



國立臺灣大學電機資訊學院資訊工程學研究所

碩士論文

Department of Computer Science and Information Engineering College of
Electrical Engineering and Computer Science

National Taiwan University

Master Thesis

分布式強化學習應用於投資組合管理

Distributional Reinforcement Learning in Portfolio Management

吳海韜

Hai-Tao Wu

指導老師：呂育道 教授

Advisor: Yuh-Dauh Lyuu, Prof.

中華民國 112 年 1 月

January, 2023

國立臺灣大學碩士學位論文
口試委員會審定書
MASTER'S THESIS ACCEPTANCE CERTIFICATE
NATIONAL TAIWAN UNIVERSITY

分布式強化學習應用於投資組合管理

Distributional Reinforcement Learning in Portfolio
Management

本論文係吳海韜君（學號 R08922051）在國立臺灣大學資訊工程學系完成之碩士學位論文，於民國 112 年 1 月 16 日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Department of Computer Science and Information Engineering on 16 January 2023 have examined a Master's thesis entitled above presented by WU, HAI-TAO (student ID: R08922051) candidate and hereby certify that it is worthy of acceptance.

口試委員 Oral examination committee:

呂育道

(指導教授 Advisor)

張弘明

陸裕豪

系主任/所長 Director:

洪士瀨

摘要



在金融市場中，如何在不同資產配置比例間找出最適的投資組合，一直是重要的課題。隨著近年人工智慧技術蓬勃發展，越來越多研究將強化學習運用在投資問題中。在基於值方法的強化學習中，有一分支為分布式強化學習技術：以投資問題來說相當於先預測投資組合未來報酬率的統計分布，再根據分布做出合理的投資決策。

本文運用這種方法在台灣股票市場上進行研究。我們發現相較於傳統只有預測期望值來做決策的值方法，預測統計分布的方法在投資問題中能捕捉到風險的資訊，進而找出更好的投資組合，有效提升年化報酬率以及夏普率。

關鍵字：分布式強化學習、深度學習、投資組合管理、機器學習、風險管理

Abstract



In financial markets, finding the most suitable asset allocation has always been key. With the rapid development of artificial intelligence in recent years, more research has applied reinforcement learning to investment problems. Distributional reinforcement learning is a branch of value-based reinforcement learning. In investment terms, it is equivalent to predicting the statistical distribution of the future rate of return of the portfolio and then making investment decisions based on it.

This thesis applies distributional reinforcement learning to the Taiwan stock market. Compared with the traditional value-based reinforcement learning, which only uses the predicted expected value to make decisions, the method that uses the predicted distribution can capture risk and thus find a better portfolio, simultaneously improving the annualized rate of return and the Sharpe ratio.

Keywords: distributional reinforcement learning, deep learning, portfolio management, machine learning, risk management

目錄



摘要	i
Abstract.....	ii
目錄	iii
圖表	iv
1 緒論	1
1.1 研究背景	1
1.2 研究目的	1
2 文獻回顧	3
2.1 強化學習簡介	3
2.2 值函數定義	5
2.3 Q-Learning.....	6
2.4 Deep Q-Networks	6
2.5 分布式深度強化學習	8
3 研究方法	11
3.1 實驗設計	11
3.2 資料集	14
4 模型架構	16
4.1 特徵提取	16
4.2 損失函數	17
4.3 模型訓練	18
5 模型分析	20
5.1 投資組合比較	20
5.2 強化學習決策分析	24
5.3 值分布與風險管理	26
6 結論	30
6.1 研究結果	30
6.2 未來展望與建議	30
參考文獻	32



圖表

圖表 2-1 模型 C21-SR 論文實驗結果節錄 [9].....	9
圖表 3-1 台灣市值前十大股票	14
圖表 3-2 歷史資料劃分	15
圖表 4-1 演算法 SGDR 的學習率變化	18
圖表 5-1 投資組合實驗結果	20
圖表 5-2 投資策略擊敗買進持有策略比例	21
圖表 5-3 分布式強化學習與傳統方法比較(年化報酬率)	22
圖表 5-4 分布式強化學習與傳統方法比較(夏普率)	23
圖表 5-5 強化學習決策比率	25
圖表 5-6 風險實驗組別與決策函數	27
圖表 5-7 不同風險決策指標下的分布式強化學習投資組合	27
圖表 5-8 不同風險決策指標所對應年化報酬率	28
圖表 5-9 不同風險決策指標所對應夏普率	29



1 緒論

1.1 研究背景

投資組合管理一直是金融投資領域的重要課題。這類問題可以歸總為：從給定資產池中找出合適的配置比例，在風險能承受的前提下提高投資報酬率。為了完成前述目標，人們會利用統計學習或機器學習等方法，分析未來資產報酬率與風險，期待借助電腦運算來輔助決策。

近年來隨著電腦硬體的進步，機器學習技術蓬勃發展，其中深度學習模型由於對算力需求極高，更是有所助益。2013 年 DeepMind 發表了著名的 DQN (Deep Q-Networks)模型 [1,2]，¹將深度學習成功結合到強化學習中，在玩小遊戲這個情境中訓練出能媲美甚至超越人類績效的 AI。對投資感興趣的研究者自然地會將 DQN 及其方法論應用到資產配置的決策上，於是各式深度強化學習投資的研究如雨後春筍般出現。


爾後更多進階的深度強化學習方法被開發出來，例如 DeepMind 團隊 2017 年發表的 C51 模型 [3] 與 2018 年的 IQN 模型 [4]。然而專注於強化學習方法的研究者在評估其模型好壞時，大部分會選擇探討常見任務(例如小遊戲)的表現，極少以投資問題作為評估目標。因此，這些各式各樣的新方法是否有助於投資是實務上值得探索的議題。

1.2 研究目的

相比於其他常見的強化學習任務，投資任務表現好壞能同時從報酬和風險兩個維度來衡量。本文將兼顧這兩者，以基於值方法的(value-based)強化學習為範疇，針對傳統值方法以及其分支分布式強化學習²(distributional reinforcement learning)進行研究：

¹ 完整的實作細節於 2015 年發表於 *Nature*，亦有人認為這是 DQN 發表的時點，詳見 2.4 節。

² 亦有翻譯作值分布強化學習。本文分布一詞指的是統計學上的機率分布(probability distribution)。有另外專門探討分散式運算(distributed computing)運用在強化學習的研究領域，少數人也會將其中文翻譯成分布式強化學習，為避免混淆在此說明。

- 
1. 驗證分布式方法相較傳統方法而言是否更適合投資問題。
 - 分布式方法在年化報酬率(annualized rate of return)、夏普率(Sharpe ratio)均勝過傳統方法。³
 2. 討論分布式強化學習在投資問題上的經濟意義。
 - 透過預測統計分布，AI 能學會在投資任務中分散風險。⁴
 - 研究者可以透過分析其他風險指標，例如風險值(VaR, Value at Risk)[5]，設計出更符合實際需求的 AI 投資策略。⁵

本文在接下來第二章將藉由文獻回顧介紹預測統計分布與傳統預測期望值的不同，第三章定義投資問題中本研究採取的強化學習設定。在這兩章的基礎下，第四章描述深度學習模型架構與訓練方法。實驗結果將在第五章進一步分析與討論，最後給出結論，並提供我們對這個問題未來發展的建議。

³ 詳見 5.1 節投資組合比較。

⁴ 詳見 5.2 節強化學習決策分析。

⁵ 詳見 5.3 節值分布與風險管理。



2 文獻回顧

2.1 強化學習簡介

本節以投資的觀點介紹強化學習模型及其常用符號。我們以馬可夫決策過程 (Markov decision process) 來建模環境 \mathcal{E} ：

$$\mathcal{E} = (\mathcal{X}, \mathcal{A}, R, P)$$

- x 表示狀態集合(states)，其中時刻 t 的狀態為 $x_t \in \mathcal{X}$ ，例如資產價格、目前持有的資產組合、與金融市場上的其他資訊等。
- \mathcal{A} 表示行動集合(actions)，採取的行動 $a_t \in \mathcal{A}$ 可能包括買進、賣出資產的種類與數量。
- P 表示狀態轉移核心(transition kernel) $P(\cdot | x, a)$ 。本文的符號中將次期狀態 x_{t+1} 服從一個給定 x_t, a_t 下的統計分布，簡記作 $x_{t+1} \sim P(x_t, a_t)$ 。 P 描述了在給定狀態與行動下轉移至未來狀態的規則，例如買賣成交與否可能影響下一期持有組合，甚至可能影響市場上其他參與者的行為等。
- R 為獎賞函數(reward)，其中採取行動 a_t 的獎賞為 $R(x_t, a_t, x_{t+1})$ 。這是判斷行動好壞的標準，以投資來說可以是交易賺賠的金額或報酬率。由於轉移後的狀態已經由環境定義 $x_{t+1} \sim P(x_t, a_t)$ ，研究者通常省略 x_{t+1} ，記作 $R(x_t, a_t)$ 。

