Assignment 1: Q-learning for Discrete-time Asset Allocation

Zhu Zheng, 20916267, zheng.zhu@connect.ust.hk

He Xinyi, 20914738, xhebm@connect.ust.hk

(GitHub: https://github.com/ZHUZheng1999/Reinforcement-learning-in-financial-applications)

1. Introduction

Asset allocation is a vital process for investors to determine the amount apportioned in risk-free and various risky asset classes. This assignment deals with the discrete-time asset allocation between risk-free asset and 1 unit of risky stock, and the stock return is supposed to follow binomial distribution. Q-learning is used in this assignment to find the optimal strategy in each single-time-step for investors.

2. Problem Formulation

The initial wealth is set as W_0 at $t = 0$. At each of discrete time steps labeled $t = 0, 1, \dots, T-1$, there are two actions A_t: buy 1 unit of underlying stock while investing the remaining in riskless asset ($A_t = 1$) and only invest in risk-free asset ($A_t = 0$). Assumes the stock yields a random binomial return Y_t at u with probability p and d with probability $1-p$ in each time step, and the riskless asset would get return r at each time unit. Also, transaction costs are not taken into consideration. Then, the wealth at time $t+1$ is calculated based on action and wealth at time t as following:

$$W_{t+1} = A_t P_t(1+Y_t) + (W_t - A_t P_t)(1+r) = W_t(1+r) + A_t P_t(Y_t - r)$$

where P_t is the price of stock at time t.

The aim of the example in section 8.4 of Rao and Jelvis is to find the strategy that maximizes the expected utility of wealth at the final time step, that is W_T, and the utility of wealth is calculated by the following CARA function:

$$U(W_T) = \frac{1 - e^{-aW_T}}{a} \; for \; fixed \; a \neq 0$$

However, this assignment focuses more on the return at each time step to simulate the consideration of investors, which is calculated by the following formula. Therefore, the optimal strategy should be the one that maximizes the expected return of investors at each step.

$$R_{t+1} = \frac{W_{t+1}}{W_T} - 1$$

For the Q-learning model, this assignment sets time as state, and action A_t is generated using greedy algorithm at each state. The reward is given at each time step t as mentioned above. The initial Q-value of last time is initialized as 0 with the assumption that investors would just keep the wealth on hand without any investment after time T, so that they would get zero return. The Q-learning algorithm is based on following formula with learning rate α and discount factor γ:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma * max_{A_{t+1}} Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)]$$

3. Methodology

This assignment defines initial wealth $W_0 = 10$, initial stock price $P_0 = 1$, the probability of stock price increasing $p = 0.5$, rising stock return $u = 0.1$ and falling stock return $d = -0.1$, while the riskless rate $r = 0.05$. The mathematical reasoning of this problem is first formulated to provide a theoretical basis, and then this assignment writes code in Python to implement Q-learning algorithm.

3.1 Mathematical Implementation

3.1.1 Solve the initial problem in textbook

First, according to the discrete-time asset allocation example in section 8.4 of Rao and Jelvis, the value function of the state is defined as:

$$V_t^{\pi}(W_t) = E_{\pi}[\frac{-e^{-\alpha W_T}}{\alpha} | (t, W_t)]$$

Therefore, the optimal value function is:

$$V_t^*(W_t) = max_\pi E_\pi[\frac{-e^{-\alpha W_T}}{\alpha}|(t, W_t)]$$

And,

$$V_t^*(W_t) = max_{x_t} Q_t^*(W_t, X_t)$$

Since the Y_t follows the binomial distribution, the optimal function is set as following:

$$V_t^*(W_t) = max_{x_t} E_{Y_t \sim B(a,b,p)}[V_{t+1}^*(W_{t+1})] \; for \; t = 0,1, \dots T-2$$

Therefore,

$$V_{T-1}^*(W_{T-1}) = max_{x_{T-1}} Q_{T-1}^*(W_{T-1}, X_{T-1}) = max_{x_{T-1}}(E_{Y_{T-1} \sim B(a,b,p)}[\frac{-e^{-\alpha W_T}}{\alpha}])$$

This assignment assumes $T = 10$, so the optimal value function at $t = 9$ is:

$$V_9^*(W_9) = max_{x_9}\left[(p-1)e^{-\alpha X_9(b-r)} - pe^{-\alpha X_9(a-r)}\right] * \frac{-e^{-\alpha W_9(1+r)}}{\alpha}$$

by defining $a = 1 + u$ and $b = 1 + d$.

