强化学习在期权定价中的应用

1. 摘要

本文首先通过构建期权“复制组合“（Replicating Portfolio）方式将最优期权定价问题转化成强化学习下的最小化期权价格和风险溢价的问题。然后展示了两种强化学习算法在离散时间下对期权定价的应用，分别是Model-based Dynamic Programming算法和Model-free Deep Deterministic Policy Gradient（DDPG）算法。

2. 简介

强化学习是目前在机器学习和深度学习领域较为热门的专业，通过与外界环境互动来解决序列决策问题以达到奖励最大化，其非监督学习的特性能够使之在已知或未知的环境（Environment）下学习到全局最优的解决方案。这种设定使强化学习非常适合解决金融领域的问题，其两者本质都是人（强化学习是Agent）以过去有限的信息在金融市场中达到最优状态，以期权定价为例，期权的卖方需要在当前时刻定出期权的价格，该期权价格不能过高，否则失去市场竞争力，也不能太低，否则不能覆盖未来期权买房行权所支付的金额。本质上期权价格是买卖双方在当前时刻下的均衡状态下的价格。

强化学习在对环境已知或未知下的分两种，当环境已知时时Model-Based强化学习，即已知环境的转移概率，在这种设定下，序列决策问题下最大化奖励可以通过从结果反向推导出最优行动策略（Policy）的方法解决。另一种当环境未知时Model-free强化学习，通过不断探索（Exploration）和利用（Exploitation）已知信息学习到最优行动策略。

3. 期权定价问题

3.1 构建期权复制组合

我们从欧式期权卖方的角度出发，利用BSM模型中对期权复制组合的定义，在无交易摩擦成本下，通过复制组合中股票和现金来模拟期权，在 t 时刻

$$\Pi_t = u_t S_t + B_t \tag{1}$$

其中 u_t 是组合中股票的仓位，Π_t ， S_t 和 B_t 分别是复制组合价格、股票价格和现金金额

那么在期权到期时 T 时刻，我们假设 $u_T = 0$ 即将全部股票平仓转成现金用于支付买方，所以

$$\Pi_T = B_T = H_T(S_T) \tag{2}$$

其中 $H_T(S_T)$ 是到期日期权的收益。

为了确定组合中现金在 $t < T$ 的金额，我们假设复制组合是自融资的（Self-financing），那么在 $t+1$ 时刻对组合股票进行调仓（Rebalance）有如下关系

$$u_t S_{t+1} + e^{r\Delta t} B_t = u_{t+1} S_{t+1} + B_{t+1} \tag{3}$$

公式 (3) 可以将复制组合中的现金写成反向计算的模式

$$B_t = e^{-r\Delta t}\left[B_{t+1} + (u_{t+1} - u_t) S_{t+1}\right], \ \ t = T-1, \cdots, 0 \tag{4}$$

将公式 (4) 带入公式 (2) 可以得到复制组合价格的反向计算公式

$$\Pi_t = e^{-r\Delta t}\left[\Pi_{t+1} - u_t \Delta S_t\right], \ \ \Delta S_t = S_{t+1} - e^{r\Delta t} S_t, \ \ t = T-1, \cdots, 0 \tag{5}$$

3.2最优行动策略和期权本质价格

首先定义期权的本质价格（Fair price）$\hat{C}_t$ 为复制组合在 t 时刻的期望价格

$$\hat{C}_t = \mathbb{E}_t \left[\Pi_t \right] \tag{6}$$

由于复制组合和现金都可以通过下一时刻的信息和股票价格来反向推导，那么给定一个股票仓位策略 $\{u_t\}_{t=0}^T$ 和股票价格 $S_1, S_2, \cdots, S_T$ 我们可以推导出所有复制组合和现金的信息，从而也能决定该策略下的期权价格。但是由于策略的不同会得到不同的期权价格， 这就需要最优行动策略。

最优行动策略的定义决定了对强化学习要解决的问题，在下文分别给出两种定义来构建强化学习问题，分别为在未来股票所有路径上最小化复制组合的波动，对应着Model-based Dynamic Programming问题；和单一股票路径上最小化组合价格和风险溢价，对应着Model-free Q-learning问题。