給定環境 \mathcal{E} ，一個策略(policy)可以被定義為將狀態映射到行動的策略函數⁶(policy function) $\pi(x_t) = a_t$ ：

$$\pi: \mathcal{X} \rightarrow \mathcal{A}$$

強化學習是找出最佳策略⁷ $\pi^* = \arg \max_{\pi \in \Pi} g(\pi)$ ，其中目標函數為：

$$g(\pi) \equiv E \left[\sum_{t=0}^{\infty} \gamma^t R(x_t, \pi(x_t)) \right]$$

⁶ 如同一般值方法的研究，本文的策略指的都是確定性策略(deterministic policy)。

⁷ 最佳策略 π^* 可能有多個，所形成的集合一般記作 Π^* 。

意即給定折現因子 $0 \leq \gamma \leq 1$ ，⁸最佳策略是最大化未來各期獎賞折現加總的期望值。值得注意的是現實中最佳策略的評價方法不是唯一，跳脫強化學習框架之外的評價函數 $g'(\pi)$ 是存在的。假設一個以對數報酬率為獎賞函數的投資問題 \mathcal{E} ，其目標函數 g 即為期望對數報酬率，我們不只關心期望報酬率 $g(\pi)$ ，也在乎一個投資策略的風險報酬比 $g'(\pi)$ ，例如將平均值除以標準差：

$$g'(\pi) = \frac{\mu(g(\pi))}{\sigma(g(\pi))}$$

其經濟意涵為 $g'(\pi)$ 數值較高的策略將會有較高的夏普率，⁹本文的實驗分析中同時也會報告夏普率做為風險高低的參考指標。

雖然折現因子 γ 不是馬可夫決策過程的必要成分，卻是強化學習中定義策略好壞的重要參數，因此近代強化學習研究文獻(例如 C51 [3])經常直接將其納入環境的定義中：

$$\mathcal{E} = (\mathcal{X}, \mathcal{A}, R, P, \gamma)$$

強化學習演算法如何找出 π^* ，方法可分為三大類：

1. 基於值(value-based)：找出一個能準確評估未來好壞的決策函數 $f: \mathcal{X} \times \mathcal{A} \rightarrow R$ ，基於這個函數形成策略 $\pi_f(x_t) = \arg \max_{a \in \mathcal{A}} f(x_t, a)$ 。
2. 基於策略(policy-based)：直接學習策略函數 π 本身。
3. 基於模型(model-based)：利用對環境 \mathcal{E} 本身的知識去建構 π ，例如在 AlphaGo [6] 的策略中使用了有關圍棋規則的資訊。

本文專注於討論基於值方法的深度強化學習。設深度學習模型參數為 θ ，訓練出良好的決策函數 $f_\theta(x_t, a)$ 以後，用來結合基於策略方法(例如 D4PG [7] 結合了 C51)或模型方法得到更好的結果是有可能的，雖然這不是本研究的主題。

⁸ 適當地定義環境能確保 $g(\pi)$ 不會發散，即使 $\gamma = 1$ 亦然。本文就是採取這種設定，詳見 3.1 節。

⁹ 在夏普率的定義中，本文假設無風險資產報酬率是固定常數 0.01365。為了將日報酬率轉換為年化報酬率，本文假設一年有 244 個交易日。



2.2 值函數定義

給定環境 \mathcal{E} 及策略函數 $\pi(s)$ ，強化學習的目標函數是：

$$g(\pi) \equiv E \left[\sum_{t=0}^{\infty} \gamma^t R(x_t, \pi(x_t)) \right]$$

考慮給定策略 π 以及目前狀態 s ，定義價值函數(value function)：

$$V_{\pi}(x_t) \equiv E \left[\sum_{t=0}^{\infty} \gamma^t R(x_t, a_t) \mid a_t = \pi(x_t) \right]$$

對於同樣的初始狀態 x_0 來說，求解強化學習問題就是最大化價值函數：

$$V_{\pi}(x_0) = g(\pi)$$

由貝爾曼方程(Bellman equation)，價值函數可以被分解為本期獎賞以及未來獎賞兩個部分，得到比較容易操作的形式：

$$V_{\pi}(x_t) = E[R(x_t, a_t) + \gamma V_{\pi}(x_{t+1}) \mid a_t = \pi(x_t)]$$

推廣 V 函數進一步考慮行動 a 帶來的影響，定義 Q 為行動價值函數(action-value function)：

$$Q_{\pi}(x_t, a) = E[R(x_t, a) + \gamma V_{\pi}(x_{t+1}) \mid a_t = a]$$

其中 V 是 Q 的一個特例，即 $V_{\pi}(x_t) = Q_{\pi}(x_t, \pi(x_t))$ 。代入得到值函數¹⁰ Q 常用的操作型定義：

$$Q_{\pi}(x_t, a) = E[R(x_t, a) + \gamma Q_{\pi}(x_{t+1}, \pi(x_{t+1})) \mid a_t = a]$$

¹⁰ 所謂值函數有可能代表 Q 、 V 或 g ，本文指的是 Q 。



2.3 Q-Learning

假設我們掌握了最佳策略 π^* 對應的值函數 $Q_{\pi^*}(x_t, a)$ ，那麼遵循這個完美值函數 Q_{π^*} 就是一種最佳策略：

$$\pi^*(x_t) = \arg \max_{a \in \mathcal{A}} Q_{\pi^*}(x_t, a)$$

Q-learning 的目標是為這個完美值函數 $Q_{\pi^*}(x_t, a)$ 找出一個良好的決策函數 $f(x_t, a)$ ，由此得出策略：

$$\pi_f(x_t) = \arg \max_{a \in \mathcal{A}} f(x_t, a)$$

通常 Q-learning 中決策函數 $f(x_t, a)$ 以及對應的策略 π_f 會不斷地迭代改變 $f_1, f_2, \dots, f_{i-1}, f_i, f_{i+1}, \dots$ ，其中 f_i 所對應的策略簡記為 $\pi_i \equiv \pi_{f_i}$ 。一個良好的強化學習演算法追求的目標是讓這個過程最終能得出一個最佳策略：

$$\lim_{i \rightarrow \infty} g(\pi_i) = g(\pi^*)$$

最理想的作法是以 $f(x_t, a)$ 估計 $Q_{\pi^*}(x_t, a)$ ，也就是說決策函數 f 的目標是值函數 Q 。然而實際上 π^* 在迭代的過程中是未知的，實務作法是每次迭代 f_i 所估計的是目前策略 π_i 所對應的值函數 Q_{π_i} 。由於 π^* 與 π_i 並不相同，這導致了在過去深度強化學習有著訓練不穩定的問題 [1,2,8]，直到 DQN [1,2] 的問世才為這個問題提供了實務上可行的訓練方法。

2.4 Deep Q-Networks

DeepMind 團隊於 2013 年於 *NIPS* [1] 首度發表 Deep Q-Networks 模型，成功將深度學習與 Q-learning 結合，2015 年於 *Nature* [2] 公布了更深入的實作細節，後續的研究者通常將這兩篇論文提供的模型與方法論簡稱 DQN。

每次迭代會先採取目前的策略 π_i 實際與環境 $\mathcal{E} = (\mathcal{X}, \mathcal{A}, R, P, \gamma)$ 互動後，將結果收集到經驗回放空間(experience replay buffer)中，形成資料集 $\mathcal{D} = \{e_1, e_2, \dots\}$ 。每一筆互動 e_j 紀錄一個決策時間點的狀態、採取的行動以及結果：

$$e = (x_t, a_t, r_t, x_{t+1})$$

其中 r_t 是該時點實際獲得的獎賞，是分布 $R(x_t, a_t)$ 的實現值。¹¹

本文借鑑模型 Double DQN 的論文 [8] 中對原始 DQN 的分析方法與符號來介紹。設此輪迭代的深度學習模型參數為 θ_i ，決策函數形式為 $f_i \equiv f(x, a; \theta_i)$ ，此時以監督式學習(supervised learning)的角度可以看成一個回歸問題，訓練資料 $e \in \mathcal{D}$ ，欲估計的目標值簡記作 Y_t ，採取均方誤差(mean-squared error)，¹²損失函數為：

$$L_i(\theta_i) = E_{e \in \mathcal{D}} \left[(Y_t - f(x_t, a_t; \theta_i))^2 \right]$$

決策函數 f_i 所估計的目標為 Q_{π_i} ，回顧其定義：

$$Q_{\pi_i}(x_t, a) = E \left[R(x_t, a_t) + \gamma Q_{\pi_i}(x_{t+1}, \pi_i(x_{t+1})) \mid a_t = a \right]$$

將已實現的部分替代隨機變數，用估計值 f_i 替代未知的部分 Q_{π_i} ，得到其中一種 Y_t 的表達方式：

$$Y_t^{\text{online}} \equiv r_t + \gamma \max_{a \in \mathcal{A}} f(x_{t+1}, a; \theta_i)$$

DQN 指出這樣做並不可行，因為目標 Y_t 和估計 f_i 同時依賴相同的參數 θ_i 將會導致訓練震盪甚至發散。DQN 提出的解決方案是另外儲存一組目標網路(target network)參數 θ_i^- ，固定於每 τ 輪迭代後才更新 $\theta_i^- \leftarrow \theta_i$ ，這 τ 輪裡模型有固定的參考目標，從而達成訓練的穩定性：

$$Y_t^{\text{target}} \equiv r_t + \gamma \max_{a \in \mathcal{A}} f(x_{t+1}, a; \theta_i^-)$$

在 Atari 小遊戲裡面，DQN 僅採用了遊戲畫面像素作為模型輸入。對於環境 \mathcal{E} 規則沒有先備知識的情況下，用同樣的模型架構與方法，分別學會數個不同規則的小

¹¹ 這裡假設獎賞是隨機變數。

¹² 實際上 DQN [1,2] 為了數值穩定做了其他微調，不完全是 MSE，此處不贅述其細節。

遊戲，達成無須環境模型(model-free)的強化學習。這是一個驚人的成果。在這之後來自各種不同領域的研究者開始將深度強化學習應用到現實問題中，包含對投資感興趣的研究者。



