S30BBF

Note: before using this routine, please read the Users’ Note for your implementation to check the interpretation of **bold italicised** terms and other implementation-dependent details.

1 Purpose

S30BBF computes the price of a floating-strike lookback option together with its sensitivities (Greeks).

2 Specification

```fortran
SUBROUTINE S30BBF(CALPUT, M, N, SM, S, T, SIGMA, R, Q, P, LDP, DELTA,
                   1 GAMMA, VEGA, THETA, RHO, CRHO, VANNA, CHAR, SPEED,
                   2 COLOUR, ZOMMA, VOMMA, IFAIL)
INTEGER M, N, LDP, IFAIL
DOUBLE PRECISION SM(M), S, T(N), SIGMA, R, Q, P(LDP,N), DELTA(LDP,N),
1 GAMMA(LDP,N), VEGA(LDP,N), THETA(LDP,N), RHO(LDP,N),
2 CRHO(LDP,N), VANNA(LDP,N), CHAR(LDP,N), SPEED(LDP,N),
3 COLOUR(LDP,N), ZOMMA(LDP,N), VOMMA(LDP,N)
CHARACTER*1 CALPUT
```

3 Description

S30BBF 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, \( S_{\text{min}} \), observed during the lifetime of the contract. A put option gives the holder the right to sell the underlying asset at the maximum price, \( S_{\text{max}} \), observed during the lifetime of the contract. Thus, at expiry, the payoff for a call option is \( S / C_0 \) \( S_{\text{min}} \) \( e^{rT} \), and for a put, \( S_{\text{max}} / C_0 \) \( S_{\text{min}} \) \( e^{rT} \) - \( S_{\text{max}} \).

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

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

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

The corresponding put price is

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

In the above, \( \Phi \) denotes the cumulative Normal distribution function,

\[
\Phi(x) = \int_{-\infty}^{x} \phi(y)dy,
\]

where \( \phi \) denotes the standard Normal probability density function,

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

and

Acknowledgments

This work is based on material from SABRINA project.

References

[NP3666/22]
\[
\begin{align*}
    a_1 &= \frac{\ln(S/S_m) + \left(b + \sigma^2/2\right)T}{\sigma \sqrt{T}} \\
    a_2 &= a_1 - \sigma \sqrt{T}
\end{align*}
\]

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

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

1: CALPUT – CHARACTER*1

On entry: 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: M – INTEGER

On entry: the number of minimum or maximum prices to be used.

Constraint: \(M \geq 1\).

3: N – INTEGER

On entry: the number of times to expiry to be used.

Constraint: \(N \geq 1\).

4: SM(M) – double precision array

On entry: \(SM(i)\) must contain \(S_{\text{min}}(i)\), the \(i\)th 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:

- if CALPUT = 'C', \(SM(i) \leq S\), for \(i = 1, 2, \ldots, M\);
- if CALPUT = 'P', \(SM(i) \geq S\), for \(i = 1, 2, \ldots, M\);
- otherwise \(SM(i) \geq z\) and \(SM(i) \leq 1/z\), where \(z = \text{X02AMF()}\), the safe range parameter, for \(i = 1, 2, \ldots, M\).

5: S – double precision

On entry: \(S\), the price of the underlying asset.

Constraint: \(S \geq z\) and \(S \leq 1/z\), where \(z = \text{X02AMF()}\), the safe range parameter.

6: T(N) – double precision array

On entry: \(T(i)\) must contain \(T_i\), the \(i\)th time, in years, to expiry, for \(i = 1, 2, \ldots, N\).

Constraint: \(T(i) \geq z\), where \(z = \text{X02AMF()}\), the safe range parameter, for \(i = 1, 2, \ldots, N\).
7: SIGMA – double precision
   Input
   On entry: the volatility of the underlying asset. Note that a rate of 15% should be entered as 0.15.
   Constraint: SIGMA > 0.0.

8: R – double precision
   Input
   On entry: 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 R ≠ Q.

9: Q – double precision
   Input
   On entry: the annual continuous yield rate. Note that a rate of 8% should be entered as 0.08.
   Constraint: Q ≥ 0.0 and Q ≠ R.

10: P(LDP,N) – double precision array
    Output
    On exit: the leading M and N part of the array P contains the computed option prices.