4. Model-based Dynamic Programming问题

4.1 期权价格和风险溢价

由于是Model-based问题，我们假设股票价格模型已知，本文使用对数正态分布来模拟未来股票价格路径（也可以使用其他模型来模拟），那么使用蒙特卡洛模拟可以得到 N 条股票价格路径，此时最优行动策略 $\{u_t^*(S_t)\}_{t=0}^T$ 是最小化所有路径上的复制组合的波动（Cross-sectional variance minimization）

$$\begin{aligned} u_t^*(S_t) &= \underset{u}{\operatorname{argmax}} \operatorname{Var}[\Pi_t | \mathcal{F}_t] \\ &= \underset{u}{\operatorname{argmax}} \operatorname{Var}[\Pi_{t+1} - u_t \Delta S_t | \mathcal{F}_t], \ \ t = T-1, \cdots, 0 \end{aligned} \tag{7}$$

其中 $\mathcal{F}_t$ 为 t 时刻及以前所有股票价格路径上的信息，由于 Π_t 是可以反向计算的，所以 $u_t^*(S_t)$ 也可以通过使公式 (7) 导数为零反向计算，即

$$u_t^*(S_t) = \frac{\operatorname{Cov}(\Pi_{t+1}, \Delta S_t | \mathcal{F}_t)}{\operatorname{Var}(\Delta S_t | \mathcal{F}_t)}, \ \ t = T-1, \cdots, 0 \tag{8}$$

由于在该问题设定下，我们有多个股票路径，那么期权本质价格的定义为

$$\hat{C}_t = \mathbb{E}_t \left[\Pi_t | \mathcal{F}_t \right] \tag{9}$$

带入公式 (5) 可以得到

$$\begin{aligned} \hat{C}_t &= \mathbb{E}_t \left[e^{-r\Delta t} \Pi_{t+1} | \mathcal{F}_t \right] - u_t(S_t) \mathbb{E}_t \left[\Delta S_t | \mathcal{F}_t \right] \\ &= \mathbb{E}_t \left[e^{-r\Delta t} \mathbb{E}_{t+1} \left[\Pi_{t+1} | \mathcal{F}_{t+1} \right] | \mathcal{F}_t \right] - u_t(S_t) \mathbb{E}_t \left[\Delta S_t | \mathcal{F}_t \right] \\ &= \mathbb{E}_t \left[e^{-r\Delta t} \hat{C}_{t+1} | \mathcal{F}_t \right] - u_t(S_t) \mathbb{E}_t \left[\Delta S_t | \mathcal{F}_t \right], \ \ t = T-1, \cdots, 0 \end{aligned} \tag{10}$$

由公式 (10) 可以得到期权本质价格的Bellman Equation。

但是作为期权卖方，不能仅仅使用期权本质价格，需要对未来复制组合的波动进行风险溢价（有很多种理解方式，例如进行风险溢价以保证在卖期权时构建复制组合的钱足以覆盖未来波动，以保证自融资的方式进行；或者例如期权作为金融市场存在的投资工具，有内在风险溢价的需求）。配合之前对最优行动策略的定义，此时期权价格为期权本质价格加上未来所有股票路径上复制组合价格的波动风险溢价，即

$$C_0 = \mathbb{E}_0 \left[\Pi_0 + \lambda \sum_{t=0}^{T} e^{-rt} \operatorname{Var}[\Pi_t | \mathcal{F}_t] \right] \tag{11}$$

其中 λ 为风险厌恶系数。

4.2 Model-based马尔科夫决策过程（MDP）问题

如果将期权定价作为MDP问题，期权卖方就是强化学习中的Agent，其对复制组合中股票仓位的控制就是强化学习中的Action。我们定义行动策略（Policy），即对复制组合调仓的策略 $\pi(t,S_t)$ 为确定性策略（Deterministic policy）

$$\begin{aligned}\pi:\{0,\cdots,T-1\}\times\mathcal{S}\to\mathcal{A}\\ u_t=\pi(t,S_t)\end{aligned}$$