2.5 分布式深度強化學習

2.5.1 Categorical DQN (C51)

延續 DQN 的成功，2017 年 DeepMind 團隊發表了 C51 模型 [3]，是一種分布式強化學習演算法。其模型與方法論本文簡稱為 C51。值函數 $Q_\pi(x, a)$ 是未來可能獎賞的期望值，C51 則進一步探討取期望值之前其背後對應的值分布 (value distribution) $Z_\pi(x, a)$ 為何，也就是說：¹³

$$Q_\pi(x, a) \equiv E[Z_\pi(x, a)]$$

相比於傳統 DQN 預測值函數 Q_π ，C51 則嘗試預測值分布 Z_π ，並提出了 Categorical DQN 演算法。為了以深度學習建模統計分布，C51 採取長條圖的概念，從某個範圍 $[V_{\min}, V_{\max}]$ 內選出 N 個代表值，¹⁴例如：

$$z_1, z_2, z_3, \dots, z_{50}, z_{51} = \{-10, -9.6, -9.2, \dots, 9.6, 10\}$$

沿用 2.4 節對 DQN 的分析，這裡的任務變成估計目標值 Y_t 落在分類 z_j 的機率 $p_j(x_t, a)$ ，設法訓練模型參數 θ ，以便最佳化其對應的統計分布 Z_θ 。最終模型採取 $N = 51$ 個分類，因此 [3] 將其模型命名為 C51。決策函數的形式如下：¹⁵

$$f(x_t, a; \theta) = \sum_{j=1}^{51} z_j p_j(x_t, a; \theta) = E[Z_\theta]$$

這 51 組 (z_j, p_j) 代表了深度學習模型所學到的統計分布 $z \sim Z_\theta$ ，在長條圖的例子裡是離散統計分布：

$$Pr(z = z_j) = p_j$$

¹³ 從本節開始不再強調迭代過程的部分，以 π_i 為例我們將省略下標 i 。

¹⁴ 原文若直接翻譯則為原子值(atoms)，在這裡寫成代表值表達更清楚。

¹⁵ 在長條圖這個的例子中取 $E[Z_\theta]$ 只是算加權平均，似乎沒有描述的必要。然而在進階的分布式強化學習方法中則未必如此，例如 IQN 模型 [4]。



其損失函數為 Y_t 所對應類別 m_t 與 p_j 的交叉熵(cross entropy)。從監督式學習的觀點出發，可以看作是一個多元分類(multiclass classification)問題，本研究便是採取這種方法，我們將在 4.2 節展示其細節。

C51 的研究結果顯示在原有 DQN 框架下，僅是將估計目標改為統計分布就能大幅改善模型表現。並且針對其中一個小遊戲 SPACE INVADERS 舉例分析，採取統計分布的一個好處是模型能捕捉到極端事件，例如做錯決定輸掉遊戲將再也得不到獎賞。我們認為這說明了 Z_θ 蘊含跟風險有關的資訊。

2.5.2 C51 應用於投資問題

Harnpadungkij 等人於 2019 年基於 C51 的精神提出了 C21-SR 模型 [9]，在投資問題上應用分布式強化學習，專注於提升投資組合的夏普率，是開創性的嘗試。該團隊從美國股市中挑選了 10 組股票對 $(A_1, B_1), (A_2, B_2), \dots, (A_{10}, B_{10})$ 進行實驗，其中 C21-SR 模型所具備決策函數是：

$$f(x_t, a; \theta) = \frac{E(Z_\theta)}{\sigma(Z_\theta)}$$

其中 SR 指的是夏普率。¹⁶在傳統 Q-learning 的思想裡，如果希望優化夏普率應該要直接將夏普率設置為獎賞函數才對，然而這樣做實際上並不容易。C21-SR 巧妙地運用了值分布估計 Z_θ 容易操作的特性，將決策函數抽換為平均值除以標準差，是一個與夏普率完全正相關的統計量，在不具備值方法強化學習理論基礎支持的前提下，採取了這種獨特的決策函數形式，期望能改善夏普率。

組別名稱	夏普率(平均數)	說明
C21-SR	0.838	相當於本文 5.3 節的 DRL-SR 組別
C21	0.557	相當於本文 5.1 節的 DRL 組別
DQN	0.665	相當於本文 5.1 節的 MRL 組別

圖表 2-1 模型 C21-SR 論文實驗結果節錄 [9]

圖表 2-1 節錄了 [9] 的部分實驗結果，其中特殊形式的 C21-SR 組別所對應 10 組獨立股票對實驗的平均夏普率確實較其他組別高。我們對此的解讀是：一個決策函數 f_θ 如果能捕捉到環境 \mathcal{E} 的特性，即便在理論基礎未明的情況下，仍有可能取得不錯的結果。然而組別 C21，也就是正規的分布式強化學習，其表現卻略遜於組

¹⁶ 類似 2.1 節中 g' 的概念，忽略無風險資產報酬率在這裡並不影響模型決策。

別 DQN 對應的傳統值方法強化學習。針對這個現象，我們設計了規模更大的實驗，並且將在 5.1 節與 5.3 節透過類似的設定展開討論，以期能得到更可靠的實證數據與結論。

我們將在後續的章節中以實驗數據佐證值分布模型在投資問題上確實能帶來好處，並且分析其中蘊含的經濟意義。最後嘗試討論 C21-SR 這一類特殊決策函數形式對投資績效表現的影響究竟為何。



3 研究方法

3.1 實驗設計

本節參考 C21-SR [9] 的方法來描繪投資組合問題。給定兩種資產 A 、 B ，每個決策時點需決定配置比例 $0 \leq \alpha_t \leq 1$ ，將資金按市值占比 α_t 投入 A 資產、剩餘部分 $(1 - \alpha_t)$ 投入 B 資產。如果資產有任何形式的現金股利(dividend)也按這個方式再投資。¹⁷定義強化學習問題如下：

$$\mathcal{E}_{\mathcal{A},B} = (\mathcal{X}, \mathcal{A}, P, R, \gamma)$$

不失一般性地，觀察點與決策點皆採取日為間隔單位，在每個交易日收盤時刻進行買賣。資產 $S \in \{A, B\}$ 收盤價格記作 S_t^{close} ，其對數報酬率為：

$$S_t^{\text{return}} = \log S_t^{\text{close}} - \log S_{t-1}^{\text{close}}$$

定義 (A, B, α_t) 所形成的投資組合於次日的對數報酬率為：

$$R'(A, B, \alpha_t) \equiv \alpha_t \cdot A_{t+1}^{\text{return}} + (1 - \alpha_t) \cdot B_{t+1}^{\text{return}}$$

本文旨在探討分布式強化學習的經濟意涵，為求表達精煉，實驗過程與後續分析將採取以下假設：

1. 個人買賣不會對金融市場產生影響。
2. 每個交易日收盤時刻均能以收盤價成交，並且沒有交易成本。
3. 評估投資問題的期長是有限值 $T < \infty$ 。

這樣一來強化學習問題 $\mathcal{E}_{\mathcal{A},B}$ 將具備以下性質：

1. 每期得到的獎賞 $R(x_t, a_t)$ 只與市場價格變化和當期決策有關，狀態 $x_t \in \mathcal{X}$ 只代表當前金融市場上能取得的公開資訊，不必包含過去資產配置情形。
2. 根據假設 2，給定行動 $a_t \in \mathcal{A}$ 對應的獎賞函數定義為次日對數報酬率

$$R(x_t, a_t) \equiv R'(A, B, a_t)。$$

3. 根據假設 1、2，市場狀態轉移與個人投資行動無關 $x_{t+1} \sim P(x_t)$ 。

¹⁷ 簡言之採取調整後股價，見 3.2 節。

4. 根據假設 3，強化學習目標函數是有限值 $g(\pi) < \infty$ ，最大化各期總報酬率可以令折現因子 $\gamma = 1$ ，也是有限值：

$$g(\pi) = E \left[\sum_{t=0}^T R(x_t, \pi(x_t)) \right]$$

回顧值函數的定義：

$$Q_{\pi}(x_t, a) = E \left[R(x_t, a_t) + \gamma Q_{\pi}(x_{t+1}, \pi(x_{t+1})) \mid a_t = a \right]$$

其中完美值函數 Q_{π^*} 將對應到一個最佳策略：

$$\pi^*(x_t) = \arg \max_{a \in \mathcal{A}} Q_{\pi^*}(x_t, a)$$

沿用前述符號決策函數 f 以及其對應的策略 π_f ：

$$\pi_f(x_t) = \arg \max_{a \in \mathcal{A}} f(x_t, a)$$

近代深度強化學習的一般框架下，研究者的目標是以決策函數 $f(x_t, a)$ 估計 $Q_{\pi^*}(x_t, a)$ ，以回歸問題的角度來看，期待均方誤差較小的 f 其對應的 $g(\pi_f)$ 表現也越好。承 2.5.1 小節所述，C51 指出 Q 值非常極端的情況下，例如在小遊戲中死亡，用統計分布的方式看待獎賞值能讓深度學習模型更好地捕捉環境 \mathcal{E} 的特性，並且提出用直方圖的概念來操作統計分布，相當於一個分類問題，採取交叉熵為損失函數訓練，結果大幅改善模型表現。

在金融問題上監督式學習的應用非常廣泛，其中回歸問題包含估計資產未來價格，分類問題則常見用於預測股票未來漲跌的研究。我們認為估計價格之於預測漲跌就像 DQN 之於 C51 的關係，也就是說監督式學習和基於值方法強化學習在投資任務上可以互相結合。本文所設計的 $\mathcal{E}_{\mathcal{A}, \mathcal{B}}$ 就具備這種特色：只要分別做好每一期的估計任務，就能做好整個投資任務。這個投資問題中任何策略 π 其 Q 值如下：

$$Q_{\pi}(x_t, a) = E_{\mathcal{E}_{\mathcal{A}, \mathcal{B}}} \left[R(x_t, a_t) + \gamma Q_{\pi}(x_{t+1}, \pi(x_{t+1})) \mid a_t = a \right]$$

由於期望值的線性性質(linearity of expectation)，可以將本期獎賞與未來獎賞分解：

$$Q_{\pi}(x_t, a) = E_{\mathcal{E}_{\mathcal{A}, \mathcal{B}}} [R(x_t, a_t) \mid a_t = a] + E_{\mathcal{E}_{\mathcal{A}, \mathcal{B}}} [\gamma Q_{\pi}(x_{t+1}, \pi(x_{t+1})) \mid a_t = a]$$

