基于强化学习的投资组合管理

方圆，高云开，赵鹏飞

2023 年 1 月 14 日

1 背景介绍

人工智能的火热推动着机器学习算法的广泛应用，在金融领域，机器学习算法在 α 因子的创建与聚合，提升交易策略自动化执行效率等领域有突出贡献。较投资者自己进行交易而言，基于机器学习算法的量化投资凭借其模型化的交易方式能够同时发掘并处理更多信息，利用更多的盈利机会。

强化学习作为机器学习的一个分支在量化金融领域也有相应的应用，如资产组合管理、交易执行、高频的单资产交易以及期权与对冲。股票交易的过程包含市场环境、买卖动作与回报率等多种元素，与强化学习的主要概念相契合并很容易通过强化学习的框架进行阐述，同时，强化学习在量化金融中的应用也属较前沿的部分，富有极大的挑战性。在本文中，我们不仅采用 DDPG、PPO、SAC 等不同模型进行交易并对比，还将基于 PPO 模型作出一些创新尝试。同时，不同于传统模型中奖励设置和训练过程，我们尝试调整奖励函数，加入回撤率作为惩罚，令模型兼顾收益和风险；并采用随机时点开始的训练方式，使模型能适应不同环境，减少对入市时机的路径依赖并使其更具有稳定性，最终取得了较好的结果。

2 现有方法综述

将 DRL 应用于投资组合管理的方法主要有两种[1]，1）训练策略对单个资产进行交易，该资产可以包括各自资产的多头和空头头寸；2）培训策略对资产组合进行分配，学习最佳加权策略。其次，受投资公司下不同投资经理风险分散作用的启发，Multi-Agent 也被应用于投资组合管理中[2]，除此之外，针对训练集与测试集中数据分布不同的问题，

[1]High-dimensional stock portfolio trading with deep reinforcement learning

[2]Lee, J., Kim, R., Yi, S. W., Kang, J. (2020). MAPS: multi-agent reinforcement learning-based portfolio management system. arXiv preprint arXiv:2007.05402.

也可采用对抗学习[3]以及采样[4]，达到更好的泛化效能和鲁棒性。

3 数据集

我们使用的数据集是从 Tushare 上爬取的上证 50 日频数据，长度为 2009 年 1 月 1 日-2020 年 12 月 31 日，选取 2020 年 12 月 31 日的成分股作为数据集的成分股。

1. 训练期：2009 年-2018 年，测试期：2019 年-2020 年
2. 缺失数据集处理：对当日无法交易，未上市，已退市的股票，我们做了收盘价置 0 处理，在 Env 交易规则中不允许这部分股票交易。
3. 上证 50 特点：上证 50 基本是超大股，特点是流动性比较低，股价也比较稳定。虽然相比 CSI500 样本比较局限，但作为实验性质的数据集，上证 50 股价的稳定性也有利于模型的训练和分析。
4. 因子选定：高开低收 + 技术面因子

量价指标	技术指标
Close, High, Low, Open	"boll_ub", "boll_lb", "rsi_20", "close_20_sma", "close_60_sma", "close_120_sma", "macd", "volume_20_sma", "volume_60_sma", "volume_120_sma"

图 1: 因子示意图

4 Market Env 介绍

4.1 市场环境简介

market envrioment 继承于 gym 库中的 Env 类，并对市场环境进行了简化。状态空间以及动作空间采取了 gym 的 Box 类连续空间，大小分别为 $1^5 + s^6 + s * f^7$ 以及 $s+1$，市场环境主要函数有 reset、step。还有部分辅助函数例如奖励函数、终止函数、获取当日截面数据函数、计算实际交易股数函数以及用于保存账户信息以及状态监测的函数。

[3]Liang, Z., Chen, H., Zhu, J., Jiang, K., Li, Y. (2018). Adversarial deep reinforcement learning in portfolio management. arXiv preprint arXiv:1808.09940.