其中 $\mathcal{S}$ 和 $\mathcal{A}$ 分别为MDP的状态空间（State space）和动作空间（Action space）

我们定义MDP的价值函数（Value function）在策略 π 下 $V_t^\pi(S_t)$ 为带有风险溢价的期权价格的负数，以满足强化学习最大化价值函数的目的

$$\begin{aligned}V_t^\pi(S_t)&=\mathbb{E}_t\left[-\Pi_t-\lambda\sum_{t'=t}^{T}e^{-r(t'-t)}\,\mathrm{Var}[\Pi_{t'}|\mathcal{F}_{t'})]\,\middle|\,\mathcal{F}_t\right]\\&=\mathbb{E}_t\left[-\Pi_t-\lambda\,\mathrm{Var}[\Pi_t]-\lambda\sum_{t'=t+1}^{T}e^{-r(t'-t)}\,\mathrm{Var}[\Pi_{t'}|\mathcal{F}_{t'})]\,\middle|\,\mathcal{F}_t\right]\end{aligned}\tag{12}$$

由公式 (12) 可以构建价值函数的Bellman equation

$$V_t^\pi(S_t)=\mathbb{E}_t^\pi\left[R(S_t,u_t,S_{t+1})+\gamma V_{t+1}^\pi(S_{t+1})\right],\ \gamma\equiv e^{-r\Delta_t}\tag{13}$$

其中MDP的奖励函数由价值函数的定义而得来

$$\begin{aligned}R_T(S_t,u_t,S_{t+1})&=\gamma u_t\Delta S_t-\lambda\,\mathrm{Var}[\Pi_t|\mathcal{F}_t]\\&=\gamma u_t\Delta S_t-\lambda\gamma^2\mathbb{E}_t\left[\hat{\Pi}_{t+1}^2-2u_t\Delta\hat{S}_t\hat{\Pi}_{t+1}+u_t^2\left(\Delta\hat{S}_t\right)^2\right]\end{aligned}\tag{14}$$

其中 $\hat{\Pi}_{t+1}=\Pi_{t+1}-\bar{\Pi}_{t+1}$，$\bar{\Pi}_{t+1}$ 是股票价格所有路径上复制组合价格的样本均值，$\bar{S}_{t+1}$ 同理。奖励函数可以理解为每一次Agent调仓，要最大化的是复制组合的收益减去复制组合可能的波动风险。注意到公式 (14) 是对于 u_t 的二次函数，那么价值函数 $V_t(S_t)$ 是奖励函数的线性组合，也是对于 u_t 的二次函数，理论上存在 u_t 使得价值函数最大化。

类似的，根据公式 (12) 构建动作函数（Action value function或者Q function）

$$Q_t^\pi(s,u)=\mathbb{E}_t\left[-\Pi_t|S_t=s,u_t=u\right]-\lambda\mathbb{E}_t^\pi\left[\sum_{t'=t}^{T}e^{-r(t'-t)}\,\mathrm{Var}[\Pi_t'|\mathcal{F}_t]\middle|S_t=s,u_t=u\right]\tag{15}$$

MDP问题中的最优行动策略（Optimal Policy）$\pi_t^*(S_t)$ 定义为使得价值函数最大的行动策略，也等价于使动作函数最大的行动策略

$$\pi_t^*(S_t)=\underset{\pi}{\mathrm{argmax}}\,V_t^\pi(S_t)=\underset{u_t\in\mathcal{A}}{\mathrm{argmax}}\,Q_t^*(S_t,u_t)\tag{16}$$

在最优行动策略的定义下，最优价值函数（Optimal value function）满足Bellman optimality equation

$$V_t^*(S_t)=\mathbb{E}_t^{\pi^*}\left[R_t(S_t,\pi_t^*(S_t),S_{t+1})+\gamma V_{t+1}^*(S_{t+1})\right]\tag{17}$$