其中針對本期獎賞取期望值，也就是期望報酬率，可以視為一個監督式學習任務，定義其為 F ：

$$F(x_t, a) \equiv E_{\mathcal{E}_{\mathcal{A}, \mathcal{B}}} [R(x_t, a_t) \mid a_t = a]$$

前述性質 3 說明 x_{t+1} 不受 a_t 影響，所以值函數的未來部分也與本期決策無關：

$$Q_{\pi}(x_t, a) = E_{\mathcal{E}_{\mathcal{A}, \mathcal{B}}} [R(x_t, a_t) \mid a_t = a] + E_{\mathcal{E}_{\mathcal{A}, \mathcal{B}}} [\gamma Q_{\pi}(x_{t+1}, \pi(x_{t+1}))]$$

在值方法強化學習情境下我們關心的是能帶來結果最好的 a_t^* ，然而 F 與 Q 這兩個函數都會給出一樣的結果：

$$a_t^* = \arg \max_{a \in \mathcal{A}} F(x_t, a) = \arg \max_{a \in \mathcal{A}} Q_{\pi}(x_t, a)$$

其中 π 可以是任何策略，包含 π^* ，所以通過學習 F 值一樣能達成 Q-learning 的目標找出最佳策略：¹⁸

$$\pi^*(x_t) = \arg \max_{a \in \mathcal{A}} F(x_t, a) = \arg \max_{a \in \mathcal{A}} Q_{\pi^*}(x_t, a)$$

在這個無摩擦的投資框架 $\mathcal{E}_{\mathcal{A}, \mathcal{B}}$ 下，本研究將決策函數 $f(x_t, a)$ 的估計目標設置為 $F(x_t, a)$ 進行強化學習，這樣做有以下優點：

1. 研究 [1,2,8] 指出估計目標 Y_t 不固定是一個訓練不穩定的原因。然而此處 F 不會隨著迭代變化，消除了一部分訓練不穩的因素。
2. 估計 f_{θ} 相當於估計資產報酬率，有利於後續模型分析。

¹⁸ 一般的馬可夫決策過程只要設定 $\gamma = 0$ 也可以得到這個結果，換句話說本文採取的研究方法與傳統值方法在理論分析上有相通之處。



3.2 資料集

給定資產池 $\{S_1, S_2, \dots, S_n\}$ ，每次選出資產對 $(A, B) = (S_i, S_j)$ 都對應到一個獨立的投資問題 $\mathcal{E}_{A,B} \equiv \mathcal{E}(A, B)$ 。本文比較不同方法時，會將這 $n \times (n - 1)$ 種資產對以及其實驗結果同時納入考量：¹⁹

$$\{\mathcal{E}(S_i, S_j) \mid i \neq j\}$$

本研究實證部分從台灣股票市場中，選出市值前十大組成資產池，²⁰也就是 90 種資產對，見圖表 3-1。

證券代號	證券名稱
2330	台積電
2317	鴻海
2454	聯發科
2412	中華電
6505	台塑化
2308	台達電
2881	富邦金
2882	國泰金
1303	南亞
1301	台塑

圖表 3-1 台灣市值前十大股票

股票 S 包含了各交易日的開盤價、最高價、最低價與收盤價這四樣數值，²¹經由台灣經濟新報資料庫 TEJ 提供的軟體，排除減資、除權這類無實質經濟意義的事件後，²²得到調整後股價。我們希望深度學習模型能從股價歷史資料中捕捉到未來股價的行為，按照一般的研究方法將歷史資料劃分為三個部分：訓練集(training set)用於學習模型參數，驗證集(validation set)用於調整模型架構或超參數


¹⁹ 我們將 (S_x, S_y) 與 (S_y, S_x) 視為相異資產對的理由是方便比較極端情形，例如將所有資金投入資產 A ，也就是各期持有比例 $\alpha_t = 1$ ，詳見 5.1 節投資組合比較。

²⁰ 取市值高低的時點為 2023 年 1 月 1 日。

²¹ 不包含成交量。我們相信最精簡研究方法得到的實驗結果更能突顯分布式強化學習的經濟意義，追求投資報酬率的讀者可以參考 6.2.1 小節所列出的建議。

²² 說法來自 TEJ 股價模組介紹：<https://www.tej.com.tw/webtej/doc/wprcd.htm>

(hyperparameters)，最後期待在測試集(testing set)上得到良好的投資表現。見圖表 3-2。



劃分	起訖日	樣本數
訓練集	2010/11/23 至 2018/12/27	2000
驗證集	2018/12/28 至 2020/01/10	250
測試集	2020/01/13 至 2022/01/26	500

圖表 3-2 歷史資料劃分



4 模型架構

本研究將訓練兩種基於值方法的深度強化學習模型。分布式強化學習模型如同 C51，將預測未來報酬率視為分類問題。並且在深度學習設定盡量保持一致的前提下，訓練了傳統值方法模型進行比較，也就是將其視為回歸問題。本章將介紹模型架構細節。

4.1 特徵提取

為了數值穩定我們將輸入資料進行預處理(preprocessing)。給定股票 S 其股價原始資料為 \hat{s}_t ，取其時點前 60 期股價平均值與標準差，標準化得到 s_t 。承 3.2 節，對於時點 t 而言 A 、 B 兩檔股票有各自的開盤價、最高價、最低價與收盤價四個價格，標準化後合併得到每期「觀察值」是一個八維向量：

$$u_t \equiv (a_t^{\text{open}}, a_t^{\text{high}}, a_t^{\text{low}}, a_t^{\text{close}}, b_t^{\text{open}}, b_t^{\text{high}}, b_t^{\text{low}}, b_t^{\text{close}})$$

理論上狀態 x_t 包含了時點 t 能觀測到的所有資訊 $(u_t, u_{t-1}, u_{t-2}, \dots)$ ，實務上我們採取移動窗格(sliding window)的技巧，取最近 $T = 10$ 期的觀察值作為深度學習模型的輸入：

$$x_t = (u_t, u_{t-1}, u_{t-2}, \dots, u_{t-9})$$

網路架構(network architecture)部分我們採用長短期記憶(long short-term memory, LSTM) [10] 作為提取時間序列特徵的主要元件，並且結合多個近代深度學習常見技巧：

1. dropout [11] 有助於防止過擬合(overfitting)。
2. 層標準化(layer normalization) [12] 有助於加速模型訓練。
3. 殘差連結(residual connection) [13] 解決深度學習中模型層數越深反而越難訓練的問題。

著名的 Transformer 模型 [14] 在自然語言處理領域裡取得了重大的成功。並且它同時也是一種序列處理模型，這與金融時間序列有相似之處。我們在模型架構實作上便是採用 Transformer 的實作來結合了以上三種技術，網路深度為 16 層，每層

特徵維度為 256。²³也就是說，將其中自注意機制(self-attention mechanism)的部分抽換為 LSTM，我們認為這樣更符合金融時間序列的特質。²⁴



4.2 損失函數

承 3.1 節的定義，模型的目標是預測對數報酬率 $r_t \equiv R(x_t, a_t)$ 。同時為了模型訓練與分析方便，我們將投資決策的行動集合 \mathcal{A} 限縮到 \mathcal{A}_{11} ，離散化為 11 種可能性： $\alpha \in \{0, 0.1, 0.2, \dots, 0.9, 1.0\} \equiv \mathcal{A}_{11} \subset \mathcal{A}$ 。在這個框架下，傳統值方法可以被描述為優化 11 個回歸問題，以 $f(x_t, a_t; \theta)$ 估計對數報酬率 r_t ，採取均方誤差為損失函數：

$$L_\alpha(\theta) = E_{e \in \mathbb{D}} \left[(r_t - f(x_t, a_t; \theta))^2 \mid a_t = \alpha \right]$$

在均方誤差損失函數下估計值函數，這種情境 C51 形容其為平均值(mean)。本文借用這個說法，在後續章節中將以縮寫 MRL (mean reinforcement learning)代指傳統值方法、²⁵DRL (distributional reinforcement learning)則代指分布式強化學習方法。

MRL 所對應的策略為：

$$\pi_{\text{MRL}}(x_t) = \operatorname{argmax}_{a \in \mathcal{A}_{11}} f(x_t, a; \theta)$$

分布式強化學習的部分我們採取 C51 的方法論，以長條圖為操作模型參數 θ 所對應的值分布 Z_θ ，將欲估計的資產報酬率 r_t 分為 $C = 21$ 種不同的類別 $z_j = \{-0.025, -0.0225, -0.02, \dots, 0.02, 0.0225, 0.025\}$ ，以交叉熵為損失函數，將其視為 11 個分類問題：

$$L_\alpha(\theta) = E_{e \in \mathbb{D}} \left[- \sum_{j=1}^{21} \left(\delta(r_t, j) \cdot \log(p_j(x_t, a_t; \theta)) \right) \mid a_t = \alpha \right]$$

²³ 在 Transformer 中這個超參數通常稱為 d_model，這裡列出 Pytorch 的說明文件作為參考：<https://pytorch.org/docs/stable/generated/torch.nn.TransformerEncoderLayer.html>

²⁴ 自然語言處理中的句子是序列資料，嚴格來講這不是時間序列資料。它們的共通點僅僅是序列，在程式實作中具備一樣的輸入格式，參考 Transformer 的架構來設計並抽換為 LSTM 這種作法有其便利性。至於自注意機制是不是能成功應用在金融時間序列資料上，我們認為是一個可能的改進方向，有待未來研究。

²⁵ 一般深度強化學習應用研究經常將這種形式簡稱 DQN，然而實際上作為對照組，在這些研究中究竟只是採取 MSE 所以簡稱 DQN，還是確實使用了 DQN 的完整訓練技術經常不得而知。本文並不是採用完整 DQN，因此自創用詞 MRL 區分之。

