基于深度强化学习的金融投资组合管理问题

杨可芸 12012438 **潘腾** 12012825 **陈驿来** 12013025

指导老师：杨鹏

March 2023

目录

摘要

金融投资组合管理是将基金不断重新分配到不同的金融产品中的过程。我们在这个背景下提出一种无需金融模型的强化学习框架，以提供投资组合管理问题的深度机器学习解决方案。该框架包括独立同分布评估器 Ensemble of Identical Independent Evaluators(EIIE) 拓扑结构、投资组合向量存储器 Portfolio-Vector Memory(PVM)、在线随机批量学习 Online Stochastic Batch Learning(OSBL) 方案以及明确的奖励函数。本文在加密货币市场中的三个时间点上实现了这个框架，分别使用卷积神经网络（CNN）、基本循环神经网络（RNN）和长短期记忆（LSTM）。在三个回测实验中，交易周期为 30 分钟，在最近审查或发布的投资组合选择策略的比较下，所有三个框架实例都占据前三位，并超过其他交易算法。尽管回测中佣金率较高，为 0.25%，但该框架能够在 50 天内实现至少 4 倍的回报。[13]

关键词

机器学习，卷积神经网络，长短时记忆网络，强化学习，深度学习，投资组合管理，量化金融

1 问题背景

这篇论文的问题背景是金融投资组合管理。在金融领域中，投资组合管理是一项关键的任务，具体表现为不断重新分配金额的决策过程，将资金投入多种不同的金融投资产品，旨在最小化风险的前提下，最大化投资组合的回报。投资者通常需要将资金投资于多种不同的资产类别，例如股票、债券、商品等，以实现最佳的投资组合。然而，由于市场的不确定性和复杂性，投资者通常难以准确预测市场走势，从而很难实现最佳的投资组合。此外，投资组合管理通常需要考虑多种因素，如资产收益、波动率、流动性、交易费率等，因此也是一项具有挑战性的任务。

为了解决这些问题，近年来越来越多的研究者开始探索使用机器学习和深度学习方法来实现投资组合管理。这些方法通常可以让智能代理从历史市场数据中学习到最优的投资策略，并且可以快速适应市场的变化。然而，由于投资组合管理问题的复杂性，这些方法仍然存在许多挑战和困难，需要更深入的研究和探索。本文提出的深度强化学习框架就是为了应对这些挑战而提出的，旨在提供一种新的机器学习解决方案来优化金融投资组合管理。

2 问题描述

投资组合管理是将资本不断重新分配到许多金融资产中的行为。对于自动交易机器人来说，这些投资决策和行动是周期性的。在这个部分，我们首先从数学的角度去描述，定义这个问题。

2.1 交易周期

在这项工作中，交易算法是随着时间运行的，其中时间被划分为相等长度的周期 T。在每个周期开始时，交易代理在资产之间重新分配资金。在本次进行的所有实验中 $T = 30$ 分钟。各个资产的价格会在一段时间内涨跌，我们使用四个重要的价格特征点去描绘该资产在一段时间内的整体走势，即开盘价、最高价、最低价和收盘价（Rogers 和 Satchell，1991）。在连续的市场交易情景下，金融产品在一段时间内的开盘价是前一段时间的收盘价。在回测实验中，我们就假设在每个时期开始时，资产可以按照该时期的开盘价进行买卖。

2.2 数学描述

首先，投资组合由 m 项资产组成。对于第 t 个周期的收盘价，定义为价格向量 $\boldsymbol{v}_t$，也就是说，第 i 个金融产品在 t 时刻的收盘价格为 $v_{i,t}$。相似的，我们定义 $\boldsymbol{v}_t^{(hi)}$ 和 $\boldsymbol{v}_t^{(lo)}$ 分别为在 t 时刻内出现的最高价和最低价。对于连续市场，$\boldsymbol{v}_t$ 的元素是时段 $t+1$ 的开盘价和时段 t 的收盘价。第 t 个交易时段的价格相对向量 $\boldsymbol{y}_t$ 定义为 $\boldsymbol{v}_t$ 除以 $\boldsymbol{v}_{t-1}$ 的元素

$$\boldsymbol{y}_t = \boldsymbol{v}_t \oslash \boldsymbol{v}_{t-1} = (1, \frac{v_{1,t}}{v_{1,t-1}}, \frac{v_{2,t}}{v_{2,t-1}}, ..., \frac{v_{m,t}}{v_{m,t-1}})^T \quad (1)$$