相同的，最优动作函数（Optimal Q function）也满足Bellman optimality equation

$$Q_t^*(s,u)=\mathbb{E}_t\left[R_t(S_t,u_t,S_{t+1})+\gamma\max_{u_{t+1}\in\mathcal{A}}Q_{t+1}^*(S_{t+1},u_{t+1})|S_t=s,u_t=u\right]\tag{18}$$

最优价值函数和最优动作函数的终值条件在 $t=T$ 时成立，即期权行权

$$V_T^*(S_T)=Q_T^*(S_t,u_t=0)=-\Pi_T(S_T)-\lambda\,\mathrm{Var}[\Pi_T(S_T)]\tag{19}$$

4.3 最优策略（Optimal Policy）和期权价格

将公式 (14) 带入公式 (18) 我们得到

$$Q_t^*(S_t,u_t)=\gamma\mathbb{E}_t\left[Q_{t+1}^*(S_{t+1},u_{t+1}^*)\right]+\gamma\mathbb{E}_t\left[u_t\Delta S_t\right]-\lambda\gamma^2\mathbb{E}_t\left[\hat{\Pi}_{t+1}^2-2u_t\Delta\hat{S}_t\hat{\Pi}_{t+1}+u_t^2\left(\Delta\hat{S}_t\right)^2\right]\quad(20)$$

如果假设我们对股票的买卖对市场没有影响，那么等式右边第一个部分 $\mathbb{E}_t\left[Q_{t+1}^*(S_{t+1},u_{t+1}^*)\right]$ 与当前 u_t 无关，此时最优动作函数关于 u_t 是二次函数，理论上最优策略存在解析解

$$\pi_t^*(S_t)=u_t^*(S_t)=\frac{\mathbb{E}_t\left[\Delta\hat{S}_t\hat{\Pi}_{t+1}+\frac{1}{2\gamma\lambda}\Delta S_t\right]}{\mathbb{E}_t\left[\left(\Delta\hat{S}_t\right)^2\right]}\quad(21)$$

注意到求最优动作（Optimal action）和最优动作函数都在所有股票价格路径上可以反向计算，从 $T-1$ 计算到 0 时刻，那么 0 时刻期权的价格就是

$$C_0(S_0)=-Q_0^*(S_0,u_0^*(S_0))\quad(22)$$

而且与此同时 0 时刻的对冲比率（Hedge ratio）就是最优动作 $u_0^*(S_0)$，也就是BSM模型中的Delta。

4.4 Dynamic Programming算法

上一节所提到的最优动作和最优动作函数都可以通过在蒙特卡洛模拟的股票路径上实现反向计算，如果假设状态空间 $\mathcal{S}$ 和 动作状态 $\mathcal{A}$ 是离散的，我们就可以简单利用公式 (20) 和 公式 (21) 计算每一个离散股票价格下的最优动作和最优动作函数（假设蒙特卡洛模拟的股票路径足够多，每个离散点的股票价格都能足够多的样本来计算），从 T 时刻计算到 0 时刻。

但是生成如此大量的股票路径和计算每一个网格节点会极大地降低计算效率，在这里我们利用基函数的方式计算连续状态空间 $\mathcal{S}$ 和动作空间 $\mathcal{A}$ 下的问题，我们将最优动作和最优动作函数利用基函数在状态空间上展开

$$u_t^*(S_t)=\sum_n^M\phi_{nt}\Phi_n(S_t),\quad Q_t^*(S_t,u_t^*)=\sum_n^M\omega_{nt}\Phi_n(S_t)\quad(23)$$

其中 M 为基函数的节点个数（Knots）。带入公式 (23) 至公式 (20) 中计算最大时基函数对的系数 ϕ_n，注意到之前提到过公式 (20) 中的第一部分与 u_t 无关，所以省略，即解决如下问题

$$\phi^*=\underset{\phi}{\operatorname{argmin}}\sum_{k=1}^{N_{MC}}\left(\sum_n^M\phi_{nt}\Phi_n(S_t^{(k)})\Delta S_t^{(k)}+\gamma\lambda\left(\hat{\Pi}_{t+1}^{(k)}-\sum_n^M\phi_{nt}\Phi_n(S_t^{(k)})\Delta\hat{S}_t^{(k)}\right)^2\right)\quad(24)$$