其中函數 $\delta(y, j)$ 代表數值 y 映射到長條圖的結果，若對應類別 j 則為 1、反之為 0。²⁶

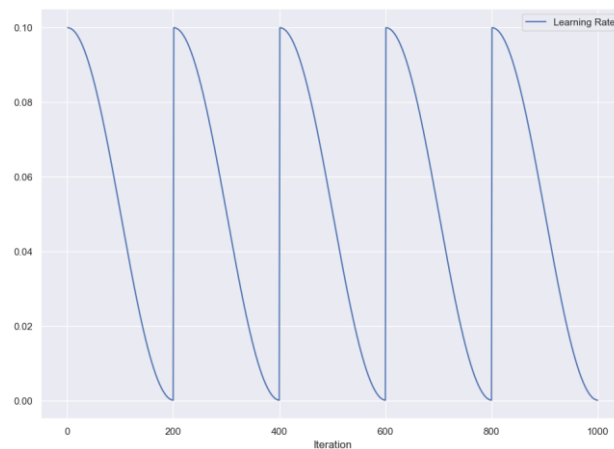
在 DRL 的策略中我們將模型所預測的對數報酬率分布記作 Z_θ ，以該分布期望值作為決策函數 $f(x_t, a_t; \theta) = E[Z_\theta]$ ，在後續章節中也將沿用這套定義。其中期望值的計算方法沿用 2.5.1 小節中 C51 長條圖的概念，整理後得到 DRL 所對應的策略：

$$\pi_{\text{DRL}}(x_t) = \operatorname{argmax}_{a \in \mathcal{A}_{11}} \sum_{j=1}^{21} z_j p_j(x_t, a; \theta) = \operatorname{argmax}_{a \in \mathcal{A}_{11}} E[Z_\theta]$$

然而均方誤損失與交叉熵損失在數值單位上沒有可比性。為了消除這種差異，盡力讓這兩種模型設定相似，我們參考了 RNN 訓練中常用 gradient norm clip 的技巧 [15] 達成穩定性。具體來說，將損失函數值乘以一個非常大的數值例如 10000 倍，再取 norm clip=1.0，相當於強制每個 mini-batch 得出的 gradient norm 都恰好等於 1.0。這樣一來可以保持 MRL 與 DRL 在訓練過程中得出的梯度數量級盡量一致。

4.3 模型訓練

本文採取論文 SGDR (stochastic gradient descent with warm restarts) [16] 提出的訓練方法，以隨機梯度下降(stochastic gradient descent, SGD)來優化模型參數，並且循環式地調整學習率(learning rate)。以本文模型的超參數為例，初始學習率 0.1，每 200 次迭代一個循環，訓練五個循環，總共 1000 次迭代，其學習率變化如下圖：



圖表 4-1 演算法 SGDR 的學習率變化

²⁶ C51 [3] 原始模型對此做了更精細的設定，本文並未採取。有興趣的讀者可以參考其原文關於 categorical algorithm 的部分。

這種演算法有以下好處：

1. 近代許多研究 [17,18] 顯示 SGD 相較 Adam 不容易過擬合。
2. SGDR 的實驗結果顯示，在各循環中採取這種餘弦退火(cosine annealing)形式學習率能幫助模型更快收斂。
3. 每個循環之間採取熱重啟(warm restart)機制，也就是學習率突然上升，這樣做實務上有助於模型最終在測試集上的表現。SGDR 提出一個可能的原因是突然提高的學習率讓參數有機會跳脫局部最優點(local optima)，再次探索其他更好的可能性。

深度強化學習模型參數對權重隨機初始化(random weight initialization)是敏感的，換言之，不同的亂數種子(random seeds)可能導致非常不同的結果。一般來說會取 N 個獨立的亂數種子重複多次實驗，回報其平均值或中位數。Henderson 等人於 2018 年深度強化學習模型實驗可重現性(reproducibility)的研究 [19] 中，針對這個現象做了詳細的討論，他們認為過少的重複實驗 $N < 5$ 可能會產生誤導的結果。

27

為力求實驗結果完整，本研究非常重視這個問題。因此在我們訓練與評估模型的方法裡，對於每個資產對及其投資問題環境 $\mathcal{E}(S_i, S_j)$ 都會取 10 個獨立的亂數種子重複實驗，承 3.2 節定義的 90 組資產對，這代表 MRL 與 DRL 各訓練了 900 個模型。若將資產對順序交換視為等價 $(S_x, S_y) \equiv (S_y, S_x)$ ，也就是以一般強化學習評估的環境來看，本研究相當於從 45 個環境裡分別做了 $N = 20$ 次實驗。

²⁷ 我們認為這是個嚴厲的批評。一般來說常見的設定有 $N = 3$ 或 $N = 5$ ，尤有甚者有些論文的實驗對此隻字未提，很可能是採取 $N = 1$ 。



5 模型分析

5.1 投資組合比較

本節將針對分布式強化學習 DRL 所產生的 900 組實驗結果，與其他方法進行比較。除了傳統值方法 MRL 之外，為了實驗的完整性本研究進一步提供其他參考基準(baseline)。對於投資問題 $\mathcal{E}(A, B)$ 這裡定義一族固定式策略，給定常數 $0 \leq c \leq 1$ ，其對應的固定式策略為： $\pi_c(x_t) = c$ ，也就是說無論金融市場環境如何變化都採取總資金的固定比例 $\alpha_t = c$ 進行再平衡。據此衍生的參考基準如下：


$\pi_{c=1}$ 將所有資金固定投入 A 資產，這相當於買進持有策略(buy and hold)。

$\pi_{c=0.5}$ 將資金平均分配於 A、B 兩資產，也就是各半策略(fifty-fifty)。

	年化報酬率 (平均數)	年化報酬率 (中位數)	夏普率 (平均數)	夏普率 (中位數)
Buy & Hold	0.212661	0.191339	0.730538	0.809365
Fifty-Fifty	0.212661	0.203256	0.864359	0.86354
MRL	0.208529	0.253128	0.747972	0.978461
DRL	0.228168	0.24698	0.867273	0.972809

圖表 5-1 投資組合實驗結果

以平均值作為參考指標，從圖表 5-1(粗體為該欄最大值)可以看出 DRL 在年化報酬率與夏普率的平均數中均勝過所有其他方法。然而平均值易受極端值影響，中位數也僅僅是分位函數(quantile function)的一個特例 $Q(p = 0.5)$ ，僅採取平均數與中位數這兩個常用統計量作為比較基準稍嫌有欠公平。以下我們將以買進持有策略作為參考基準，回報各策略在 100 種不同的分位函數 $\{Q(p = \frac{j}{99}) \mid 0 \leq j \leq 99\}$ 下擊敗買進持有策略的比例。