[4]Miao, Y. H., Hsiao, Y. T., Huang, S. H. (2020, December). Portfolio Management based on Deep Reinforcement Learning with Adaptive Sampling. In 2020 International Conference on Pervasive Artificial Intelligence (ICPAI) (pp. 130-133). IEEE.

[5]1 代表当前持有现金量

[6]s 代表股票数量，此处代表某时间各股票的持仓量

[7]f 代表因子数量

4.2 市场环境参数设置

市场环境主要参数有初始资金量、买卖手续费、最大可交易数量、是否随机开始以及是否无杠杆操作。其中初始资金量设定为 100 万元，买卖手续费用均为千分之三，最大可交易数量设定为 10，意味着在时间截面上购买或卖出某只股票的数量不超过 10，如此设定一方面是防止 Agent 直接在某一时刻将当前现金量全部注入到某只股票从而影响之后策略的学习，另一方面则出于投资组合管理的目的，将全部资金注入某只股票无益于通过分散化来降低系统性风险。在进行训练时随机开始为 true，反之为 false。训练时随机选择某一交易日至最后一期交易日的训练数据开始训练，如此可以减轻因不同交易日市场行情不同而产生的收益路径依赖，使得智能体能学习到更普适的投资组合交易管理策略。无杠杆操作默认为 true，意即当持有现金额低于支付费用时无法执行购买操作。同时设定不允许卖空操作。

```
df: 构建环境时所需要用到的行情数据
buy_cost_pct: 买股票时的手续费
sell_cost_pct: 卖股票时的手续费
date_col_name: 日期列的名称
hmax: 最大可交易的数量
print_verbosity: 打印的频率
initial_amount: 初始资金量
daily_information_cols: 构建状态时所考虑的列
cache_indicator_data: 是否把数据放到内存中
random_start: 是否随机位置开始交易
patient: 是否在资金不够时不执行交易操作
currency: 货币单位
```

4.3 reset 函数略解

reset 函数在 Agent 抵达 Terminal 后对环境进行重置。重置的内容包括动作列表、交易量列表、状态列表以及账户信息，并返回初始化的状态。

4.4 step 函数详解

step 函数用于 Agent 的状态转移，输入参数为动作列表，输出为 Agent 根据动作列表执行相应交易行为后的下一状态、奖励以及是否到达 terminal 的相应信息。

step 函数主要涉及奖励的计算以及交易量的计算。其中第一种奖励由以下公式给出:

$$reward_t = return_t + retreat_t \tag{1}$$

值得注意的是，模型中的奖励兼顾了收益率以及回撤率，回撤率作为收益率的惩罚项，而同样考虑收益与风险指标的还有 Sharpe ratio 以及 Omega ratio[8]，但在刚开始训练时波动比较小，Sharpe ratio 以及 Omega Ratio 会趋于无穷大，考虑即使在后期把这两项指标加入到奖励中会违反奖励的一致性原则，因此在奖励中只加入了回撤率来平衡收益率。

第二种奖励方式为单纯的收益率，预期采用第二种方式训练出的模型会有较高的收益，但同时风险也会有所上升。

交易数量是交易动作的线性映射，有如下的关系：

$$transction = action * hmax \tag{2}$$

这是因为在设定中，action 被设定为-1 到 1 之间的连续变量，同时当交易行为为卖出时，若交易量大于目前的持仓量，则会将交易量设置为目前持仓量。

整个 step 函数的流程如下：

Algorithm 1 step 流程图

Input: action

Output: $state_{next}, reward, terminal$

```
if 下一状态为 terminal then
    调用终止函数
else
    执行卖操作，计算交易费用以及卖出所得
    执行买操作
    if 现金不足 then
        无法买入，并将买信号修改为 0
    else
        计算交易费用以及买入支出
    end
end
计算奖励
更新现金持有量，更新持仓信息
```