其中 N_{MC} 为蒙特卡洛模拟股票路径的数目。解决公式 (24) 的问题，我们在每一个时刻 t 构建线性方程式

$$\begin{aligned}\sum_m^M A_{nm}^{(t)}\phi_{mt}&=B_n^{(t)},\ \ n=1,\cdots,M\\A_{nm}^{(t)}&=\sum_{k=1}^{N_{MC}}\Phi_n(S_t^{(k)})\Phi_m(S_t^{(k)})\left(\Delta\hat{S}_t^{(k)}\right)^2\\B_n^{(t)}&=\sum_{k=1}^{N_{MC}}\Phi_n(S_t^{(k)})\left[\hat{\Pi}_{t+1}^{(k)}\Delta\hat{S}_t^{(k)}+\frac{1}{2\gamma\lambda}\Delta S_t^{(k)}\right]\end{aligned}\quad(25)$$

那么最优动作对应的基函数系数 $\phi_t^*=\mathbf{A}_t^{-1}\mathbf{B}_t$。对于最优动作函数的基函数系数，因为我们利用蒙特卡洛模拟，所以我们可以将公式 (18) 改写为

$$Q_t^*(S_t,u_t^*)=R_t(S_t,u_t^*,S_{t+1})+\gamma\max_{u_{t+1}\in\mathcal{A}}Q_{t+1}^*(S_{t+1},u_{t+1})+\epsilon_t \tag{26}$$

其中 ϵ_t 是均值为零随机噪音，我们利用最小二乘优化法（least-square optimization）来求解基函数系数，将公式 (22) 带入公式 (26)，并写成最小二乘的形式使随机噪音最小（即 ω_t 拟合的最好）

$$\omega^*=\underset{\omega}{\operatorname{argmin}}\sum_{k=1}^{N_{MC}}\left(R_t(S_t^{(k)},u_t^*,S_{t+1}^{(k)})+\gamma\max_{u_{t+1}^*\in\mathcal{A}}Q_{t+1}^*(S_{t+1}^{(k)},u_{t+1})-\sum_n^M\omega_{nt}\phi_n\left(S_t^{(k)}\right)\right)^2 \tag{27}$$

类似的，在每一个时刻 t 构建线性方程式

$$\begin{aligned}
&\sum_m^M C_{nm}^{(t)}\omega_{mt}=D_n^{(t)},\ \ n=1,\cdots,M\\
C_{nm}^{(t)}&=\sum_{k=1}^{N_{MC}}\Phi_n(S_t^{(k)})\Phi_m(S_t^{(k)})\\
D_n^{(t)}&=\sum_{k=1}^{N_{MC}}\Phi_n(S_t^{(k)})\left[R_t(S_t^{(k)},u_t^*,S_{t+1}^{(k)})+\gamma\max_{u_{t+1}^*\in\mathcal{A}}Q_{t+1}^*(S_{t+1}^{(k)},u_{t+1})\right]
\end{aligned} \tag{28}$$

最优动作函数对应的基函数系数 $\omega_t^*=\mathbf{C}_t^{-1}\mathbf{D}_t$ 。

最后将公式 (25) 和公式 (28) 计算出来的最优动作和最优动作函数带入公式 (20) 中，按照从 $T-1$ 时刻计算到 0 时刻的反向计算方法，就可以计算出 0 时刻期权的价格和对冲比率。

5. Model-free Q-learning问题

5.1 Model-free马尔科夫决策过程（MDP）问题

对于Model-free问题，由于环境，即股票价格模型是未知的，我们不能通过蒙特卡洛模拟出的多条股票价格路径来反向计算最优动作函数。只能通过与环境的互动，获取样本 (S_t,u_t,R_t,S_{t+1}) 的方式找到最优行动策略。