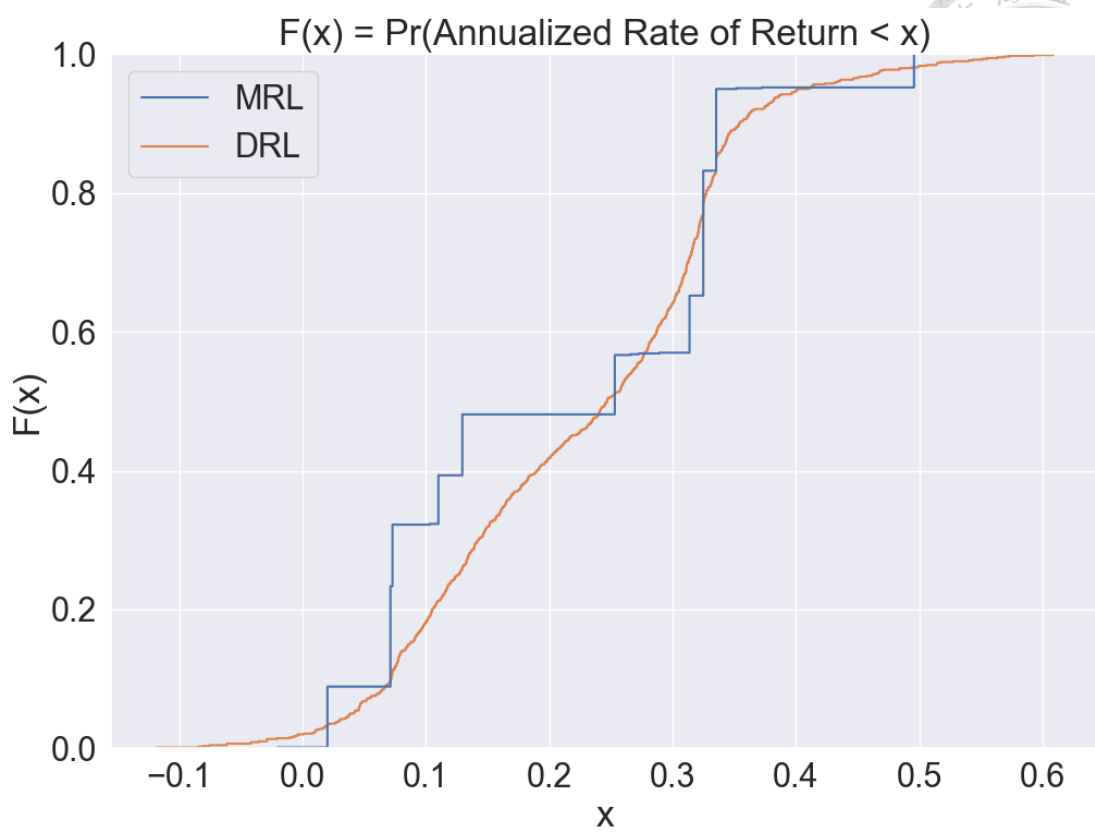


	年化報酬率 (擊敗買進持有策略比率)	夏普率 (擊敗買進持有策略比率)
Fifty-Fifty	50%	82%
MRL	12%	27%
DRL	64%	96%

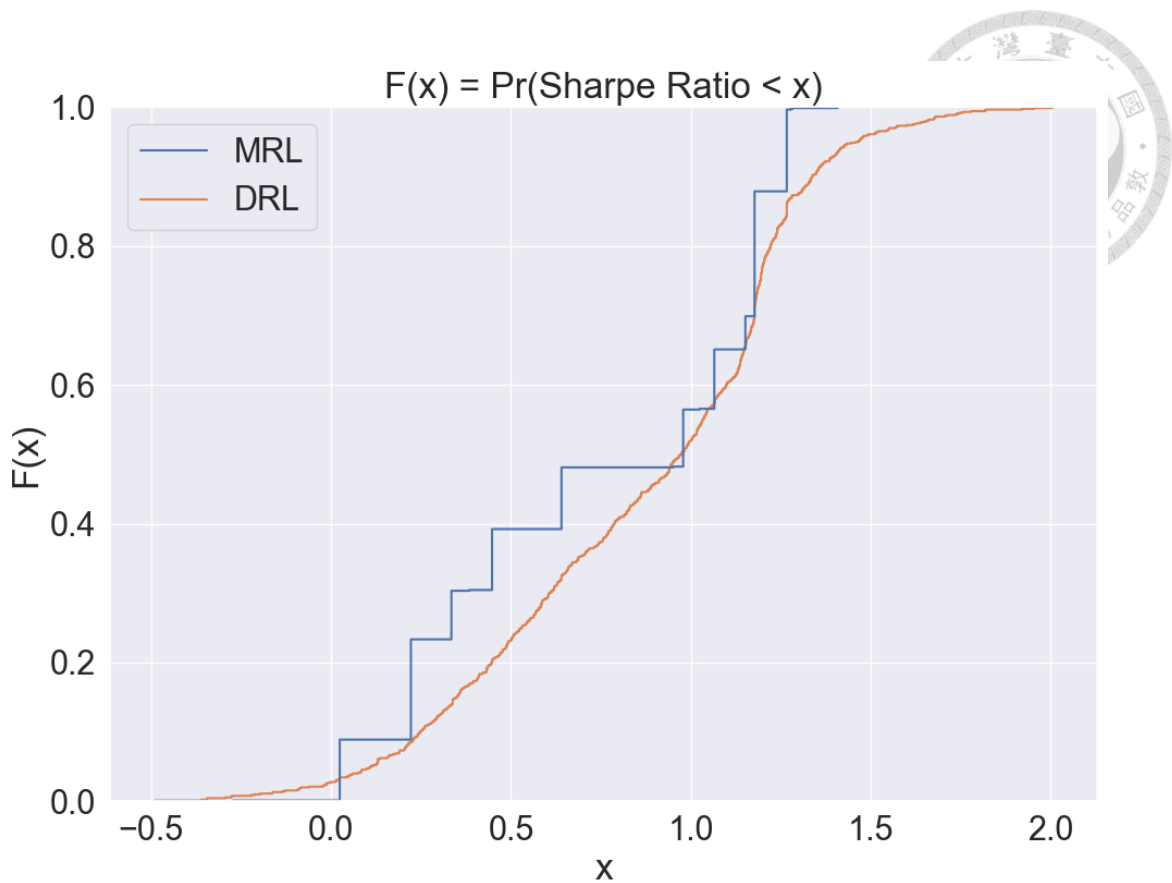
圖表 5-2 投資策略擊敗買進持有策略比例

圖表 5-2(粗體為該欄最大值)報告了各個策略在年化報酬率與夏普率擊敗買進持有策略的比率。年化報酬率方面各半策略擊敗買進持有的比例恰好是 50%，合乎直覺。而 DRL 成功突破了這個基準，在將近三分之二的情形可以勝過買進持有策略。夏普率方面各半策略作為投資學中分析分散風險的常用工具，其表現不俗。然而 DRL 表現更加優異，其擊敗買進持有的情況所占比率將近百分之百。實驗結果顯示分布式強化學習確實是一種有效的投資工具，尤其是其分散風險的能力。

為進一步比較 DRL 與傳統方法 MRL 的績效差異，我們分別對 900 組實驗結果的年化報酬率與夏普率繪製其經驗累積分布函數(empirical cumulative distribution function)圖做為比較工具。在此簡短說明比較方法：以夏普率為例，函數 $F_X(x) = Pr(X < x)$ ，其橫軸代表給定一個參考點 x ，夏普率小於該參考點的機率 $F_X(x)$ 即為縱軸。反過來說給定參考點機率，其橫軸對應到的參考值越大越好，也就是曲線越靠右，夏普率越優秀。



圖表 5-3 分布式強化學習與傳統方法比較(年化報酬率)



圖表 5-4 分布式強化學習與傳統方法比較(夏普率)

從年化報酬率(圖表 5-3)來看 DRL 曲線大部分都落在 MRL 右邊，也就是勝過 MRL。即使是輸給 MRL 的部分其差距也較小，相較於 DRL 贏過 MRL 時所超過的量。

從夏普率(圖表 5-4)來看 DRL 曲線幾乎所有部分都落在 MRL 右邊，可謂完勝 MRL。與前述數值比較(圖表 5-2)的結論類似，DRL 在夏普率的表現特別優異。從經驗累積分布函數圖中更可以看出，分布式強化學習相較傳統方法更擅長分散風險。

本文接下來將深入分析其決策行為，進一步探討分布式強化學習成功的原因。



5.2 強化學習決策分析

分布式強化學習在工程方面具備諸多優點，例如模型容易訓練，這在 C51 與 IQN 兩篇論文 [3,4] 中已經有詳細的研究。本節將透過實驗數據分析，嘗試針對分布式強化學習在投資問題上的經濟意義展開論述。在這之前不免俗地需要先討論經濟學是什麼，我們引述 Lionel Robbins 於 1932 年提出其中一種經濟學的定義 [20]：

經濟學是一門研究人類在有限的資源情況下作出選擇的科學。

在投資問題中什麼是有限的資源？為了輔助說明，我們先解釋無限的資源可能會是什麼樣子。沿用本研究對投資問題的設定 $\mathcal{E}(A, B)$ ，假設存在一個能完美預測未來狀態的真知(oracle)函數 $\Theta: \mathcal{X} \times \mathcal{A} \rightarrow \mathcal{X}$ ，其滿足：

$$Pr(x_{t+1} = \Theta(x_t, a_t)) = 1$$

掌控真知函數將使得獎賞值 $r_t \equiv R(x_t, a_t)$ 不再是隨機變數，而是一個確定的值 $C_t(x_t, a_t, x_{t+1})$ ，也就是未來對數報酬率。其對應的真知策略如下：


$$\begin{aligned} \pi_{\text{oracle}}(x_t) &= \operatorname{argmax}_{a_t \in \mathcal{A}} (E[R(x_t, a_t) | x_{t+1} = \Theta(x_t, a_t)]) \\ &= \operatorname{argmax}_{a_t \in \mathcal{A}} (C_t(x_t, a_t, x_{t+1})) \end{aligned}$$

如果將投資者在金融市場上掌握的資訊多寡類比為資源，那麼掌控了無限資源的 π_{oracle} 其選擇行為顯而易見地只要要在 A 、 B 兩資產中全部投資表現較好的資產即可，²⁸無須分散風險，其具備：

$$\pi_{\text{oracle}}(x_t) \in \{0, 1\}$$

也就是說，就真知策略的觀點而言世界是非黑即白的。那麼實際上在有限的資源下，傳統值方法強化學習與分布式強化學習的觀點分別為何？我們試著根據實驗數據回答這個問題，列出 MRL 與 DRL 兩者分別在 900 次實驗中所選擇 $a_t \in \mathcal{A}_{11}$ 值的比例分析其行為。

²⁸ 排除了瑣碎的情形例如 $A = B$ 。



a_t	MRL	DRL
0.0	50.21%	20.50%
0.1	0.00%	10.03%
0.2	0.00%	5.74%
0.3	0.00%	6.22%
0.4	0.00%	4.74%
0.5	0.00%	5.27%
0.6	0.00%	5.13%
0.7	0.00%	6.09%
0.8	0.00%	4.68%
0.9	0.00%	10.49%
1.0	49.79%	21.12%

圖表 5-5 強化學習決策比率

圖表 5-5(粗體為該列最大值)代表了各方法在 900 次試驗、測試期長 500 個交易日中所做出決策 a_t 的比率，模型決策 $a_t = 1$ 或 $a_t = 0$ 屬於兩個極端，前者代表全部投資 A 資產、後者全部投資 B 資產。觀察這兩列數據發現傳統值方法與真知策略的理論行為極為相似，決定的投資比例非 0 即 1，其觀點也是非黑即白。相比之下分布式方法可能做出折衷的決定，例如 $a_t = 0.3$ ，我們認為這代表 DRL 學會了如何分散風險。

然而傳統值方法具備類似真知策略非黑即白的觀點，從來不做 0 或 1 以外的決定。這種結論顯然並不是因為 MRL 能完美預測未來，我們認為這是 MRL 受制於估計形式導致的結果。

$a_t = 1$ 與 $a_t = 0$ 對 MRL 來說相當於分別估計 A、B 的期望報酬率，對於其他情形 $0 < a_t < 1$ 而言其估計目標僅僅是兩個資產的線性組合，是數個互不相干的數值。這種估計框架下 MRL 無需考慮這兩個資產是否有任何相關性，統計學上視為理所當然，在投資領域中 MRL 非 0 及 1 這種武斷的決定與分散風險這個一般常識背道而馳。