[8] $$\omega_r = \frac{\int_r^{\infty} 1 - F(x)dx}{\int_{-\infty}^{r} F(x)dx}$$

其中 r 为指定的临界分布函数，F(x) 为收益率的累计分布函数

4.5 其余函数略解

其余函数的主要用于训练时的监测以及信息的存储。例如每隔 10 次便会输出当前 episode、step、持有现金量、状态终结原因、资产价值、奖励、回撤率、收益率等信息以及在训练完成后会以 dataframe 的形式保存交易行为信息与账户信息。

4.6 市场环境总结

在市场环境中加入了两个较为创新的点：1）将回撤率作为收益率的惩罚性加入到奖励判定中。2）采用了随机开始的设定来减弱入市时机的路径依赖问题。但也存在许多不足之处，例如可以通过加入根据量化因子判断的交易准则评判交易行为，并加入到最终的奖励：例如认为收盘价向上突破布林带上界时卖出为一项不错的交易行为，给予正奖励，但出于对市场的不确定性，这样的评价标准太过单一以及很可能并不正确的考虑，并没有加入到目前的模型之中。还可以考虑更复杂的市场环境，例如允许杠杆化交易等等行为。除此之外，我们的市场环境更适用于日频交易，而当期限较长时，会发生智能体摆烂的现象。

5 Deep Reinforcement Learning 模型

在前文构建了市场交易环境 Marketgym 后，我们尝试了使用开源库（1）**Stable_baselines**,尝试简单的集成模型(2)**Ensemble** 以及自己搭建的模型(3)**PPO_team** 和（4）**PPO_Attention**，前者是为了比较和展现主流 DRL 算法的表现，分析各个 DRL 模型在投资环境的风格, 后者是为了对 DRL 模型做出一些想法的改进与拓展, 试验自己的想法。

5.1 OpenAI 模型

OpenAI 是 AI 领域的非常有名且有实力的公司，最近大火的 ChatGPT 就是他们的研究成果。而 **Stable Baselines** 是 OpenAI 的开源强化模型库，里面包括了强化学习领域的主流算法的实现与 doc 说明, 以及环境的封装函数。

Stablesline3 开源网址：https://stable-baselines3.readthedocs.io/en/master/

我们调用了有连续型动作的有 **A2C，DDPG，PPO，SAC, TD3** 五个模型。其中 DDPG 为 DQN 的连续型动作拓展，PPO 是 RL 领域使用最多，泛用性最好的模型，A2C，SAC，TD3 是 RL 领域也非常具有代表性的新兴算法（SOTA）。Stable_baseline3 提供了 Fully_Connected 和 Convolutional 网络两种选择，我们选择的是基本的 FC 网络作为对比对象。训练参数上选择的 Stable_baselines3 的默认参数，训练长度分别为 50000，100000，150000，200000 步（50000 步约 25 次 2009 年-2019 年的训练）。

此外，我们尝试了简单的 Ensemble，对五个 RL 算法的增减进行均值加权: 对每只标的，Ensemble 按时间截面取单个模型输出的均值 mean(A2C，DDPG，PPO，SAC，TD3)，比较其与单个模型的效果。

5.2 PPO 手动搭建

在训练 OpenAI 库中，我们发现 PPO 模型表现与在论文中高泛用性相差较大，回测表现较差（见回测分析部分），所以决定手工实现并进行调整，并分析原因在什么地方。

我们实现的 PPO 与常规的 PPO 主体没有变化，采取了（1）Actor，critic 网络（2）Clip Value （3）Batch memory 更新等设计。

值得一提是连续动作的改进上，在 Market_env 中的模型要考虑连续动作问题，其实施细节也很简单，在于动作网络输出一个动作的概率分布，然后压到 [-1,1] 中，这个概率分布可以是有充分统计量的分布（即分布取决于参数），如高斯分布，Beta 分布等。查阅了 Stable_Baseline3 的 doc 后，选取了和其一致的 Beta 分布。