11: LDP – INTEGER
    Input
    On entry: the first dimension of the arrays P, DELTA, GAMMA, VEGA, THETA, RHO, CRHO, VANNA, CHARM, SPEED, COLOUR, ZOMMA and VOMMA as declared in the (sub)program from which S30BBF is called.
    Constraint: LDP ≥ M.

12: DELTA(LDP,N) – double precision array
    Output
    On exit: the leading M and 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.

13: GAMMA(LDP,N) – double precision array
    Output
    On exit: the leading M and 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.

14: VEGA(LDP,N) – double precision array
    Output
    On exit: the leading M and N part of the array VEGA contains the sensitivity, \( \frac{\partial P}{\partial \sigma} \), of the option price to change in the volatility of the underlying asset.

15: THETA(LDP,N) – double precision array
    Output
    On exit: the leading M and N part of the array THETA contains the sensitivity, \( \frac{\partial P}{\partial T} \), of the option price to change in the time to expiry of the option.

16: RHO(LDP,N) – double precision array
    Output
    On exit: the leading M and N part of the array RHO contains the sensitivity, \( \frac{\partial P}{\partial r} \), of the option price to change in the annual risk-free interest rate.

17: CRHO(LDP,N) – double precision array
    Output
    On exit: the leading M and N part of the array CRHO containing the sensitivity, \( \frac{\partial P}{\partial b} \), of the option price to change in the annual cost of carry rate, b, where b = r – q.
18: VANNA(LDP,N) – double precision array

On exit: the leading M and N part of the array VANNA contains the sensitivity, \( \frac{\partial V}{\partial S} \), of VEGA to change in the price of the underlying asset or, equivalently, the sensitivity of DELTA to change in the volatility of the asset price.

19: CHARM(LDP,N) – double precision array

On exit: the leading M and N part of the array CHARM contains the sensitivity, \( \frac{\partial \Delta}{\partial T} \), of DELTA to change in the time to expiry of the option.

20: SPEED(LDP,N) – double precision array

On exit: the leading M and N part of the array SPEED contains the sensitivity, \( \frac{\partial \Gamma}{\partial S} \), of GAMMA to change in the price of the underlying asset.

21: COLOUR(LDP,N) – double precision array

On exit: the leading M and N part of the array COLOUR contains the sensitivity, \( \frac{\partial \Gamma}{\partial T} \), of GAMMA to change in the time to expiry of the option.

22: ZOMMA(LDP,N) – double precision array

On exit: the leading M and N part of the array ZOMMA contains the sensitivity, \( \frac{\partial \Gamma}{\partial \sigma} \), of GAMMA to change in the volatility of the underlying asset.

23: VOMMA(LDP,N) – double precision array

On exit: the leading M and N part of the array VOMMA contains the sensitivity, \( \frac{\partial \Gamma}{\partial \sigma} \), of VEGA to change in the volatility of the underlying asset.

24: IFAIL – INTEGER

Input/Output

On entry: IFAIL must be set to 0, −1 or 1. If you are unfamiliar with this parameter you should refer to Section 2.3 in the Essential Introduction for details.

On exit: IFAIL = 0 unless the routine detects an error (see Section 6).

For environments where it might be inappropriate to halt program execution when an error is detected, the value −1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this parameter, the recommended value is 0. When the value −1 or 1 is used it is essential to test the value of IFAIL on exit.

6 Error Indicators and Warnings

If on entry IFAIL = 0 or −1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL = 1

On entry, CALPUT ≠ 'C' or 'P'.

IFAIL = 2

On entry, M ≤ 0.

IFAIL = 3

On entry, N ≤ 0.
IFAIL = 4
On entry, SM(i) < z or SM(i) > 1/z, where z = X02AMF(), the safe range parameter,
or CALPUT = 'C' and SM(i) > S,
or CALPUT = 'P' and SM(i) < S.

IFAIL = 5
On entry, S < z or S > 1/z, where z = X02AMF(), the safe range parameter.

IFAIL = 6
On entry, T(i) < z, where z = X02AMF(), the safe range parameter.

IFAIL = 7
On entry, SIGMA ≤ 0.0.

IFAIL = 8
On entry, R < 0.0.