Let

$$f(x) = (p-1)e^{-\alpha x(b-r)} - pe^{-\alpha x(a-r)},$$

then

$$f'(x) = \alpha(1-p)(b-r)e^{-\alpha x(b-r)} + \alpha p(a-r)e^{-\alpha x(a-r)}$$

Set

$$f'(x) = 0$$

then

$$x = \frac{ln\frac{p(a-r)}{(1-p)(b-r)}}{\alpha(a-b)} \text{ to maximize } f(x).$$

So

$$V_9^*(W_9) = \{(p-1)[\frac{p(a-r)}{(1-p)(b-r)}]^{-\frac{b-r}{a-b}} - p[\frac{p(a-r)}{(1-p)(b-r)}]^{-\frac{a-r}{a-b}}\} * \frac{-e^{-\alpha W_9(1+r)}}{\alpha}$$

Since

$$V_8^*(W_8) = max_{x_8} E[V_9^*(W_9)], \; W_9 = x_8(Y_8 - r) + W_8(1+r),$$

we have

$$V_8^*(W_8) = max_{x_8}[-e^{-\alpha[x_8(a-r)+W_8(1+r)](1+r)} * p - e^{-\alpha[x_8(b-r)+W_8(1+r)](1+r)} * (1-p)]$$

And for the same reason, $x_8 = \frac{ln\frac{p(a-r)}{(1-p)(b-r)}}{\alpha(a-b)}$, which means

$$V_8^*(W_8) = \{(p-1)[\frac{p(a-r)}{(1-p)(b-r)}]^{-\frac{b-r}{a-b}} - p[\frac{p(a-r)}{(1-p)(b-r)}]^{-\frac{a-r}{a-b}}\}^2 * \frac{-e^{-\alpha W_8(1+r)^2}}{\alpha}$$

And so on, we let $V = (p-1)[\frac{p(a-r)}{(1-p)(b-r)}]^{-\frac{b-r}{a-b}} - p[\frac{p(a-r)}{(1-p)(b-r)}]^{-\frac{a-r}{a-b}}$, so

$$V_0^*(W_0) = V^{10} * \frac{-e^{-\alpha W_0(1+r)^{10}}}{\alpha}$$

And the optimal policy is:

$$\pi^*(W_t) = [\frac{ln\frac{p(a-r)}{(1-p)(b-r)}}{\alpha(a-b)}, \frac{ln\frac{p(a-r)}{(1-p)(b-r)}}{\alpha(a-b)}, \frac{ln\frac{p(a-r)}{(1-p)(b-r)}}{\alpha(a-b)}, \ldots, \frac{ln\frac{p(a-r)}{(1-p)(b-r)}}{\alpha(a-b)}] \text{ if } \frac{p(a-r)}{(1-p)(b-r)} > 0$$

Under the assumption of this assignment, $\frac{p(a-r)}{(1-p)(b-r)} < 0$, so that the optimal strategy is

$$\pi^*(W_t) = [0,0,\ldots,0]$$

for decreasing differential coefficient.

3.1.2 Solve the problem under assumptions

Similar to the calculation in above part, we have

$$V_t^*(W_t) = max_{x_t} E_{Y_t \sim B(a,b,p)}[V_{t+1}^*(W_{t+1})] \; for \; t = 0,1,\ldots T-2$$

And

$$V_{T-1}^*(W_{T-1}) = max_{x_{T-1}} Q_{T-1}^*(W_{T-1}, X_{T-1}) = max_{x_{T-1}}(E_{Y_{T-1} \sim B(a,b,p)}[\frac{-e^{-\alpha W_T}}{\alpha}])$$

So

$$V_9^*(W_9) = max_{x_9}[\frac{(1-p)x_9(b-r) + px_9(a-r)}{W_9}] + r$$

Let

$$\mathrm{f(x)} = (1-p)x(b-r) + px(a-r)$$

Then

$$f'(x) = (1-p)(b-r) + p(a-r), where \; x \in [0, W_9]$$

Under our assumptions,

$$(1-p)(b-r) + p(a-r) = -0.05 < 0$$

So, we choose $x_9 = 0$.