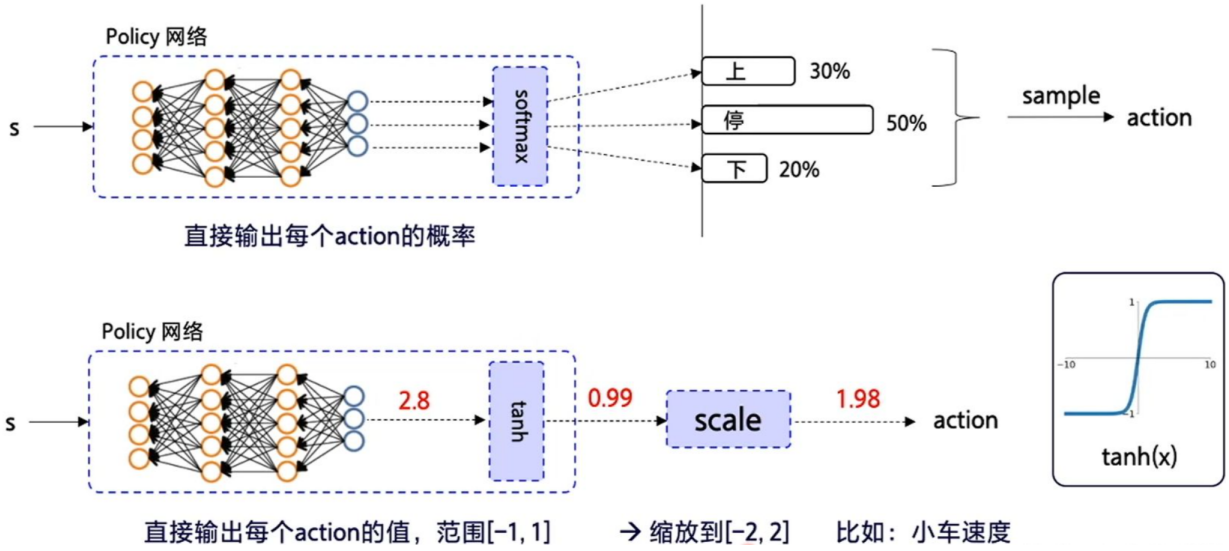

图 2: 连续动作示意图

在回测比较以后发现，同样训练 50000 步的 PPO 手动搭建版比 StableBaseline3 版本表现好很多。在对比参数后，发现应该是原版 PPO 的 learningrate 太低导致，相比其他库中模型默认 1e-3，PPO 只有 5e-5（可能是因为 PPO 学习较快导致设计者把这个参数调的比较低），但在我们的金融任务下，它学习的就过慢了。

5.3 PPO 与 Attention 拓展与尝试

Attention 机制是 Google 在 2017 年的 Attention is All your need [1] 中提出的一种网络设计层。Attention 通过计算 Q，K，V 三个矩阵可以得出输入向量各自的相关重要性，在原文中用来表示句子中各个词向量与剩余词向量的相关性。

而在本次 Project 中，我们的环境状态也是一个 801 维的向量，用来表示 Agent 的持仓 + 市场信息，加入 Attention 层直觉上可以计算出各支股票的相关性，更好地调仓。基于这样的想法，我们在 PPO 的 Critic 部分加入 Attention 层的改进。

具体实现上是基于 torch.nn.MultiheadAttention 网络层，向量 Batch 形状为 Batch * 1 * State_length。在三层 FC 后面各加入 Attention 层，做一个简单的尝试。