另外，相似地，定义 $\boldsymbol{w}_t$ 为投资组合权重向量，则可以认为 $\boldsymbol{w}_t$ 的元素之和总为 1。$\boldsymbol{w}_{i,t-1}$ 是资产 i 在 t 时刻开始时所占的比例

$$\boldsymbol{w}_0 = (1, 0, ..., 0)^T \quad (2)$$

$$\boldsymbol{w}_t = (w_{1,t}, w_{2,t}, ..., w_{m,t})^T \quad (3)$$

定义 p_t 为 t 时刻的投资组合价值，那么相邻时刻的 p_t 可以按照如下公式表示：

$$p_t = p_{t-1}\boldsymbol{y_t} \cdot \boldsymbol{w_{t-1}} \quad (4)$$

定义 ρ_t 为回报率，t 时刻的回报率为结束时的投资组合价值相对于开始时的投资组合价值的增长率

$$\rho_t = \frac{p_t}{p_{t-1}} - 1 = \boldsymbol{y}_t \cdot \boldsymbol{w}_{t-1} - 1 \quad (5)$$

对其取对数获得对数回报率 r_t：

$$r_t = \ln \frac{p_t}{p_{t-1}} = \ln \boldsymbol{y}_t \cdot \boldsymbol{w}_{t-1} \tag{6}$$

在我们拥有以上数学变量的基础上，我们可以认为在没有交易费率的情况下，本项目的目标可以表述为尽可能获得最大的最终投资组合价值 p_f

$$p_f = p_0 \exp \sum_{t=1}^{t_f+1} r_t = p_0 \prod_{t=1}^{t_f+1} \boldsymbol{y_t} \cdot \boldsymbol{w}_{t-1} \tag{7}$$

2.3 交易费率/佣金

在真实的市场环境中，交易并不是没有成本的，每次交易涉及到要支付一定比例的佣金，那么这里假设佣金费率不变的情况下，需要重新计算每次调整资产占比之后的投资组合价值。

在第 t 个周期结束时，投资组合经理需要将 w'_t 更新为 w_t,(重新分配资金比例)，这个过程中将要支付佣金，假设重新分配使得投资组合的价值缩小了 μ_t, $\mu_t \in (0, 1]$。因此在调整资金占比前后，有这样的投资组合价值的改变:

$$p_t = \mu_t p'_t \tag{8}$$

整个交易过程可以用第四页顶部的图片贴切地描述。时刻 t 期间的市场波动，以价格相对向量 $\boldsymbol{y}_t$ 表示，推动投资组合价值和投资组合权重从 p_{t-1} 和 $\boldsymbol{w}_{t-1}$ 到 p_t 和 $\boldsymbol{w}_t$。时刻 t 的资产买卖行为将基金重新分配到 $\boldsymbol{w}t$。作为结果，这些交易将投资组合缩小到 p_t 的 μ_t 倍。

相对应的，上文提到的回报率等量也要做相应调整:

$$\rho_t = \frac{\mu t p'_t}{p_{t-1}} - 1 = \mu_t \boldsymbol{y}_t \cdot \boldsymbol{w}_{t-1} - 1 \tag{9}$$

$$r_t = \ln \frac{p_t}{p_{t-1}} = \ln \mu_t \boldsymbol{y}_t \cdot \boldsymbol{w}_{t-1} \tag{10}$$

最终的投资组合价值也应表述为:

$$p_f = p_0 \exp \sum_{t=1}^{t_f+1} r_t = p_0 \prod_{t=1}^{t_f+1} \mu_t \boldsymbol{y_t} \cdot \boldsymbol{w}_{t-1} \tag{11}$$

具体的，我们知道买入和卖出涉及到不同的佣金，我们假设卖出地佣金比率为 c_s, 那么在 $p'_t w'_{t,i} > p_t w_{t,i}$ 或者 $w'_{t,i} > \mu_t w_{t,i}$ 这样的需要卖出的情况下，我们通过卖出所有资产可以得到的资金为:

$$(1 - c_s) p'_t \sum_{i=1}^{m} (w'_{i,t} - \mu_t w_{i,t})^+ \tag{12}$$

随后我们再进行买入操作，假设买入的佣金比率为 c_p。将以上卖出获得的资金，加上原先位于 w_0 的没有启用的资金为所有资金，操作买入:

$$\begin{aligned}(1 - c_p)[w'_{t,0} + (1 - c_s) \sum_{i=1}^{m} (w'_{i,t} - \mu_t w_{i,t})^+ - \mu_t w_{t,0}] \\ = \sum_{i=1}^{m} (\mu_t w_{i,t} - w'_{i,t})^+\end{aligned} \tag{13}$$

以上公式最终简化为使用 c_s,c_p 表示 μ_t:

$$\mu_t = \frac{1}{1 - c_p w_{0,t}} \left(1 - c_p w'_{0,t} - (c_s + c_p - c_s c_p) \sum_{i=1}^{m} (w'_{i,t} - \mu_t w_{i,t})^+ \right) \tag{14}$$

3 算法研究

3.1 三个创新点

论文有三个创新点，提出了独立模型集成的方案（EIIE，Ensemble of Identical Independent Evaluators），向量化的投资组合记忆（PVM，Portfolio-Vector Memory），在线随机批学习（OSBL，Online Stochastic Batch Learning）技术