分布式強化學習嘗試刻劃投資組合報酬率的統計分布，這種方法為決策提供更多元的資訊，而非如傳統方法一般僅僅依賴期望值作為參考指標。如同圖表 5-5 實驗數據顯示，DRL 在投資問題中有可能做出介於 0 到 1 的決定，進而在有限的資訊下學習到如何分散風險，而非如同真知策略一般採取簡單武斷的決定。據此我們認為分布式強化學習的經濟意義是：充分利用有限的資訊，進而分散風險，為投資問題提供更合理的決策。



5.3 值分布與風險管理

值方法強化學習的精神是藉由捕捉值函數的性質，例如估計期望值或值分布，形成決策函數，進而持續改善決策函數找出最佳策略。本研究採取的方法與推論皆遵循值方法基礎理論所設計，回顧 4.2 節中我們使用平均決策函數： $f(x_t, a_t; \theta) = E[Z_\theta]$ ，這與一般的分布式強化學習並無二致。

那麼改取平均值以外的統計量會發生什麼事？例如 C21-SR 中便直接將值分布的期望值與標準差相除，希望通過操作值分布 Z_θ 改善夏普率。我們受到 C21-SR 的啟發，本節將展示值分布在風險管理中可以作為一種靈活運用的工具，討論其他跳脫傳統理論框架的可能性。

對於值分布 Z_θ 我們定義其累積分布函數(cumulative distribution function)為 $F(x) \equiv \Pr(Z_\theta \leq x) = p$ ，採用其反函數作為分位函數的定義： $Q(Z_\theta, p) \equiv F^{-1}(p) = x$ 。為了方便計算值分布 Z_θ 的統計量，我們從依據模型中長條圖所對應機率，抽 1000 個樣本點對統計量進行抽樣估計。²⁹以下我們嘗試探討兩類不同的風險管理情境。

首先，評估一個投資組合的風險可以採取夏普率以外的指標，例如風險值(VaR, value at risk) [5]。假設投資組合日報酬率為 Z ，管理目標是希望風險值 $\text{VaR}_\alpha(Z)$ 越小越好。為了實驗這種情境，這裡取其常用參數 $\alpha = 0.05$ ，在本文的符號裡等價於希望分位函數值 $Q(Z_\theta, 0.05)$ 越大越好。另外一種類似風險值的情境是條件風險值(CVaR, conditional value at risk) [21]，目標是希望當損失超過 $\text{VaR}_\alpha(Z)$ 的時候，其條件期望損失越小越好。

其次，不同的投資者對風險的態度可能不同。一般來說我們認為風險是不好的，例如現代投資組合理論(modern portfolio theory)的創始者 Markowitz 於 1952 年發表的知名著作 [22] 中便是如此描述。這裡展示透過操作值分布 Z_θ 去嘗試討論積極樂觀的投資態度，有時甚至接近風險追求(risk-seeking)的態度是可行的。我們試圖描繪一類樂觀投資者的行為，其積極地追求更高的投資報酬率，願意承擔更多風險。在本節中設定他們的決策標準是希望分位函數值 $Q(Z_\theta, 0.95)$ 越大越好，命名為 DRL-Optimist 組，與風險值的情境相對，樂觀投資者關注的是未來不確定性中光明的一面。

²⁹ 在本節中只有 DRL 組維持 4.2 節的計算方式，也就是說其實驗數據將與 5.1 節完全一致。

同時我們也模仿 C21-SR [9] 的方法，在本研究架構下設計了相應的 DRL-SR 組別，以便探討其績效為何。實驗組別名稱與具體的決策函數說明如圖表 5-6。

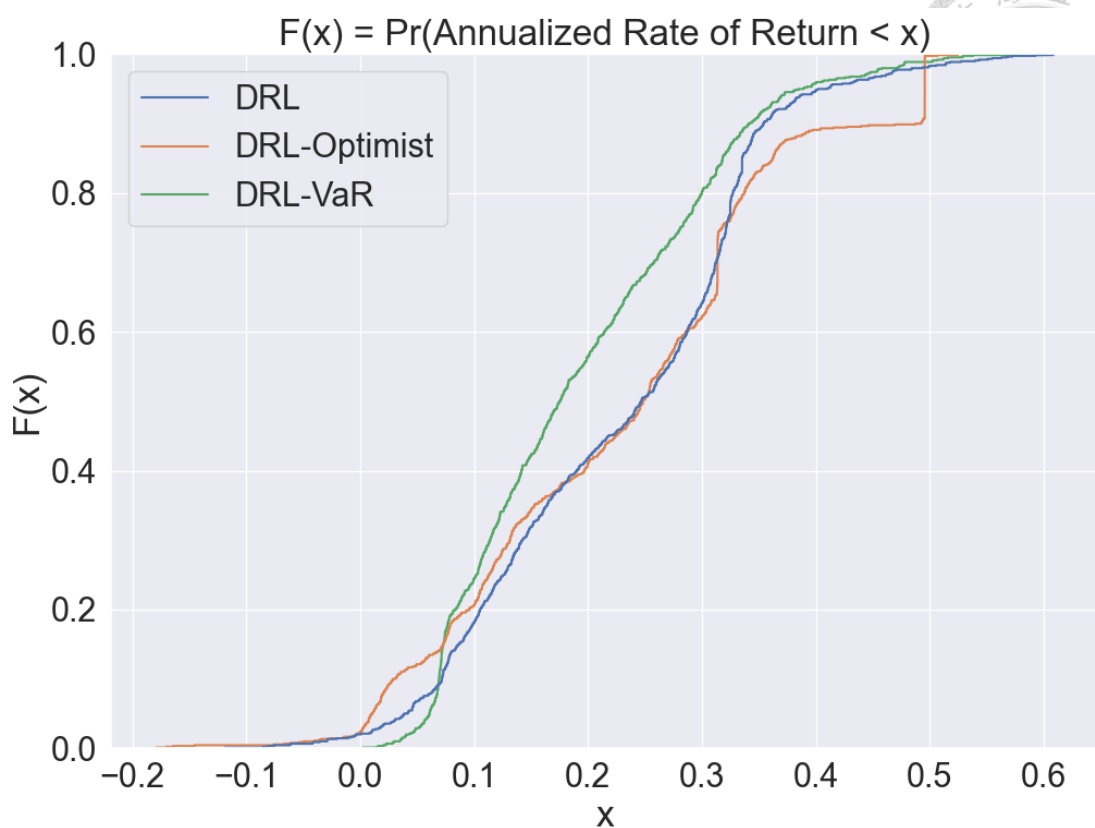
組別名稱	決策函數
DRL	$E[Z_\theta]$
DRL-SR	$\frac{E[Z_\theta]}{\sigma[Z_\theta]}$
DRL-Median	$Q(Z_\theta, 0.5)$
DRL-Optimist	$Q(Z_\theta, 0.95)$
DRL-VaR	$Q(Z_\theta, 0.05)$
DRL-CVaR	$E[Z_\theta Z_\theta < Q(Z_\theta, 0.05)]$

圖表 5-6 風險實驗組別與決策函數

組別名稱	年化報酬率 (平均數)	年化報酬率 (中位數)	夏普率 (平均數)	夏普率 (中位數)
DRL	0.228168	0.246980	0.867273	0.972809
DRL-SR	0.209038	0.214106	0.808312	0.850926
DRL-Median	0.216683	0.227006	0.771584	0.902137
DRL-Optimist	0.231689	0.248181	0.744598	0.912493
DRL-VaR	0.195215	0.175324	0.808310	0.770366
DRL-CVaR	0.193391	0.191896	0.816053	0.795588

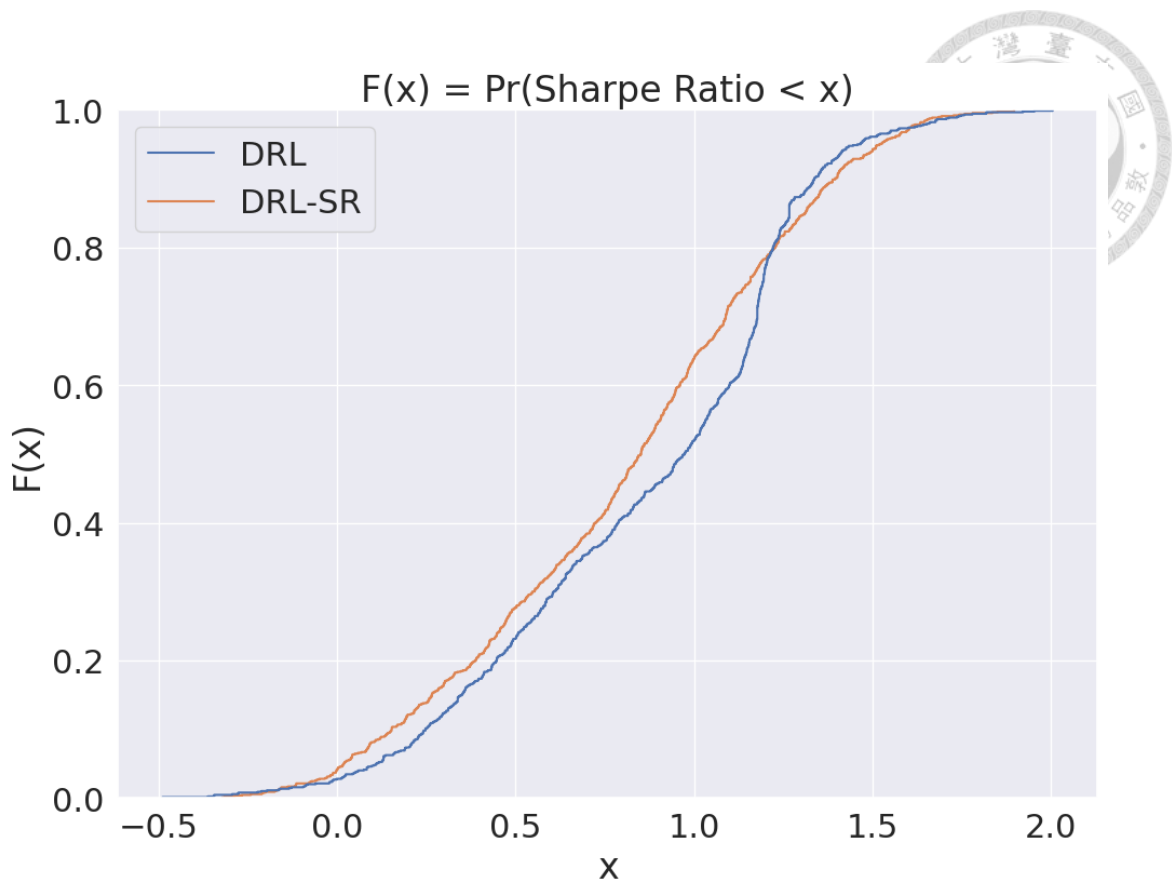
圖表 5-7 不同風險決策指標下的分布式強化學習投資組合

從實驗數據圖表 5-7(粗體為該欄最大值)可以看出 DRL-Optimist 組的年化報酬率勝過其他組別，然而夏普率表現卻較差，尤其平均夏普率是所有組別中最低的。這部分結果與高報酬伴隨高風險的理財常識相符。