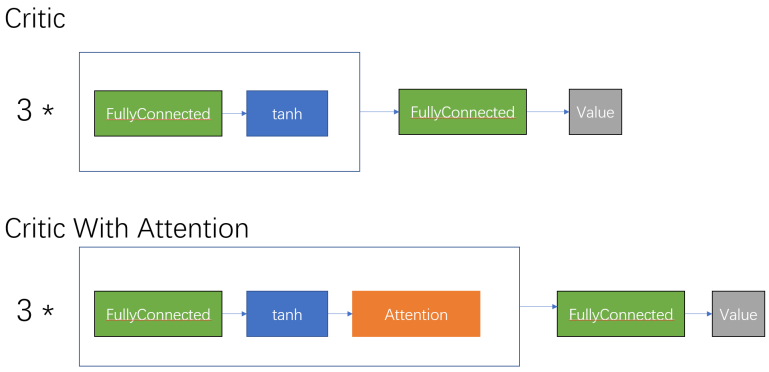

图 3: Critic 网络改进

5.4 模型的反思与思考

1. **环境的高方差性无法解决**

强化学习的高方差性一直以来都是个很难解决的问题，在金融环境中的强化学习决策问标更加剧了这个问题。资产组合的调仓受金融市场本身噪声影响，规律并不明显，在这段时间表现好的并不意味着在下一段时间比较会更好；其次是我们设计的基于强化学习的资产组合调整会有路径依赖问题，昨日的持仓会决定今天的增减仓，使得每次的决策路径都相差较大。这让模型的对比和分析很难像游戏那样有个显著的好坏，

2. **Attention 尝试的可完善性**

Attention 层的尝试是一个不成熟的尝试。更完整的想法应该是将数据输入改为 Stock_num * 指标数，对照 NLP 中的词向量 * 位置向量，而不是简单的 801 维向量。(1*801 维的向量其实只是加入了自己计算与自己的相关性，本质上没有体现出来 attention 的相关性比较)。这样才能将不同股票的相关性如同句子中的不同词的相关性一样提取出来。但是因为时间有限，这个改进需要数据集和模型的改写，也超出了本门课的课程范围，就没有操作，希望以后有时间可以更深入地探究。

6 回测分析

将 StableBaselines 五个模型分别在训练期训练 5、10、15、20 万步后，置入测试期进行回测分析。在不同的训练周期下，模型表现的结果有许多差异，又几个在两年的测试期内将总资产翻了 2-3 倍，而另几个模型则只得到了和指数相似甚至较低的收益。详细资产变化请见附录。

从资产价值走势我们可以看出，成绩较为瞩目的是 SAC 模型。对比没有随机策略的 A2C 和 SAC 的业绩，可能由于 SAC 策略的随机化与股市的变动风格更加契合的原因。PPO 与前期的预测表现相反，在回测中的成绩则只差强人意，分析其原因，可能是因为 PPO 目标函数中有信任区域的惩罚项，因而在市场回报较高、波动较剧烈的测试期内并不能及时顺应市场改变策略。我们推测，PPO 可能在熊市或无太大波动的市场中有更好的表现。此外，在简单的 Ensemble 下，模型的表现也十分优异，在充分训练后，能战胜单个模型，体现出 Ensemble 方法的简单有效。

在个人尝试的 PPO (stable—baselines) 和 PPO_team,PPO_attention 可以看出我们自己搭建的 PPO 模型比起 stablebaslines 表现显著更为优秀，分析原因后是学习率参数的差别。而对比是否加入 attention 网络层，发现表现差别不大，这是因为 attention 层的设计仍较为简单，需要进一步的设计。

7 总结

本次项目的亮眼之处有以下几点：1）对奖励加入了回撤率的惩罚使其兼具收益并考虑风险。2）训练时随机开始以消除入市时机不同而导致的路径依赖问题，使模型更具稳定性。3）除利用 OpenAI 的开源算法以外，本小组也着重在 PPO 算法上进行了自主化的编写训练。4）尝试在 PPO 算法中加入较为先进的 Attention 机制。