策略函数使用了三种不同的网络结构，CNN、RNN 和 LSTM。下面对 CNN 和 RNN 的方法做简要的介绍。

CNN 网络的输入是 t 时刻归一化的价格矩阵。输入的维度为 (f, m, n)。f 是特征数量，m 表示股票数，n 是周期数。在本文中，f=3，m=11，n=50。所以网络首先是一个 1x3，2 个特征的卷积，过 Relu，把价格矩阵卷成 2 通道特征图；接着是一个 1x48，20 个特征的卷积，过 Relu，把 2 通道特征图卷成了 20 通道特征图，再加上 wt-1 构成 21 个通道；接下来 1x1，1 个特征的卷积，就得到了 1 通道，11 个维度的特征向量，再加上现金的分量，最终得到 12 维度的特征向量，过 softmax，就得到了最终的 wt。在整个卷积过程中，各个资产的信息其实是没有进行混合，所以是独立的，但是模型是共用的。这样的好处在于不管有多少个资产，都可以用同样的方法进行卷积

RNN 网络的后半部分和 CNN 相同，前半部分的输入仍然是五十个时间周期，11 支股票的 3 个特征。经过循环神经网络，取最后一层单元的输出。

PVM 是记忆模块。它保存了 n 个时间周期内的 w 权重。每一个 mini-batch 沿时间线向前移动一个单位。

在线随机批学习选取 mini-batch 考虑了时间的因素用下面的公式来确定不同时间的 mini-batch 的可能性。距离当前时刻越远，被选取的可能性越小。

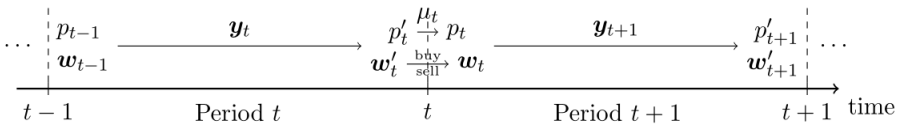


图 1: Transaction Process[13]

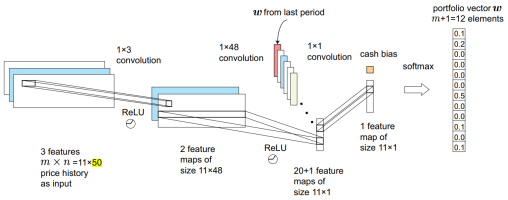


图 2: CNN 实现 [13]

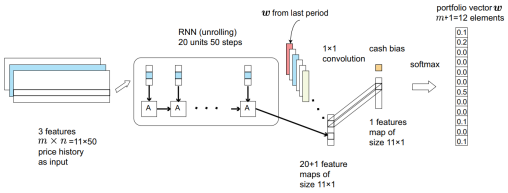


图 3: RNN 实现 [13]

$$P_\beta(t_b) = \beta(1-\beta)^{t-t_b-n_b} \tag{15}$$

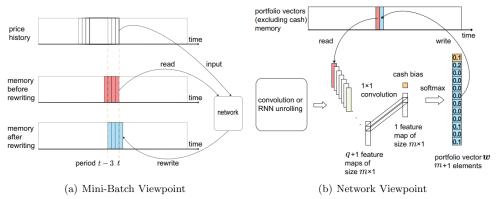
3.2 多目标强化学习思想的引入

多目标强化学习（multi-objective reinforcement learning，MORL）是一种强化学习的技术，与传统的强化学习不同的是它试图优化多个目标。在传统强化学习中，代理人（agent）被赋予了一个特定的目标，通常是最大化某个奖励信号。而在多目标强化学习中，代理人需要同时优化多个目标，这些目标通常是冲突的，即改善其中一个目标会导致其他目标的恶化。MORL 的目标是找到一个或多个最优的策略，能够在多个目标之间达到一个平衡，使得所有目标都能够得到最大化的满足。如在投资组合的问题中，我们希望在总资产越大的情况下，风险越小，这就是一个两目标优化的问题。我们根据多目标优化的思想更改了上文提到的强化学习的 Reward。

文章 [13] 中的 reward 定义为：

$$Reward = \bar{\delta p}_t \tag{16}$$

其中，$\delta p = p_t - p_{t-1}$，代表 t 时刻赚的资产数额。式17即代表从 0 时刻到 t 时刻的每时刻赚取的价值的平均值。



(a) Mini-Batch Viewpoint (b) Network Viewpoint

图 4: Mini-Batch and Network Viewpoint[13]

在引入多目标强化学习的思想以及查阅相关论文 [7] 后，我们将 Reward 重新定义为:

$$Reward1 = \max_{t=1}^{T}[\alpha * \text{mean}(\delta p_{t+k}, ...\delta p_t) - \beta * \text{std}(\delta p_{t+k}, ...\delta p_t)] \tag{17}$$