我们本节主要是通过Q-learning的强化学习方法，站在期权卖方的角度，通过不断与环境交互，获取环境样本来迭代最优动作函数（Optimal Q-function），从而得到 0 时刻的期权价格。

我们沿用上一节中对Agent，Action和行动策略的定义，同时定义新的奖励函数

$$\begin{aligned}
R_t(S_t,u_t,S_{t+1})&=u_t(\gamma S_{t+1}-S_t)-H_t(S_t)\\
&=\gamma\Pi_{t+1}-\Pi_t-H_t(S_t),\quad t<T\\
R_T(S_T,u_T=0)&=-H_T(S_T),\quad t=T
\end{aligned} \tag{29}$$

其中$\gamma\equiv e^{-r\Delta_t}$ 沿用上文定义，为折现因子，也是强化学习中的Discount factor。

基于奖励函数，我们可以构建价值函数

$$\begin{aligned}
V_t(S_t)&=\mathbb{E}_t\left[\sum_{t'=t}^{T}\gamma^{t'}R_{t'}(S_{t'},u_{t'},S_{t'+1})\right]\\
&=\mathbb{E}_t\left[\sum_{t'=t}^{T}\gamma^{t'}\left(\gamma\Pi_{t'+1}-\Pi_{t'}-H_{t'}(S_{t'})\right)\right]\\
&=\mathbb{E}_t\Bigg[-\gamma^t\Pi_t+\gamma^{t+1}\Pi_{t+1}-\gamma^{t+1}\Pi_{t+1}\\
&\qquad+\gamma^{t+2}\Pi_{t+2}-\cdots-\gamma^{T-1}\Pi_{T-1}+\gamma^T\Pi_T-\sum_{t=t'}^{T}\gamma^{t'}H_{t'}(S_{t'})\Bigg]\\
&=\mathbb{E}_t\left[-\gamma^t\Pi_t+\gamma^T\Pi_T-\gamma^T H_T(S_T)\right]\\
&=\mathbb{E}_t\left[-\gamma^t\Pi_t\right]
\end{aligned} \tag{30}$$

由公式 (30) 可知，这样构建下的价值函数就是复制组合价格负数的期望，同时在给定行动策略的时候也满足Bellman equation

$$\begin{aligned} V_t^\pi(S_t) &= \mathbb{E}_t^\pi \left[\gamma^t R_t(S_t, u_t, S_{t+1}) + \sum_{t'=t+1}^{T} \gamma^{t'} R_{t'}(S_{t'}, u_{t'}, S_{t'+1}) \right] \\ &= \mathbb{E}_t^\pi \left[R(S_t, u_t, S_{t+1}) + \gamma V_{t+1}^\pi(S_{t+1}) \right] \end{aligned} \tag{31}$$

在给定的行动策略是最优行动策略时价值函数时，最优价值函数满足Bellman optimality equation。

同理，可以用类似方法证明，动作函数和最优动作函数也分别满足Bellman equation和Bellman optimality equation。

5.2 Model-free下的风险溢价

如果按照上一节的方法学习到的 $Q_0^*(S_0, u_0)$ ，即期初期权价格的负数，是前文提到的期权本质价格，所以需要加入额外的风险溢价来使期权卖方的定价尽可能地达到买卖双方的均衡状态。

对于Model-free问题，我们不能通过蒙特卡洛模拟出的多条股票价格路径来计算复制组合价格在横截面的方差，即不能计算 $\mathrm{Var}[\Pi_t|\mathcal{F}_t]$ 。所以我们需要寻找一种新的风险溢价来代替原来的价格方差，而这种新的风险溢价只能基于观测到的历史数据 (S_t, t) 。