圖表 5-8 不同風險決策指標所對應年化報酬率

我們針對年化報酬率進一步繪製經驗累積分布函數圖進行比較。從圖表 5-8 可以看出 DRL-Optimist 組大部分情況均勝過其他組別，例如 DRL-VaR 組。DRL-Optimist 組與 DRL 組大部分時候表現接近，有時能大幅贏過 DRL 組，例如圖中右上角部分。其原因雖不清楚，我們認為這在未來有進一步討論的空間。



圖表 5-9 不同風險決策指標所對應夏普率

夏普率的部分我們在本節中試圖檢驗 C21-SR [9] 提出的方法效果如何。其中圖表 5-7 的數據顯示 DRL-SR 的四項指標皆輸給 DRL。進一步觀察夏普率對應的經驗累積分布函數，也就是圖表 5-9，DRL-SR 組的曲線幾乎都落在 DRL 組曲線的左邊，在夏普率方面明顯不如正常 DRL 方法。

參考論文 [9] 的實驗結果圖表 2-1，C21-SR 組別對應本研究類似情境的 DRL-SR 組別，C21 對應本研究 DRL 組別，根據我們的實驗結果顯示，一個單純的分布式強化學習演算法，即便不採取類似 C21-SR 獨特的決策函數形式，也能為夏普率帶來助益，在本研究中 DRL 組甚至較 DRL-SR 組表現更佳。這顯示沒有值方法理論基礎的模型應用在投資問題上效果如何，需要更進一步大規模地仔細檢驗。



6 結論

6.1 研究結果

1. 實驗結果顯示，分布式強化學習在投資問題中，不論年化報酬率或是評估風險的重要指標夏普率，結果皆優於傳統值方法強化學習。其中夏普率的表現尤其突出，更突顯分布式強化學習在控制風險方面的優勢。
2. 從經濟學的角度，我們為分布式強化學習的成功提出一種可能的原因：它是一種充分利用有限資訊在投資問題中分散風險的工具。
3. 本文展示了分布式強化學習做為一種分析工具，在不同的風險偏好下為投資決策帶來更靈活的應用。

6.2 未來展望與建議

本研究指出了值分布強化學習方法在投資問題上確實能帶來好處。接下來基於我們的方法論，對投資感興趣的研究者可能會問兩個問題：

1. 實務應用上，如何提升模型效能，進一步為投資帶來更多更穩健的獲利？
2. 深度學習模型是否能為經濟學提供更深入的洞見？

本節針對實務與理論這兩個面向，給出未來展望與建議。

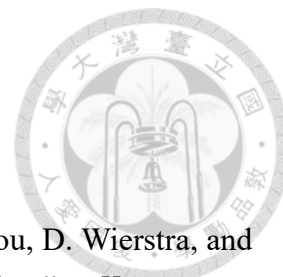
6.2.1 實務方面

1. 更好的值分布估計方法：在分布式強化學習中，本文採取的是最具代表性的研究：DeepMind 於 2017 年發表的 C51 模型 [3]。然而 DeepMind 團隊其後續提出了 QR-DQN 模型 [23]，以及 2018 年 IQN 模型 [4]，都在 Atari 小遊戲的表現上得到顯著的進步。我們認為將其實作在投資問題上很可能取得更好的效果。重視實作細節的研究者更可以進一步嘗試改進版的 IQN 模型，例如 Microsoft 2019 年提出的 FQF 模型 [24]。
2. 更豐富的輸入資訊：在股票訓練資料方面，本研究為求模型精煉，僅採取最基本的歷史價格作為特徵，甚至不包含成交量、常用技術分析指標等。每個獨立的投資實驗 $\mathcal{E}(A, B)$ 中除了 A 、 B 兩資產外別無其他資訊，以台灣股票市場來說，有興趣的研究者可以嘗試整合其他特徵，例如大盤走勢、美股走勢、個股財報資料甚至新聞資料等。我們預期更豐富的資訊將能大幅改善模型的投資表現。

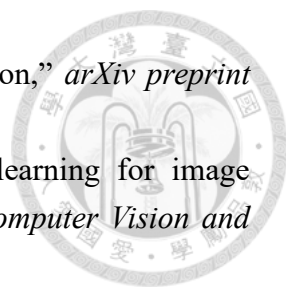
6.2.2 理論方面

我們嘗試將強化學習投資問題與估計資產報酬率相結合，在本研究所設計的架構下，分布式強化學習所估計的值分布恰好對應估計未來資產報酬率分布。過去財金領域的研究者們出於風險管理的目標，會試圖建立模型估計一個資產或投資組合的統計分布，以利為後續分析提供有用的參考指標，例如預估風險值。我們認為這與分布式強化學習為值分布提供有效估計的目標不謀而合。本研究的主旨是說明預測值分布能為投資問題帶來助益，並且在 5.2 節針對其經濟意義給出了一種可能的解釋。然而結合深度學習技術以及金融理論是可能的，透過進一步分析值分布的特性，將能更好地掌握一個資產或投資組合的行為，為理論分析帶來更深入的洞見。我們認為這在未來是一個值得研究的方向。

參考文獻



- [1] Mnih, V., K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.
- [2] Mnih, V., K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, and G. Ostrovski, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
- [3] Bellemare, M. G., W. Dabney, and R. Munos, "A distributional perspective on reinforcement learning," in *International Conference on Machine Learning*, 2017, pp. 449–458.
- [4] Dabney, W., G. Ostrovski, D. Silver, and R. Munos, "Implicit quantile networks for distributional reinforcement learning," in *International Conference on Machine Learning*, 2018, pp. 1096–1105.
- [5] Artzner, P., F. Delbaen, J. M. Eber, and D. Heath, "Coherent measures of risk," *Mathematical Finance*, vol. 9, no. 3, pp. 203–228, 1999.
- [6] Silver, D., A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, and M. Lanctot, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.
- [7] Barth-Maron, G., M. W. Hoffman, D. Budden, W. Dabney, D. Horgan, D. Tb, A. Muldal, N. Heess, and T. Lillicrap, "Distributed distributional deterministic policy gradients," *arXiv preprint arXiv:1804.08617*, 2018.
- [8] Van Hasselt, H., A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016, vol. 30, no. 1, pp. 2094–2100.
- [9] Harnpadungkij, T., W. Chaisangmongkon, and P. Phunchongharn, "Risk-sensitive portfolio management by using distributional reinforcement learning," in *2019 IEEE 10th International Conference on Awareness Science and Technology (iCAST)*, 2019, pp. 1–6.
- [10] Hochreiter, S. and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [11] Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

- 
- [12] Ba, J. L., J. R. Kiros, and G. E. Hinton, “Layer normalization,” *arXiv preprint arXiv:1607.06450*, 2016.
 - [13] He, K., X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.
 - [14] Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in Neural Information Processing Systems*, vol. 30, pp. 5998–6008, 2017.
 - [15] Pascanu, R., T. Mikolov, and Y. Bengio, “On the difficulty of training recurrent neural networks,” in *International Conference on Machine Learning*, 2013, pp. 1310–1318.
 - [16] Loshchilov, I. and F. Hutter, “SGDR: Stochastic gradient descent with warm restarts,” *arXiv preprint arXiv:1608.03983*, 2016.
 - [17] Keskar, N. S. and R. Socher, “Improving generalization performance by switching from Adam to SGD,” *arXiv preprint arXiv:1712.07628*, 2017.
 - [18] Zhou, P., J. Feng, C. Ma, C. Xiong, and S. C. H. Hoi, “Towards theoretically understanding why SGD generalizes better than Adam in deep learning,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 21285–21296, 2020.
 - [19] Henderson, P., R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, “Deep reinforcement learning that matters,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018, vol. 32, no. 1, pp. 3207–3214.
 - [20] Robbins, L., *An Essay on the Nature and Significance of Economic Science*. London: Macmillan, 1932.
 - [21] Rockafellar, R. T. and S. Uryasev, “Optimization of conditional Value-at-Risk,” *Journal of Risk*, vol. 2, pp. 21–42, 2000.
 - [22] Markowitz, H., “Portfolio selection,” *Journal of Finance*, vol. 7, no. 1, pp. 77–91, 1952.
 - [23] Dabney, W., M. Rowland, M. Bellemare, and R. Munos, “Distributional reinforcement learning with quantile regression,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018, vol. 32, no. 1, pp. 2892–2901.
 - [24] Yang, D., L. Zhao, Z. Lin, T. Qin, J. Bian, and T.-Y. Liu, “Fully parameterized quantile function for distributional reinforcement learning,” *Advances in Neural Information Processing Systems*, vol. 32, pp. 6190–6199, 2019.