表示从开始到 T 时刻，长度为 k 的时间中，δp_t 的平均值减去标准差的最大值。该函数代表了我们在希望投资组合价值最大化的同时，希望价值的波动尽可能小。其中，α,β 和 k 都是可自定义的参数。

为了只考虑减小负的价值波动而不影响正的价值波动，我们把所有 δp_t 大于 0 的情况忽略 [7]，定义了 Reward2:

$$Reward2 = \max_{t=1}^{T}[\alpha * \text{mean}(\delta p_{t+k}, ...\delta p_t) - \beta * \text{std}(N_{t+k}, ...N_t)] \tag{18}$$

其中，$N_i = min(\delta p_i, 0) i = t, ..., t+k$, 即忽略了收益为正的时间段，仅希望减小收益为负数的时间段的波动。

3.3 NSGA-II

NSGA-II 是一种基于遗传算法的多目标优化方法，它通过非支配排序和拥挤距离两个关键步骤来寻找最优解。在 [17] 这篇文章中，NSGA-II 被提出，用于优化关于服务器资源配置的多目标问题，这些问题涉及到多个冲突的目标，例如最小化能源消耗和最大化舒适度。

在这次研究中，我们使用 NSGA-II（非支配排序遗传算法 II）作为我们多目标优化的主要算法思路，目标就是找到这样的帕累托最优解集，这些解在所有对于策略的评估目标之间达到了最佳的权衡。

关于 NSGA-II 的实现，首先，通过随机初始化产生一个种群。然后，通过选择、交叉和变异操作产生新的种群。在选择过程中，使用非支配排序来确定哪些解是最优的。非支配排序是一种方法，它将种群中的解按照支配关系分成几个等级。在同一等级中，解之间没有支配关系。等级越高，解的质量越好。

然后，通过计算每个解的拥挤距离来保持种群的多样性。拥挤距离是一种度量，它表示一个解在目标空间中的邻居有多远。拥挤距离越大，表示该解周围的其他解越少，因此该解的质量更高。

在本次研究的情景下，我们使用 NSGA-II 做多目标优化。我们需要优化三个评价指标，综合三项来评估产生的投资组合策略的价值。

3.4 NSGA-II 的实现

Algorithm 1 NSGA-II main loop

get $R_t = P_t \cup Q_t$
$F = fast\_non\_dominated\_sort(R_t)$
set $P_{t+1} = \emptyset$ and $i = 1$
while $|P_{t+1}| + |F_i| <= N$ **do**
  $P_{t+1} = P_{t+1} \cup F_i$
  $i = i + 1$
end while
$sort(F_i)$
$setP_{t+1} = P_{t+1} \cup F_i[1 : (N - |P_{t+1}|)]$
get $Q_{t+1} = make\_new\_pop(P_{t+1})$
set $t = t + 1$

P_t 代表上一个周期结束后选出来的父代，Q_t 是父代通过交叉以及变异得出的子代，R_t 代表当前周期的种群。将当前周期的种群通过快速非支配排序得到 F,F_i 代表第 i 个非支配前沿。NSGA-II 的主要循环就是先按照非支配前沿等级将 F_i 填充到 P_{t+1}，当一层非支配前沿的个体数量超过 P_{t+1} 能选择的剩余数量，则将该层非支配前沿计算拥挤距离并排序，选择靠前的个体填充完 P_{t+1}。得到新父代 P_{t+1} 后，再通过交叉、变异获得子代 Q_{t+1}。

4 实验 & 回测

本文提到的三个策略网络在不同时间段的加密货币交易所 Poloniex 上进行回测。结果与许多已经成熟和最近发表的投资组合选择策略进行比较。主要比较的评价指标是投资组合价值、最大回撤和夏普比率。

4.1 实验环境

下图中中呈现了回测实验的时间范围和对应的训练集的详细信息。其中，CV 交叉验证集用于确定超参数的范围。表中所有时间都是协调世界时（UTC）。所有训练集都从 0 点开始。例如，回测 1 的训练集是从 2014 年 11 月 1 日 00:00 开始的。所有价格数据都是通过 Poloniex 官方的应用程序编程接口（API）4 进行访问的。

Data Purpose	Data Range	Training Data Set
CV	2016-05-07 04:00 to 2016-06-27 08:00	2014-07-01 to 2016-05-07 04:00
Back-Test 1	2016-09-07 04:00 to 2016-10-28 08:00	2014-11-01 to 2016-09-07 04:00
Back-Test 2	2016-12-08 04:00 to 2017-01-28 08:00	2015-02-01 to 2016-12-08 04:00
Back-Test 3	2017-03-07 04:00 to 2017-04-27 08:00	2015-05-01 to 2017-03-07 04:00

图 5: 实验数据范围 [13]