And for the same reason, the optimal policy is

$$\pi^*(W_t) = [0,0,\ldots,0]$$

3.2 Coding Implementation

The coding program stores Q-value table in a two-dimension array with row index representing action and column index representing time periods. For Q-learning method, discount rate γ is defined as 1, since investors are supposed to be visionary who pay great attention on long-time return and learning rate α is defined as 0.5 for simplicity. In greedy algorithm, epsilon is initially set at $\varepsilon = 0.1$, and it decreases in each 1000 training epoch as

$\varepsilon = 0.8 * \varepsilon$. After training in Q-learning algorithm, a convergent Q-table is generated, and then the optimal strategy is given based on the larger q-value of two actions at each time step.

4 Results

This assignment sets $T = 10$ as an example, and then gets the final Q-table and optimal strategy, checks the convergence of Q-values and tests the stability of Q-table.

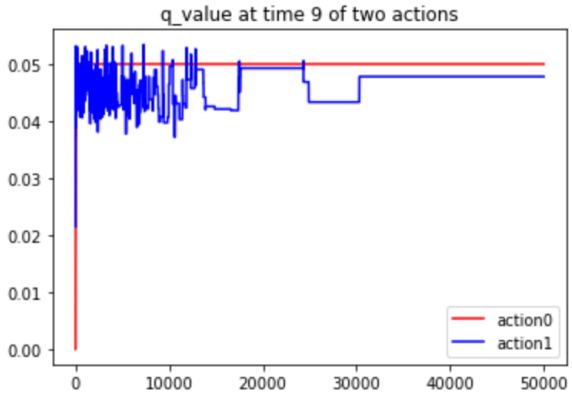
4.1 Q-table and optimal strategy

After training 50000 times for 10 period, the final Q-table is shown as follows. By choosing the action that has larger Q-value, the optimal strategy is given as taking action 0, that is investing all wealth into riskless asset, at every time step.

Time	Action 0	Action 1	Time	Action 0	Action 1
0	0.50	0.498309	**5**	0.25	0.243406
1	0.45	0.442390	**6**	0.20	0.196454
2	0.40	0.396887	**7**	0.15	0.143172
3	0.35	0.345483	**8**	0.10	0.094542
4	0.30	0.286650	**9**	0.05	0.047802

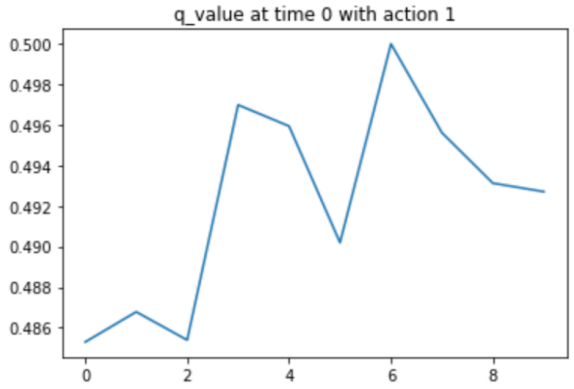
4.2 Check the convergence of Q-table

This assignment checks the convergence of Q-table by recording the Q-value at time $t = 9$ of two actions. From the following figure, the value of action 0 converges to 0.05 very quickly for constant riskless return, while the value of action 1 approaches to terminal value after around 30000 training times.



4.3 Test the stability of Q-value table

The train process is run 10 times to check the stability of Q-value at specific location. The following is the figure of q-value at time 0 with action 1 as an example. By calculating, the mean of the specific q-value is 0.4922 and standard variance is 0.0049, which shows the stability of Q-table.



5 Conclusion

By comparing the results of mathematical reasoning with the results of Q-learning programing implementation, they conclude the consistent optimal strategies. Therefore, under the assumptions of this assignment, investors are suggested to only invest in risk-free assets. Since the expectation return of the risky asset is 0 under our assumption, it's lower than the return of the riskless asset. So intuitively, we should allocate all our money to the risk-free asset. And the results from our program are consistent with our intuition.

Acknowledgement

This group assignment is completed by HE Xinyi and ZHU Zheng with equal contribution. We had lots of face-to-face discussions to design the algorithm and the experiment. HE focused more on the design of the algorithm and ZHU focused more on the experiment. In addition, HE finished the visualization of the results and ZHU finished the mathematical implementation. What's more, this report is also written by us together. Also, we would like to express our sincere appreciation to Professor Chak WONG and TA because of their patient help.