同时，本次项目中也有很多构想没有实现以及存在较多问题：1）在进行大批量的数据投入时可以通过 LTSM 网络来对因子进行进一步的聚合这一想法没有实施。2）在交易规则中加入常见的技术分析手段并作为奖励的一部分：如收盘价是否高于 MACD，同时也可加入良好交易行为的判定并自定义奖励衰减倍数。3）Attention 层的改进仍待完善。在完成了基本化的算法构建与交易回测基础上，本小组进行了有益的探索。从回测结果上来看，本次项目取得了颇为不错的结果，同时对悬而未决的问题以及探索化的想法，我们也期待能在将来进行深入化的研究。

A 附录

A.1 回测分析：资产变化

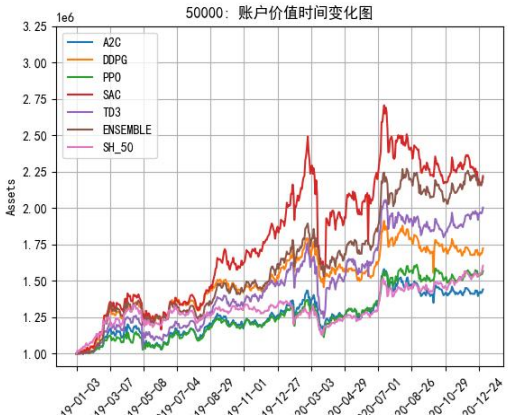

图 4: 50000 步资产变化

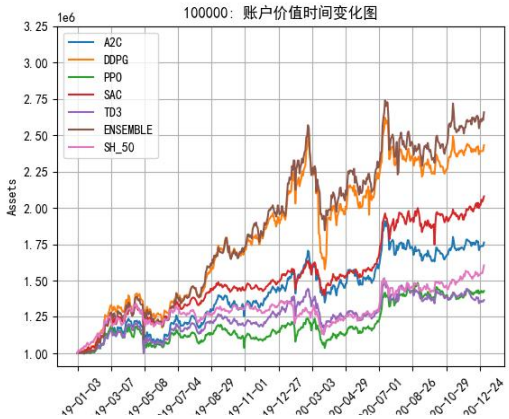

图 5: 100000 步资产变化

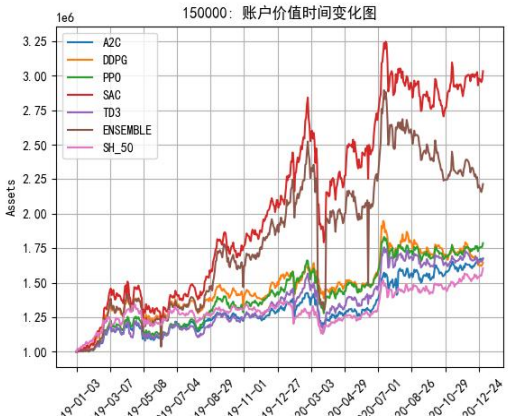

图 6: 150000 步资产变化

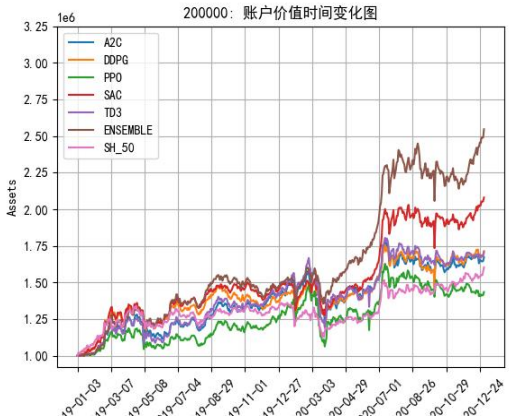

图 7: 200000 步资产变化

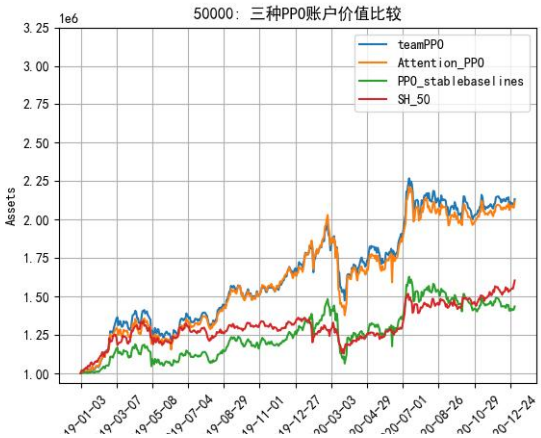

图 8: PPO 模型改进前后对比

A.2 回测分析：量化指标

	A2C	DDPG	PPO	SAC	TD3	Ensemble	SH50
Annual return	0.3120	0.6543	0.3547	0.4959	0.3346	0.4828	0.2778
Cumulative returns	0.6882	1.6401	0.7959	1.1744	0.7447	1.1377	0.6043
Annual volatility	0.2240	0.3442	0.2290	0.2694	0.2830	0.4031	0.2070
Sharpe ratio	1.3280	1.6394	1.4437	1.6338	1.1649	1.1777	1.2904
Calmar ratio	1.5053	2.0537	2.1459	2.5682	1.2510	1.4931	1.6161
Stability	0.9225	0.9125	0.8816	0.8818	0.8923	0.9224	0.6152
Max drawdown	-0.2073	-0.3186	-0.1653	-0.1931	-0.2674	-0.3234	-0.1719
Omega ratio	1.2636	1.3622	1.2966	1.3523	1.2620	1.3131	1.2630
Sortino ratio	1.8813	2.4392	2.1120	2.4461	1.6645	1.8242	1.9457
Tail ratio	1.2391	1.2279	1.3019	1.2645	1.2419	1.1757	1.2952
Daily value at risk	-0.0270	-0.0411	-0.0275	-0.0322	-0.0343	-0.0489	-0.0250

表 1: 50000 步量化指标

	A2C	DDPG	PPO	SAC	TD3	Ensemble	SH50
Annual return	0.1984	0.6695	0.2636	0.5642	0.2823	0.8381	0.2778