4.2 评价指标

4.2.1 fAPV

APV（accumulative portfolio value），即累计组合价值，是评估一个策略是否优秀的最重要的指标。令 t 时刻的组合价值为 p_t，开始时刻的组合价值为 p_0，定义 t 时刻的 APV：

$$APV = p_t / p_0 \tag{19}$$

fAPV 则是最终时刻的 APV。

4.2.2 SR

APV 的一个主要缺点是它不能衡量风险因素，因为它仅仅是简单地加总所有周期性收益，而不考虑这些收益的波动。为了考虑风险因素，第二个指标——夏普比率（Sharpe ratio, SR）（Sharpe, 1964, 1994）被使用:

$$SR = \frac{\mathrm{E_t}[\rho_t - \rho_F]}{\sqrt{\mathrm{var_t}(\rho_t - \rho_F)}} \tag{20}$$

其中，$\rho_t = p_t / p_{t-1}$，代表 t 时刻的收益率；ρ_F 是无风险资产的收益率，在这里无风险资产是比特币。因为报价货币也是比特币，所以无风险收益率为零，即 $\rho_F = 0$。

4.2.3 MDD

夏普比率（SR）虽然考虑了投资组合价值的波动性，但是它平等地对待了上升和下降的波动。事实上，上升的波动性有助于正收益，而下降的波动性则有助于亏损。为了突出下行偏差，本实验还考虑了最大回撤（Maximum Drawdown，MDD）（Magdon-Ismail and Atiya，2004）。MDD 是从峰值到低谷的最大损失，数学上可以表示为：

$$D = \max_{\substack{\tau > t \\ t}} \frac{p_t - p_\tau}{p_t} \tag{21}$$

4.3 使用的对比策略

实验的对比策略是 benchmark 中的 Constant Rebalanced Portfolios(CRP) 策略和跟随输家策略中的 OLMAR 策略。

4.3.1 CRP

CRP 策略是指，初始化一个投资组合权重（该实验是均分权重），在每个时期结束时，会重新平衡资产组合，以使其重新达到初始配置的权重。

4.3.2 OLMAR

OLMAR 策略是指，在历史上表现最差的资产上分配更多的资本，而在表现最好的资产上分配更少的资本。

4.4 实验过程

由于本项目提供了一种利用神经网络进行交易决策的方法，通过训练神经网络（CNN）来预测市场行情，并根据预测结果进行交易，从而获得更好的收益。下文介绍该项目的实验部分，包括数据准备、网络结构配置、训练和调参、回测等内容。

4.4.1 数据准备

项目提供了两种数据获取方式：在线获取和本地数据库读取。默认为在线获取，用户可以在 `net_config.json` 中设置数据获取相关参数，包括起止时间、币种数量、交易周期、测试数据占比等。

4.4.2 网络结构配置

通过修改 `net_config.json` 文件来配置网络结构。其中包括神经网络层数、类型、卷积核形状、输入矩阵窗口大小、币种数量、特征数量等。

4.4.3 训练和调参

首先运行命令行命令生成训练所需的文件夹。然后，可以通过运行指定的命令来进行训练。可以使用不同的随机数种子一次性产生多个训练文件夹，并且调用 gpu 来进行多线程运行。训练完成后，相关统计信息整理至 `train_summary.csv` 中，包括网络配置、验证集和测试集上的收益等。我们根据这些信息来调整网络结构和训练参数，然后重新运行训练过程。

4.4.4 回测

训练完成后，我们继续使用剩下的数据集进行回测。

4.4.5 绘图 & 表格

得到回测数据后，结合其他的一般性策略，绘制图片和表格，更直观地展示。

4.5 实验 & 回测结果

4.5.1 [13] 提到的结果

图6中，粗体算法 (CNN, bRNN, LSTM) 是本文介绍的 EIIE 网络。此外，本文还测试了三个基准算法（斜体）和一些最近的其他投资组合策略（Li et al.，2015a; Li and Hoi，2014）。

图6的算法被分成五类 [15]，从上到下分别是：

- 无模型神经网络 (model-free)：引入神经网络 (neural netwark) 来解决投资组合问题，而不需要先对问题建立一个明确的数学模型或规则。实验中包括本文提出的三个 EIIE 算法，以及 iCNN[12]。
- 基准算法 (benchmark): 基准策略没有复杂的主动调仓思想，常被用作与新的策略进行比较来验证新策略的优越性。实验中包括 Best stock, UCRP[14] 和 UBAH。
- 跟随输家策略 (follow the loser): 跟随输家策略基于 experts/stocks 的历史表现，增加不太成功的 experts/stocks 的相对权重。跟随输家策略的潜在假设是市场遵循均值回归原则，过去表现良好（不良）的资产将在接下来的时期表现不佳（良好）。实验中包括 Anticor[1]，OLMAR[3]，PAMR[5], WMAMR[11], CWMR[4] 和 RMR[10]。
- 跟随赢家策略 (follow the winner)：基于 experts/stocks 的历史表现，增加更成功的 experts/stocks 的相对权重。跟随赢家策略的潜在假设是市场具有一定趋势性，过去表现好的资产大概率在未来也会表现良好。实验中包括 ONS[2]，UP[8] 和 EG[9]。
- 模式匹配策略 (pattern matching): 在投资组合问题中，pattern matching 策略是一种基于历史数据分析的投资策略。它通过分析历史数据中的趋势、模式和规律，来预测未来的市场走势和股票价格走势，从而指导投资决策。实验中包括 B^K[16], 和 CORN[6]。