IFAIL = 9
On entry, Q < 0.0.

IFAIL = 11
On entry, LDP < M.

IFAIL = 12
On entry, R = Q.

7 Accuracy
The accuracy of the output is dependent on the accuracy of the cumulative Normal distribution function, \( \Phi \). 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 S15ABF and S15ADF). 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 minimum 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
* S30BBF Example Program Text
* Mark 22 Release. NAG Copyright 2007.
* .. Parameters ..
INTEGER NIN, NOUT
PARAMETER (NIN=5,NOUT=6)
INTEGER LDP, MMAX, NMAX
PARAMETER (LDP=10,MMAX=10,NMAX=10)
* .. Local Scalars ..
DOUBLE PRECISION Q, R, S, SIGMA
INTEGER I, IFAIL, J, M, N
CHARACTER PUT
.. Local Arrays ..

DOUBLE PRECISION CHARM(LDP,NMAX), COLOUR(LDP,NMAX),
+ CRHO(LDP,NMAX), DELTA(LDP,NMAX), GAMMA(LDP,NMAX),
+ P(LDP,NMAX), RHO(LDP,NMAX), SM(MMAX),
+ SPEED(LDP,NMAX), T(NMAX), THETA(LDP,NMAX),
+ VANNA(LDP,NMAX), VEGA(LDP,NMAX), VOMMA(LDP,NMAX),
+ ZOMMA(LDP,NMAX)

.. External Subroutines ..

EXTERNAL S30BBF

.. Executable Statements ..

WRITE (NOUT,'*') 'S30BBF Example Program Results'

READ (NIN,*)

READ (NIN,*) S, SIGMA, R, Q

READ (NIN,*) M, N

IF (M.LE.MMAX .AND. N.LE.NMAX) THEN

READ (NIN,*) (SM(I),I=1,M)

READ (NIN,*) (T(I),I=1,N)

IFAIL = 1

CALL S30BBF(PUT,M,N,SM,S,T,SIGMA,R,Q,P,LDP,DELTA,GAMMA,VEGA,
+ THETA,RHO,CRHO,VANNA,CHARM,SPEED,COLOUR,ZOMMA,
+ VOMMA,IFAIL)

IF (IFAIL.EQ.0) THEN

IF (PUT.EQ.'C') THEN

ELSE IF (PUT.EQ.'P') THEN

END IF

END IF

WRITE (NOUT,*) 'Spot = ', S
WRITE (NOUT,*) 'Volatility = ', SIGMA
WRITE (NOUT,*) 'Rate = ', R
WRITE (NOUT,*) 'Dividend = ', Q

WRITE (NOUT,*)

DO 60 J = 1, N

WRITE (NOUT,*)

WRITE (NOUT,99999) T(J)

WRITE (NOUT,99998) SM(I), P(I,J), DELTA(I,J),
+ GAMMA(I,J), VEGA(I,J), THETA(I,J), RHO(I,J),
+ CRHO(I,J)

20 CONTINUE

WRITE (NOUT,*)

WRITE (NOUT,99998) SM(I), P(I,J), VANNA(I,J),
+ CHARM(I,J), SPEED(I,J), COLOUR(I,J), ZOMMA(I,J),
+ VOMMA(I,J)

40 CONTINUE

60 CONTINUE
END IF

ELSE

WRITE (NOUT,99997) IFAIL

END IF

S30BBF

NAG Library Manual

S30BBF .6 [NP3666/22]
* 99999 FORMAT (1X,'Time to Expiry : ',1X,F8.4)
99998 FORMAT (8(1X,F9.4))
99997 FORMAT (1X,** S30BBF returned with IFAIL = ',I5)
END

9.2 Program Data

S30BBF Example Program Data

'P' : Call = 'C', Put = 'P'
87.0 0.3 0.06 0.04 : S, SIGMA, R, Q
1 1 : M, N
100.0 : SM(I), I = 1,2,...M
0.5 : T(I), I = 1,2,...N

9.3 Program Results

S30BBF Example Program 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
S-Max/Min Price Vanna Charm Speed Colour Zomma Vomma
100.0000 18.3530 1.9141 -0.6199 0.0007 0.0221 -0.0648 76.1292