function, Φ. This is evaluated using a rational Chebyshev expansion, chosen so that the maximum relative error in the expansion is of the order of the ***machine precision*** (see nag_specfun_cdf_normal (s15ab) and nag_specfun_erfc_real (s15ad)). An accuracy close to ***machine precision*** can generally be expected.

8 Further Comments

None.

9 Example

This example computes the price of a floating-strike lookback put with a time to expiry of 6 months and a stock price of 87. The maximum price observed so far is 100. The risk-free interest rate is 6% per year and the volatility is 30% per year with an annual dividend return of 4%.

9.1 Program Text

```
function s30bb_example

fprintf('s30bb example results\n\n');

put = 'p';
s = 87;
sigma = 0.3;
r = 0.06;
q = 0.04;
sm = [100.0];
t = [0.5];

[p, delta, gamma, vega, theta, rho, crho, vanna, charm, speed, colour, ...
 zomma, vomma, ifail] = s30bb( ...
                               put, sm , s, t, sigma, r, q);

fprintf('\nFloating-Strike Lookback\n European Put :\n');
fprintf('  Spot       =   %9.4f\n', s);
fprintf('  Volatility =   %9.4f\n', sigma);
fprintf('  Rate       =   %9.4f\n', r);
fprintf('  Dividend   =   %9.4f\n\n', q);

fprintf(' Time to Expiry : %8.4f\n', t(1));
fprintf('%9s%9s%9s%9s%9s%9s%9s%9s\n','S-Max/Min','Price','Delta','Gamma',...
        'Vega','Theta','Rho','CRho');
fprintf('%9.4f%9.4f%9.4f%9.4f%9.4f%9.4f%9.4f%9.4f\n\n', sm(1), p(1,1), ...
        delta(1,1), gamma(1,1), vega(1,1), theta(1,1), rho(1,1), crho(1,1));

fprintf('%27s%9s%9s%9s%9s%9s\n','Vanna','Charm','Speed','Colour',...
        'Zomma','Vomma');
fprintf('%18s%9.4f%9.4f%9.4f%9.4f%9.4f%9.4f\n\n', ' ', vanna(1,1), ...
        charm(1,1), speed(1,1), colour(1,1), zomma(1,1), vomma(1,1));
```

9.2 Program Results

```
s30bb example results

Floating-Strike Lookback
 European Put :
  Spot       =     87.0000
  Volatility =      0.3000
  Rate       =      0.0600
  Dividend   =      0.0400

 Time to Expiry :   0.5000
S-Max/Min    Price    Delta    Gamma     Vega    Theta      Rho     CRho
 100.0000  18.3530  -0.3560   0.0391  45.5353 -11.6139 -32.8139 -23.6374

                      Vanna    Charm    Speed   Colour    Zomma    Vomma
                     1.9141  -0.6199   0.0007   0.0221  -0.0648  76.1292
```