金融数学 —— 估值与风险测度

1. 估值——欧式期权[1]

蒙特卡洛模拟最重要的应用之一是*未定权益*（期权、衍生品、混合型工具等）的估值。在风险中立的世界中，未定权益的价值是风险中立(鞅Martingale[3])测度下的折现后预期收益。

金融期权表示在规定(行权期)日期(欧式期权)或者规定时期(美式期权)内，以规定价格(所谓*有权价*)购买(看涨期权)或者出售(看跌期权) 指定金融工具。

对于某指数的欧式看期权到期日收益通过$h(S_T)=max(S_T-K,0)$，其中S_T是到期日T的指数水平，K是行权价格。给定相关随机过程(例如几何布朗运动)的风险中立测度，或由下式给出：

$$C_0=e^{-rT}E_0^Q(h(S_T))=e^{-rT}\int_0^\infty h(s)q(s)ds \tag{1}$$

对于欧式期权对应的蒙特卡洛模拟公式如下，其中$\tilde{S}_T^i$是到期日的第i个模拟指数水平。

$$\tilde{C_0}=e^{-rT}\frac{1}{I}\sum_{i=1}^{l}h(\tilde{S_T}^i) \tag{2}$$

Python代码实现（模拟到期日的指数水平）：

```
In [56]: S0 = 100.
         r = 0.05
         sigma = 0.25
         T = 1.0
         I = 50000
         def gbm_mcs_stat(K):
             ''' Valuation of European call option in Black-Scholes-Merton
             by Monte Carlo simulation (of index level at maturity)

             Parameters
             ==========
             K : float
                 (positive) strike price of the option

             Returns
             =======
             C0 : float
                 estimated present value of European call option
             '''
             sn = gen_sn(1, I)
             # simulate index level at maturity
             ST = S0 * np.exp((r - 0.5 * sigma ** 2) * T
                          + sigma * np.sqrt(T) * sn[1])
             # calculate payoff at maturity
             hT = np.maximum(ST - K, 0)
             # calculate MCS estimator
             C0 = np.exp(-r * T) * 1 / I * np.sum(hT)
             return C0
```

作为参考，考虑行权价 K=105 的情况：

```
In [57]: gbm_mcs_stat(K=105.)
Out[57]: 10.044221852841922
```

2. 估值——美式期权[1]

在美式期权中，必须要解决*最优截止问题*，提出期权的公允价值。下式将美式期权作为最优截止问题的估值公式，该问题的公式化基于离散的时间网格（准确来说，下式是百慕大式期权的估值公式，时间间隔收敛于0长度时，百慕大期权的价值收敛于没事期权的价值）。

以最优截止问题形式出现的美式期权价格：

$$V_0 = \sup e^{-r\tau} E_0^Q(h_\tau(S_\tau)) \tag{3}$$
$$\tau \in \{0, \Delta t, 2\Delta t, \ldots, T\}$$

最小二乘蒙特卡洛方法：

由$V_t(s) = max(h_t(s), C_t(s))$,其中$C_t(s) = E_t^Q(e^{-r\Delta t}V_{t+\Delta t}(S_{t+\Delta t})|S_t = s)$,可知，给出的任何给定日期$t$的美式期权价值是给定指数$S_t = s$下的期权持续价值。

先考虑在M个等长(Δt)的时间间隔中模拟指数水平的I条路径。定义$Y_{t.i} = e^{-r\Delta t}V_{t+\Delta t,i}$为路径$i$在时间$t$时的模拟持续价值。可使用所有模拟持续价值的截面，通过最小二乘回归估算（预期）持续价值。

给定一组基函数$b_d, d = 1, \ldots, D$,然后利用回归估算公式：$\hat{C_{t,i}} = \sum_{d=1}^{D} \alpha_{d,t}^* b_d(S_{t,i})$计算出实际价值，其中最优回归参数$\alpha^*$是下式中得到的最小二乘问题的解。

美式期权估值的最小二乘回归：

$$\min_{a_{1,i},\ldots,a_{D,i}} \frac{1}{L}\sum_{i=1}^{I}\left(Y_{t,i}-\sum_{d=1}^{D}\alpha_{d,t}b_d(S_{t,i})\right)^2 \tag{4}$$

LSM算法美式看涨期权和看跌期权的Python实现：

```
In [64]: def gbm_mcs_amer(K, option='call'):
             ''' Valuation of American option in Black-Scholes-Merton
             by Monte Carlo simulation by LSM algorithm

             Parameters
             ==========
             K : float
                 (positive) strike price of the option
             option : string
                 type of the option to be valued ('call', 'put')

             Returns
             =======
             C0 : float
                 estimated present value of European call option
             '''
             dt = T / M
             df = np.exp(-r * dt)
             # simulation of index levels
             S = np.zeros((M + 1, I))
             S[0] = S0
             sn = gen_sn(M, I)
             for t in range(1, M + 1):


                 S[t] = S[t - 1] * np.exp((r - 0.5 * sigma ** 2) * dt
                         + sigma * np.sqrt(dt) * sn[t])
             # case-based calculation of payoff
             if option == 'call':
                 h = np.maximum(S - K, 0)
             else:
                 h = np.maximum(K - S, 0)
             # LSM algorithm
             V = np.copy(h)
             for t in range(M - 1, 0, -1):
                 reg = np.polyfit(S[t], V[t + 1] * df, 7)
                 C = np.polyval(reg, S[t])
                 V[t] = np.where(C > h[t], V[t + 1] * df, h[t])
             # MCS estimator
             C0 = df * 1 / I * np.sum(V[1])
             return C0
In [65]: gbm_mcs_amer(110., option='call')
Out[65]: 7.7789332794493156
In [66]: gbm_mcs_amer(110., option='put')
Out[66]: 13.614023206242445
```

3. 风险测度[2]

风险价值(VaR)是最广泛使用的风险测度之一。VaR是一个以货币单位表示的数字，表示在给定时间周期中不超过某种置信度(概率)的损失（投资组合、股票头寸等）。

参考资料

[1] 《Python金融大数据分析》 第十章 10.3-估值
[2] 《Python金融大数据分析》 第十章 10.4-风险测度
[3] 知乎：[martingale]
(https://www.zhihu.com/question/61223457/answer/186180657)

Definition 5.1 $X = (X_t, t \geq 0)$ *is said to be a martingale (supermartingale, submartingale) with respect to* $(\mathcal{F}_t, t \geq 0)$ *if*

(i) X_t *is* $\mathcal{F}_t$ *-measurable (determined),*

(ii)X_t *is integrable, i.e.,* $E[|X_t|] < \infty$,

(iii) for every $s \leq t$,

$$E[X_t|\mathcal{F}_s] = X_s \quad (E[X_t|\mathcal{F}_s] \leq X_s, \quad E[X_t|\mathcal{F}_s] \geq X_s)$$