其中，每一列中表现最好的数据都以粗体突出显示。

总体来说，在 fAPV 或 SR 方面，第 1 和第 2 次回测中表现最好的算法是 CNN，其最终财富在第一次实验中超过第二名两倍以上；第 3 次回测中表现最好的是 bRNN。在所有回测中，这两个指标排名前三的都是 EIIE 网络，可以说 EIIE 策略仅在 MDD 指标上稍稍输给了其他策略。三个 EIIE 算法都在 fAPV 和 SR 列中显著优于所有其他算法，表明 EIIE 机器学习解决方案在投资组合管理问题上的盈利性和可靠性。

	2016-09-07 to 2016-10-28			2016-12-08 to 2017-01-28			2017-03-07 to 2017-04-27		
Algorithm	MDD	fAPV	SR	MDD	fAPV	SR	MDD	fAPV	SR
CNN	0.224	**29.695**	**0.087**	0.216	**8.026**	**0.059**	0.406	31.747	0.076
bRNN	0.241	13.348	0.074	0.262	4.623	0.043	0.393	**47.148**	**0.082**
LSTM	0.280	6.692	0.053	0.319	4.073	0.038	0.487	21.173	0.060
iCNN	**0.221**	4.542	0.053	0.265	1.573	0.022	0.204	3.958	0.044
Best Stock	0.654	1.223	0.012	0.236	1.401	0.018	0.668	4.594	0.033
UCRP	0.265	0.867	-0.014	**0.185**	1.101	0.010	**0.162**	2.412	0.049
UBAH	0.324	0.821	-0.015	0.224	1.029	0.004	0.274	2.230	0.036
Anticor	0.265	0.867	-0.014	0.185	1.101	0.010	0.162	2.412	0.049
OLMAR	0.913	0.142	-0.039	0.897	0.123	-0.038	0.733	4.582	0.034
PAMR	0.997	0.003	-0.137	0.998	0.003	-0.121	0.981	0.021	-0.055
WMAMR	0.682	0.742	-0.0008	0.519	0.895	0.005	0.673	6.692	0.042
CWMR	0.999	0.001	-0.148	0.999	0.002	-0.127	0.987	0.013	-0.061
RMR	0.900	0.127	-0.043	0.929	0.090	-0.045	0.698	7.008	0.041
ONS	0.233	0.923	-0.006	0.295	1.188	0.012	0.170	1.609	0.027
UP	0.269	0.864	-0.014	0.188	1.094	0.009	0.165	2.407	0.049
EG	0.268	0.865	-0.014	0.187	1.097	0.010	0.163	2.412	0.049
B^K	0.436	0.758	-0.013	0.336	0.770	-0.012	0.390	2.070	0.027
CORN	0.999	0.001	-0.129	1.000	0.0001	-0.179	0.999	0.001	-0.125
M0	0.335	0.933	-0.001	0.308	1.106	0.008	0.180	2.729	0.044

图 6: 实验结果 [13]

4.5.2 期中实验结果

algorithm	MDD	fAPV	SR
nntrader	**0.225874**	45.32045	0.084868
nntrader1	0.226498	44.377729	0.087820
nntrader2	0.227170	**47.061779**	**0.088141**
olmar	0.604886	4.586964	0.036600
crp	0.233598	1.835544	0.03418

4.5.3 本次实验结果

algorithm	MDD	fAPV	SR
elite1	**0.220393**	**45.432344**	0.087981
elite2	0.227351	44.97729	**0.08882**
origin	0.2217	44.3561	0.08868

5 总结

5.1 项目总结与目标

本项目旨在实现并优化 Zhengyao Jiang 等人在论文《A Deep Reinforcement Learning Framework for the Financial Portfolio Management Problem》[13] 中提出的金融投资组合管理问题的深度强化学习框架。我们的目标是通过在现有框架基础上进行一系列改进和优化，提高投资组合管理的性能。项目前期到中期阶段，我们跑通了论文 [13] 对应的代码，并且复现了论文中的结果。同时将多目标强化学习的思想 [7] 与 EIIE 框架相结合，考虑了利润与风险的平衡，得出了令人满意的结果。在项目中期到后期的阶段，我们将尝试一系列的优化措施，可能包括：