沿用Model-based下的想法，我们依旧定义新的风险溢价为未来复制组合价格的波动，这里我们采用在每一次换仓的时刻（即Agent每次做出动作的时刻），奖励函数从原来的复制组合价格变动的基础上再减去价格变动的平方

$$\begin{aligned} R_t(S_t, u_t, S_{t+1}) &= u_t(\gamma S_{t+1} - S_t) - H_t(S_t) - [u_t(\gamma S_{t+1} - S_t)]^2 \\ &= \gamma\Pi_{t+1} - \Pi_t - H_t(S_t) - (\gamma\Pi_{t+1} - \Pi_t)^2, \quad t < T \\ R_T(S_T, u_T = 0) &= -H_T(S_T), \quad t = T \end{aligned} \tag{32}$$

其中 λ 为Agent的风险厌恶程度，此时 0 时刻的最优价值函数和最优动作函数为

$$V_0^*(S_0) = Q_0^*(S_0, u = u_0^*) = \sum_{t=0}^{T} \gamma^t R_t(S_t, u_t^*, S_{t+1}) = -\Pi_0 - \lambda \sum_{t=0}^{T} [u_t^*(\gamma S_{t+1} - S_t)]^2 \tag{33}$$

可以理解为起初时期权价格为期权的本质价格外加对未来期权复制组合价格波动的风险溢价，而最优行动策略 $\{u_t^*\}_{t=0}^T$ 使得期权价格在给定风险厌恶程度 λ 下最小。

5.3 传统Q-learning算法

对于传统的Q-learning 算法，在离散状态空间、离散动作空间和确定性行动策略下，将每一次样本 (S_t, u_t, R_t, S_{t+1}) 带入Q循环中（Q-iteration），动作函数最终会收敛到最优动作函数。

如果我们考虑股票价格 S_t 和复制组合股票仓位 u_t 都是离散的话，我们仅需要构建一个二维表格，在其中记录下所有的动作函数的数值 $Q(S_t, u_t)$ ，然后每一次新的样本 (S_t, u_t, R_t, S_{t+1}) 去迭代动作函数在 t 时刻的数值

$$Q_t(S_t, u_t) = R_t(S_t, u_t, S_{t+1}) + \gamma \max_{u_{t+1} \in \mathcal{A}} Q_{t+1}(S_{t+1}, u_{t+1}) \tag{34}$$

在足够多的样本下可以得到最优动作函数，而此时的最优行动策略就是

$$\pi_t^*(S_t) = \underset{u_t \in \mathcal{A}}{\mathrm{argmax}}\, Q_t^*(S_t, u_t) \tag{35}$$

尽管传统的Q-learning算法可以保证收敛到最优动作函数，但是这种方法无法处理连续状态和动作空间下的问题，且保证收敛的迭代次数会过大，降低学习效率。

5.4 DDPG算法

DDPG算法做为Q-learning算法的延申，有如下几点改进及其优势

1. 使用神经网络对策略函数 $\pi_t(S_t)$ 和动作函数 $Q_t(S_t,u_t)$ 进行模拟，可以处理连续状态和动作空间。
2. 使用额外两个目标神经网络，克服了训练神经网络时方差过大的问题，使学习过程更加稳定。
3. 加入了经验回放缓存池（replay buffer），充分利用了Q-learning的Off-policy特点，使得神经网络训练的时候可以使用mini-batch的方法，提高收敛速度，同时降低了样本之间的相关系数。

使用DDPG算法对本节中的动作函数和行动策略进行学习，当算法收敛时，最优行动策略 $\pi_0^*(S_0)$ 为期初的对冲比率，最优动作函数 $Q_t^*(S_t,\pi_0^*(S_0))$ 为期初的期权价格。

DDPG的具体做法是首先初始化策略神经网络（Actor）和动作函数神经网络（Critic），其参数分别为 θ 和 ϕ，记为 μ_θ 和 Q_ϕ，同时建立一个新的经验回放缓存池 $\mathcal{D}$。然后再建立两个目标神经网络（Target neural network）用于DDPG中估算TD target，它们的参数等于初始化后的策略和动作函数神经网络的参数，记为 $\mu_{\theta targ}$ 和 $Q_{\phi targ}$。