Cumulative returns	0.4177	1.6871	0.5703	1.3696	0.6154	2.2349	0.6043
Annual volatility	0.2166	0.3180	0.2577	0.2841	0.2416	0.3446	0.2070
Sharpe ratio	0.9460	1.7765	1.0394	1.7223	1.1528	1.9456	1.2904
Calmar ratio	1.2016	2.7309	1.2156	3.1642	1.1712	3.1518	1.6161
Stability	0.7676	0.9512	0.7974	0.9254	0.8866	0.9617	0.6152
Max drawdown	-0.1651	-0.2452	-0.2169	-0.1783	-0.2411	-0.2659	-0.1719
Omega ratio	1.1885	1.3977	1.2160	1.3855	1.2281	1.4526	1.2630
Sortino ratio	1.3426	2.6238	1.5127	2.5324	1.6729	2.8907	1.9457
Tail ratio	1.1411	1.2213	1.3457	1.3192	1.1241	1.2280	1.2952
Daily value at risk	-0.0265	-0.0378	-0.0314	-0.0339	-0.0293	-0.0408	-0.0250

表 2: 100000 步量化指标

	A2C	DDPG	PPO	SAC	TD3	Ensemble	SH50
Annual return	0.2686	0.5248	0.2219	0.5049	0.2606	0.3971	0.2778
Cumulative returns	0.5823	1.2560	0.4718	1.1997	0.5631	0.9058	0.6043
Annual volatility	0.3750	0.2844	0.2191	0.2676	0.2368	0.2589	0.2070
Sharpe ratio	0.8245	1.6299	1.0268	1.6666	1.0990	1.4251	1.2904
Calmar ratio	0.7492	2.3940	1.2486	2.6101	1.3960	2.2805	1.6161
Stability	0.8057	0.9059	0.7868	0.9418	0.7491	0.9018	0.6152
Max drawdown	-0.3585	-0.2192	-0.1777	-0.1935	-0.1867	-0.1741	-0.1719
Omega ratio	1.2083	1.3411	1.2034	1.3584	1.2191	1.2854	1.2630
Sortino ratio	1.1960	2.4334	1.4777	2.4643	1.5964	2.0688	1.9457
Tail ratio	1.2513	1.2798	1.2067	1.2257	1.1845	1.2669	1.2952
Daily value at risk	-0.0460	-0.0340	-0.0267	-0.0319	-0.0288	-0.0312	-0.0250