1) 完善多目标强化学习：17和18中的 α,β 和时间段长度 k 都属于可以自定义的参数，我们将尝试用演化算法训练参数，以获得一组帕累托最优解，分别代表了不同的风险偏好，实现策略的多样性。
2) 更换强化学习模型：我们将尝试使用不同的强化学习模型（如 DQN、DDPG 等），以找到适用于金融投资组合管理问题的最佳模型。
3) 数据集扩展：我们将使用更多样化和丰富的数据集进行训练和测试，包括不同市场、不同资产类型和不同时间段的数据，以提高模型的泛化能力。

:

5.2 评价与改进

本学期的项目以深度强化学习的角度对股市做了投资组合的解答和分析，提出的模型距离实际情况仍有距离和偏差，例如时间原因，样本数据集不够全面，没有区分市场大环境，周期，可能缺乏一定的跨周期性能（牛市，熊市），且评估指标不够完善等。

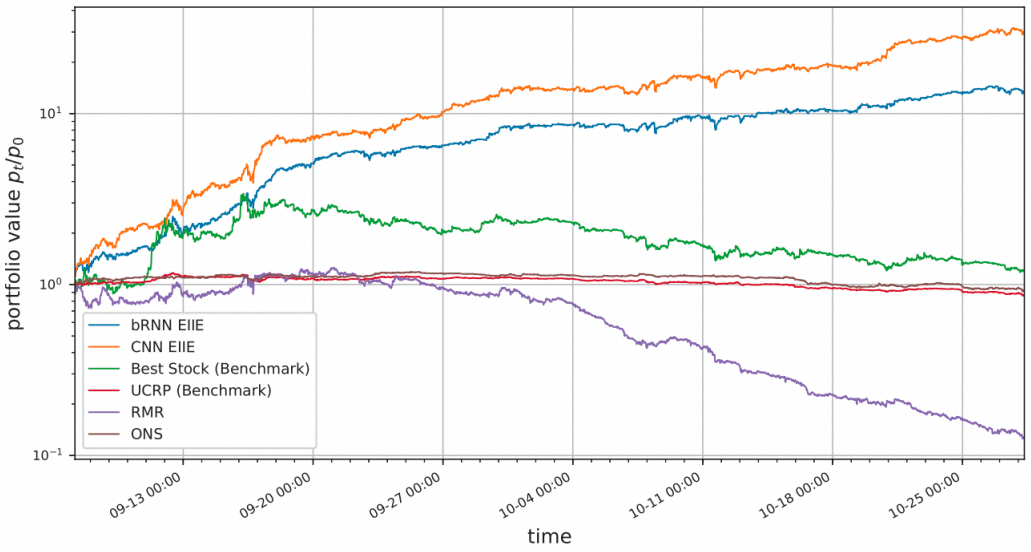


图 7: Back-Test 1[13]

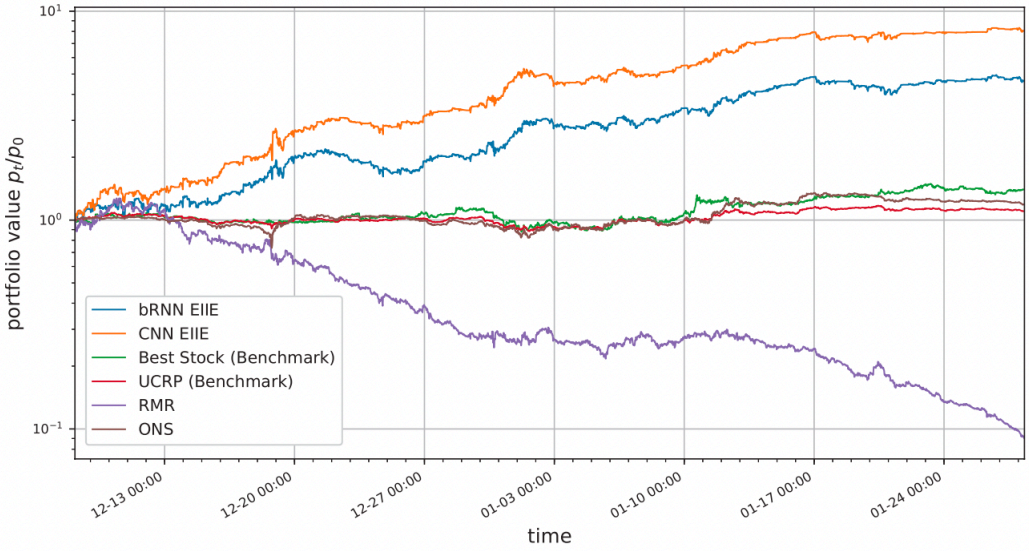


图 8: Back-Test 2[13]