接下来重复进行如下活动来学习

1. 观察股票价格 S_t 采取行为 $u_t=clip(\mu_\theta(S_t)+\epsilon,-1,1)$，其中 ϵ 是随机噪音服从正态分布，目的是平衡算法的探索（Exploration）和利用（Exploitation）问题，在尝试多种动作的前提下找到最优动作。
2. 将采取动作后的样本 (S_t,u_t,R_t,S_{t+1}) 加入 $\mathcal{D}$ 中
3. 在 $\mathcal{D}$ 中随取抽取一批样本，$\mathcal{B}=(s,u,r,s')$
4. 利用梯度 G_ϕ 更新 Q_θ

$$G_\phi=\nabla_\theta\frac{1}{|\mathcal{B}|}\sum_{(s,u,r,s')\in\mathcal{B}}\left(Q_\phi(s,a)-r+\gamma Q_{\phi targ}(s',u_{\theta targ}(s'))\right)^2 \tag{36}$$

5. 利用梯度 G_θ 更新 μ_θ

$$G_\phi=\nabla_\theta\frac{1}{|\mathcal{B}|}\sum_{(s,u,r,s')\in\mathcal{B}}Q_\phi(s,\mu_\theta(s)) \tag{37}$$

6. 更新目标神经网络

$$\begin{aligned}\phi_{targ}&\leftarrow\rho\phi_{targ}+(1-\rho)\phi\\ \theta_{targ}&\leftarrow\rho\theta_{targ}+(1-\rho)\theta\end{aligned} \tag{38}$$

直至训练到策略和动作函数神经网络收敛，进而带入初始条件 S_0，计算出 0 时刻期权的价格和对冲比率。

6. 总结和展望

在本篇文章中，我们站在构建期权复制组合和加入风险溢价的角度出发，展示了两种计算期权价格的算法。

第一种方法是强化学习中Planning的概念，即已知了环境的转换概率，通过模拟股票价格路径来规划最优的对冲比率和期权价格。它的优势首先是在给定初始条件 (S_0,T,K) 后通过蒙特卡洛模拟出来的股票价格路径反向计算出期初的对冲比率和期权价格，学习速度快（只学习了一次）且结果稳定；其次由于对复制组合有限次数的调仓更加符合了现实世界中对期权价格的设定（不同对BSM模型的连续时间对冲，因此加入了对冲不及时的风险在期权价格中）；最后通过对超参数 λ，即Agent的风险厌恶程度进行调整可以去拟合市场上不同的期权类型和期权卖方。但是它的缺点也很明显，就是前提假设了股票价格分布模型，但是现实生活中并不能保证股票价格负冲假设的分布模型。总的来说Dynamic Programming算法是对原有BSM模型的一个拓展，解决了如何将风险中性测度下的期权价格延伸到现实世界中。

第二种方法是强化学习中Learning的概念，通过与未知环境的互动，获取样本来学习最优的行动策略。它的优势首先是摒弃了对股票价格分布模型的假设，可以使用真实市场数据作为学习的环境；其次DDPG算法保证了其学习过程的稳定性和结果的收敛性。但是它的缺点是神经网络需要大量的样本进行训练，而且对超参数 λ 敏感，很容易使最优行动，即对冲比率收敛到边界 ± 1，从而导致结果不正确。

两种方法在未来都有进一步改进的空间，如构建更加符合股票真实价格分布的模型作为蒙特卡洛模拟的分布模型；又如加入期权复制组合在每次换仓时候的交易成本，使其可以更好的符合现实金融市场中的环境；再如在复制组合价格未来波动的风险溢价以外添加新的风险溢价因子，如果市场新闻和政策情绪，股票价格过去动量等等。

在以上的研究中发现，决定强化学习问题的最关键的要素就是如何定义奖励函数，不同的奖励函数对应不同的期权价格以及对冲比率。最后接下来的研究方向是，利用已知的期权价格、隐含波动率和市场Delta等信息作为次最优信息（Suboptimal information），使用逆强化学习（Inverse reinforcement learning）学习市场隐含的真实的奖励函数和最优行动策略，利用对抗神经网络（GAN）与逆强化学习的相似之处构建算法，得到最后的模型用于真实的期权市场定价。