表 3: 150000 步量化指标

	A2C	DDPG	PPO	SAC	TD3	Ensemble	SH50
Annual return	0.2686	0.5606	0.3374	0.6605	0.4583	0.4912	0.2778
Cumulative returns	0.5823	1.3592	0.7519	1.6591	1.0700	1.1610	0.6043
Annual volatility	0.2207	0.2905	0.2430	0.3525	0.3915	0.2840	0.2070
Sharpe ratio	1.1913	1.6830	1.3220	1.6200	1.1620	1.5539	1.2904
Calmar ratio	1.5637	2.7913	1.7252	1.9982	1.1875	1.9647	1.6161
Stability	0.8682	0.9276	0.8914	0.9135	0.8881	0.9203	0.6152
Max drawdown	-0.1718	-0.2008	-0.1956	-0.3305	-0.3859	-0.2500	-0.1719
Omega ratio	1.2394	1.3658	1.2763	1.3732	1.2818	1.3294	1.2630
Sortino ratio	1.7252	2.4460	1.8863	2.4068	1.7302	2.2593	1.9457
Tail ratio	1.2168	1.3096	1.2670	1.2591	1.3764	1.2281	1.2952
Daily value at risk	-0.0268	-0.0347	-0.0293	-0.0421	-0.0475	-0.0340	-0.0250

表 4: 200000 步量化指标

	PPO	team PPO	attention PPO	SH50
Annual return	0.3547	0.4812	0.4755	0.2778
Cumulative returns	0.7959	1.1333	1.1176	0.6043
Annual volatility	0.2290	0.2819	0.3149	0.2070
Sharpe ratio	1.4437	1.5393	1.3970	1.2904
Calmar ratio	2.1459	1.9124	1.4799	1.6161
Stability	0.8816	0.9078	0.9009	0.6152
Max drawdown	-0.1653	-0.2516	-0.3213	-0.1719
Omega ratio	1.2966	1.3218	1.3124	1.2630
Sortino ratio	2.1120	2.2554	2.0401	1.9457
Tail ratio	1.3019	1.4177	1.3968	1.2952
Daily value at risk	-0.0275	-0.0338	-0.0379	-0.0250

表 5: 不同 PPO 量化指标对比

A.3 参考文献

参考文献

[1] Vaswani, Ashish, et al. ”Attention is all you need.” Advances in neural information processing systems 30 (2017).

[2] Wang, Jingyuan, et al. ”Alphastock: A buying-winners-and-selling-losers investment strategy using interpretable deep reinforcement attention networks.” Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery data mining. 2019.

[3] Liu, Xiao-Yang, et al. ”FinRL: A deep reinforcement learning library for automated stock trading in quantitative finance.” arXiv preprint arXiv:2011.09607 (2020).

[4] Lee, J., Kim, R., Yi, S. W., Kang, J. (2020). MAPS: multi-agent reinforcement learning-based portfolio management system. arXiv preprint arXiv:2007.05402.

[5] Liang, Z., Chen, H., Zhu, J., Jiang, K., Li, Y. (2018). Adversarial deep reinforcement learning in portfolio management. arXiv preprint arXiv:1808.09940.

[6] Miao, Y. H., Hsiao, Y. T., Huang, S. H. (2020, December). Portfolio Management based on Deep Reinforcement Learning with Adaptive Sampling. In 2020 International Conference on Pervasive Artificial Intelligence (ICPAI) (pp. 130-133). IEEE.,