11 隐式有限差分法计算期权价格

使用申明*

2021 年 2 月 25 日

目录

1 简介

我们这里也以美式看跌期权为例。类似显式有限差分法，我们也先将微分方程表示为差分形式。只是这时 $\frac{\partial f}{\partial t}(i,j)$ 的表示是向前近似。首先同样的，我们让

$$\Delta S = \frac{S_{max}}{M},\ \Delta t = \frac{T}{N},\ S_{max} = 3S_0,\ f(i,j) = f(i\Delta t, j\Delta S),\ S(i,j) = j\Delta S\ . \tag{1}$$

然后由

$$\frac{\partial f}{\partial t}(i,j) + rS(i,j)\frac{\partial f}{\partial S}(i,j) + \frac{1}{2}\sigma^2 S(i,j)\frac{\partial^2 f}{\partial S^2}(i,j) = rf(i,j)\ , \tag{2}$$

$$\frac{\partial f}{\partial t}(i,j) = \frac{f(i+1,j) - f(i,j)}{\Delta t}\ , \tag{3}$$

$$\frac{\partial f}{\partial S}(i,j) = \frac{f(i,j+1) - f(i,j-1)}{2\Delta S}\ , \tag{4}$$

$$\frac{\partial^2 f}{\partial S^2}(i,j) = \frac{f(i,j+1) + f(i,j-1) - 2f(i,j)}{\Delta S^2}\ . \tag{5}$$

得到

$$f(i+1,j) = a_j f(i,j-1) + b_j f(i,j) + c_j f(i,j+1)\ , \tag{6}$$

$$a_j = \frac{1}{2}rj\Delta t - \frac{1}{2}\sigma^2 j^2 \Delta t\ , \tag{7}$$

$$b_j = 1 + r\Delta t + \sigma^2 j^2 \Delta t\ , \tag{8}$$

$$c_j = -\frac{1}{2}rj\Delta t - \frac{1}{2}\sigma^2 j^2 \Delta t\ . \tag{9}$$

然后对于所有 i 时刻的 $f(i,j)$ 和所有 $i+1$ 时刻的 $f(i+1,j)$ 就会有关系

$$\begin{pmatrix} b_1 & c_1 & & & & \\ a_2 & b_2 & c_2 & & & \\ & a_3 & b_3 & c_3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & a_{M-2} & b_{M-2} & c_{M-2} \\ & & & & a_{M-1} & b_{M-1} \end{pmatrix} \begin{pmatrix} f(i,1) \\ f(i,2) \\ f(i,3) \\ \vdots \\ f(i,M-2) \\ f(i,M-1) \end{pmatrix} = \begin{pmatrix} f(i+1,1) - a_1 f(i,0) \\ f(i+1,2) \\ f(i+1,3) \\ \vdots \\ f(i+1,M-2) \\ f(i_2,M-1) - c_{M-1} f(i,M) \end{pmatrix}\ . \tag{10}$$

*作者不对内容正确性负责。如果您希望使用部分内容作为报告、文章内容等，请您注明内容来源为"金融工程资料小站"网站。

我们可以将其记为

$$M \cdot F_i = F_{i+1} \ . \tag{11}$$

其中 M, F_{i+1} 都是已知的，F_i 为上一时刻除边界外的未知价格。对 M 求逆后，我们有

$$F_i = M^{-1} F_{i+1} \ . \tag{12}$$

然后加上上下边界 $f(i, M) = 0$, $f(i, 0) = K$，即得到由差分关系递推出的 i 时刻期权价格。然后考虑在每个格点处是否应该执行美式看跌期权，就完成了一次美式看跌期权价格的递推。这样重复进行到初始时刻，即可得到由隐式差分法计算出的美式看跌期权价格。

2 计算美式看跌期权价格步骤

1. 确定时间和股价变化范围，$0 \le t \le T$, $0 \le S \le S_{max}$，$S_{max} = 3S_0$，并将区间离散化为二维等间距网格点，$\Delta t = T/N$, $\Delta S = S_{max}/M$。
2. 确定 T 时刻期权价格 $f(N, j) = \max(0,\ K - j\Delta S)$。
3. 往上一时刻递推，由 $F_i = M^{-1} F_{i+1}$，先计算出 M 和 F_{i+1}，再求出 M^{-1}，即得到 $F_i = (f(i,1), f(i,2), f(i,3), ..., f(i, M-1))^T$，加上上下边界边界条件 $f(i, 0) = K$, $f(i, M) = 0$，就得出 i 时刻所有股价对应的 $f(i, j)$。
4. 考虑应不应该在 i 时刻的格点处执行美式看跌期权，即更新 $f(i, j) = \max(f(i, j),\ K - j\Delta S)$。
5. 重复步骤 3，4，直到初始时刻，然后使用线性插值法得出 $S = S_0$ 处的期权价格。

3 步骤 Python 代码实现

```python
import numpy as np

def American_put_implicit(r, sigma, S_0, K, T, M, N):
    dS = 3*S_0/M
    dt = T/N
    # 将a_j, b_j, c_j 写为3个函数。
    a = lambda j: 0.5*r*j*dt-0.5*sigma*sigma*j*j*dt
    b = lambda j: 1+r*dt+sigma*sigma*j*j*dt
    c = lambda j: -0.5*r*j*dt-0.5*sigma*sigma*j*j*dt
    # f1和f2为2列用来迭代计算的期权价格。
    f1 = [max(K-i*dS, 0) for i in range(M+1)]
    f2 = [None for i in range(M+1)]
    # coeffs为上文中的M系数矩阵。
    coeffs = np.zeros((M-1, M-1))

    for i in range(N-1, -1, -1):
        f2 = list(f1)
        coeffs[0][0] = b(1)
        coeffs[0][1] = c(1)
        coeffs[M-2][M-2] = b(M-1)
        coeffs[M-2][M-3] = a(M-1)
        for j in range(2, M-1, 1):
            coeffs[j-1][j-2] = a(j)
            coeffs[j-1][j-1] = b(j)
            coeffs[j-1][j] = c(j)
        # 参数矩阵求逆。
```

```
        coeffs_inv = np.linalg.inv(coeffs)
        F2 = f2[1:-1]
        F2[0] = F2[0]-a(1)*K
        F1 = np.matmul(coeffs_inv, F2)

        f1[1:M] = F1
        f1[0] = K
        f1[M] = 0

        #判断是否执行美式看跌期权。
        f1 = np.maximum(f1, K-np.linspace(0, M, M+1)*dS)

    pos = int(S_0/dS)
    put_price = f1[pos]+(f1[pos+1]-f1[pos])/dS*(S_0-pos*dS)

    return put_price


# 计算例子。
if __name__ == "__main__":
    put_price = American_put_implicit(0.1, 0.4, 50, 50, 5/12.0, 300, 300)
    print("American put price: {0:0.5f}".format(put_price))

Output:
    American put price: 4.27847
```

4 参考资料

参考文献

[1] 《期权、期货及其他衍生产品》（原书第 9 版）第 21 章，John C. Hull 著，王勇、索吾林译，机械工业出版社，2014.11 。