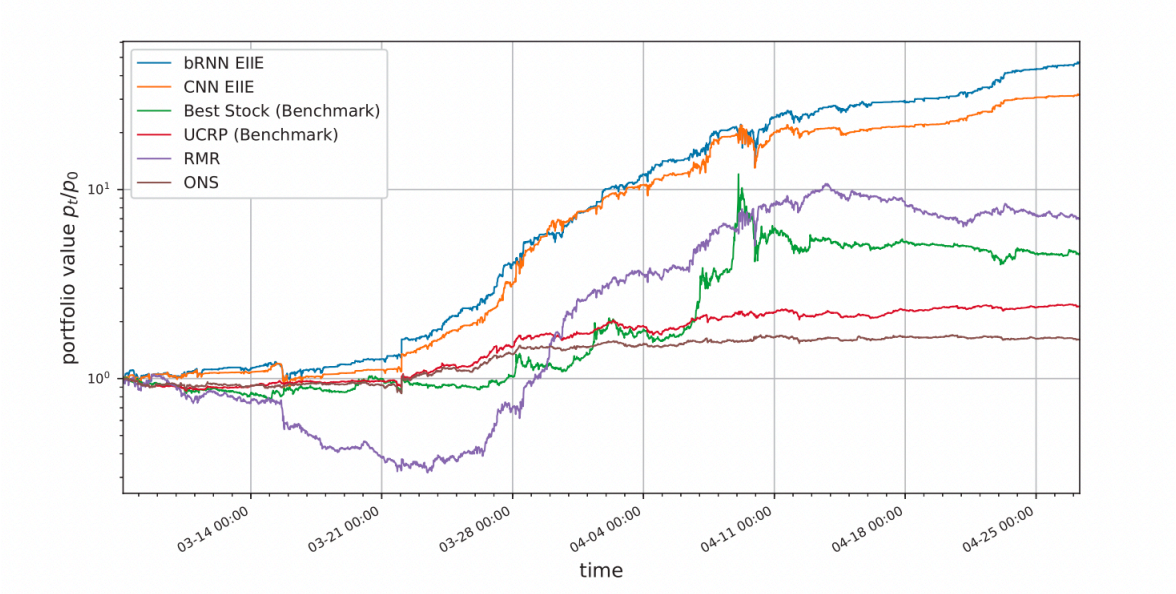


图 9: Back-Test 3[13]

参考文献

[1] Ran El-Yaniv Allan Borodin and Vincent Gogan. Can we learn to beat the best stock. *J. Artif. Intell. Res.(JAIR),*, 2004.

[2] Satyen Kale Amit Agarwal, Elad Hazan and Robert E Schapire. Algorithms for portfolio management based on the newton method. *In Proceedings of the 23rd international conference on Machine learning*, pages 9–16, 2006.

[3] Doyen Sahoo Bin Li, Steven CH Hoi and Zhi-Yong Liu. Moving average reversion strategy for on-line portfolio selection. *Artificial Intelligence*, 2015b.

[4] Peilin Zhao Bin Li, Steven CH Hoi and Vivekanand Gopalkrishnan. Confidence weighted mean reversion strategy for online portfolio selection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2013.

[5] Steven C. H. Hoi Bin Li, Peilin Zhao and Vivekanand Gopalkrishnan. Pamr: Passive aggressive mean reversion strategy for portfolio selection. *Machine Learning, 87(2):221–258*, 2012.

[6] Steven CH Hoi Bin Li and Vivekanand Gopalkrishnan. Corn: Correlation-driven non-parametric learning approach for portfolio selection. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2011.

[7] Kiran Bisht and Arun. Kumar. Deep reinforcement learning based multi-objective systems for financial trading. *2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE)*, 2020.

[8] Thomas M Cover. Universal portfolios. *Mathematical finance*, 1991.

[9] Yoram Singer David P Helmbold, Robert E Schapire and Manfred K Warmuth. On-line portfolio selection using multiplicative updates. *Mathematical Finance*, 1998.

[10] Bin Li Steven CH Hoi Dingjiang Huang, Junlong Zhou and Shuigeng Zhou. Robust median reversion strategy for on-line portfolio selection. *IJCAI*, pages 2006–2012, 2013.

[11] Li Gao and Weiguo Zhang. Weighted moving average passive aggressive algorithm for online portfolio selection. *Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, 2013.

[12] Zhengyao Jiang and Jinjun Liang. Cryptocurrency portfolio management with deep reinforcement learning. *In Proceedings of 2017 Intelligent Systems Conference.*, 2017.

[13] Zhengyao Jiang, Dixing Xu, and Jinjun Liang. A deep reinforcement learning framework for the financial portfolio management problem. *IEEE Transactions on Neural Networks and Learning Systems*, 2017.

[14] J. L. Kelly. A new interpretation of information rate. *The Bell System Technical Journal*, 1956.

[15] Bin Li and Steven CH Hoi. Online portfolio selection: A survey. *ACM Computing Surveys(CSUR)*, 2014.

[16] G´abor Lugosi L´aszl´o Gy¨orfi and Frederic Udina. Nonparametric kernel-based sequential investment strategies. *Mathematical Finance*, 2006.

[17] Haifeng LIU Peng YANG, Laoming ZHANG and Guiying LI. Reducing idleness in financial cloud via multi-objective evolutionary reinforcement learning based load balancer.