國立政治大學社會科學學院經濟學系

碩士學位論文

應用強化學習於股票的投資選擇

-以台灣股市為例

Applying Reinforcement Learning to Stock Investment

– Taiwan Stock Market as an Example

指導教授：蕭明福 博士和蔡瑞煌 博士

研究生：彭志偉 撰

中華民國 一一〇 年 七 月

DOI:10.6814/NCCU202100964

摘要

強化學習在各領域都是一門不可或缺的學科，而在金融界的實際應用已有信用借貸/違約評估、風險控管、人工智慧客服及股市預測等等，金融科技則是運用數學模型來解決金融環境中的問題，本研究將應用強化學習演算法的學習框架套用於臺灣股票金融市場環境當中，設計一個股票投資的學習環境並模擬投資人在該環境中進行演算法超參數調整的實驗，代理人的最終目的在於控制投資風險的情況下將投資報酬最大化，本研究採用已上市達 21 年，且為臺灣股市總市值前 15 大之股票作為強化學習之環境模擬的訓練對象，使用 2000 年至 2016 年的股票歷史資料作為訓練數據資料集來進行訓練，2017 年至 2021 年作為測試資料集，最後本研究將評估其實驗結果及跟其他的投資績效策略進行投資報酬績效的比較。

本研究在強化學習框架中所訓練之智慧代理人在環境模擬訓練的過程中，智慧代理人透過模擬學習在一定程度上捕捉到股票市場上股票價格的變動，並且藉由訓練達到有效的自我提升，在其後介紹的實驗測試結果中將會詳細介紹。而研究結果顯示，部分實驗測試的成果比加權股票指數及隨機分配投資策略的績效要好，在經過超參數調參後，仍以本研究之實驗二的成果為最佳選擇，並在測試結果中發現代理人在訓練的過程中有效的學習到了在控制投資風險的情況下進行投資獲利。

關鍵詞：金融股票市場、機器學習、強化學習、神經網路、股票選擇

DOI:10.6814/NCCU202100964

Abstract

Reinforcement learning is an indispensable subject in various fields, and the practical applications in the financial sector include credit lending, default assessment, risk control, artificial intelligence customer service, stock market forecasting, etc., and financial technology uses mathematical tools to explain the problems of the financial environment, this research will apply the learning framework of reinforcement learning algorithm to the Taiwan stock financial market environment, design a stock investment learning environment and simulate the experiment of investors in the environment to adjust the hyper parameters of the algorithm, and the ultimate purpose of the reinforcement learning's agent is putting effort on learning to minimize investment risks and maximize investment returns. The total time data set in this study is 21 years long, and the stock history data from year 2000 to 2016 is used as the training data set for training, from year 2017 to 2021 will be treated as a test data set. Finally, this research will evaluate its experimental results and compare its return on investment performance with other investment performance strategies.

In the process of environmental simulation training, the intelligent agent trained in this research in the framework of reinforcement learning is able to acquire the stock's price movement that changes in the stock market in a certain extent and can achieve effective self-improvement. In experiments two, five and ten The results of the test are better than the weighted stock price index and random allocation of investment strategies. In the test results of the experiments, that is found the agent is able to learn to make investment profits while controlling investment risks during the training process.

Keywords: Stock Market, Machine Learning, Reinforcement Learning, Neural Networks, Stock Selection

DOI:10.6814/NCCU202100964

目　次

DOI:10.6814/NCCU202100964

DOI:10.6814/NCCU202100964

表　次

DOI:10.6814/NCCU202100964

圖　次

DOI:10.6814/NCCU202100964

第一章　緒論

第一節 研究背景

隨著電腦運算技術的快速進步和大數據時代的來臨，現代社會已經出現越來越多方面都結合了強化學習與人工智慧，在各領域都是不可或缺的，而在金融界的實際應用已有信用借貸/違約評估、風險控管、人工智慧客服及股市預測等等，而金融科技則是運用數學模型來解決金融環境中的問題，對於金融環境的困境賦予創新式的解決方案，其中除了運用到數學外，還需要電腦科學、統計和經濟學理論等等的數學工具，其應用範圍廣泛包括風險控管、財務分析及收購案等等，此外不管是商業銀行、證券交易所還是保險公司等機構都會以金融科技的創新方法來進行模擬該情景的模型或項目研發及風險評估等等。

本文發掘新的科學理論應用於股市中的投資選擇，並期望能真正具有一定程度的可行性與有效性，再進一步完善此種投資模型，本研究在模型的建構將以強化學習為理論基礎，做出在股票市場的投資決策模型，且透過本論文的實驗以達到應用強化學習與深度學習結合去擬合投資策略之目的，探討人工智慧在金融投資領域的應用。本研究於實驗中設立前提假設即投資者在整個投資期間擁有足夠的資金進行投資，實驗會透過建構並訓練代理人的方式來對股票市場做出投資決策，期望能藉此有效的分散風險，更能成功於金融股票市場中獲取長期且穩定的投資回報，把收益的波動變化降到最低。此外，本研究將進行反複實驗(Trial and Error)來驗證演算法當中的可行性及有效性，以此期望提升投資者的利潤。

諾貝爾獎得主馬可維茲(Harry Markowitz) 於 1952 年發表了《Portfolio Selection》，對現代投資組合理論 (MPT，Modern portfolio theory)作出了很大的貢獻，奠基了資產配置的發展，其中該學者提出的 MVO 模型(Mean-

DOI:10.6814/NCCU202100964

Variance Optimization)，根據投資者的投資偏好，有兩個主要的考量因素為投資組合的預期收益及其風險，馬可維茲認為收益與風險是正相關的關係，即風險越高其投資收益則越大，故馬可維茲致力於衡量投資組合中的風險和收益，以期望達到收益最大化及風險最小化。馬可維茲根據過往的股票歷史資料，投入要素為投資組合中的期望報酬(Expected Return)及其變異數(Variance)，來衡量投資組合的收益於風險之間關係，最小變異數投資組合能衡量每個資產之間它們的相關性，降低投資組合中的標準差用以降低其投資風險，期望獲得最高報酬，進行分配其權重以降低投資組合的風險，來推算出最佳投資組合的權重，對於投資風險的部分，本研究將在第三章實驗設計中接受其回合獎勵函數，其中運用了最大回撤率來降低其投資收益風險，其描述的是在虧損最大的情況下反映其投資策略的損失程度。

Moody et al. (2001)提出了兩種不同的交易系統，分別在外匯市場及股票市場進行實驗，第一個實驗為應用 RRL[1]於外匯場域進行交易，使用 1996 年前 8 個月的美元/英鎊(USD/GBP)的匯率數據，資料集當中的 80%為訓練資料集和 20%為測試資料集，離散動作(discrete action)為選擇做多(long)、賣空(short)或觀望(neutral)的交易信號，其中在匯率交易系統的買賣價格(bid/ask price)將作為其手續費，獎勵函數為使用可微分夏普率(Differential Sharpe ratio)。第二個實驗為分別應用 RRL 及 Q-learn 演算法於美國標準普爾 500 指數(S&P 500 Index)和美國國庫券 (Treasury Bills)的資產配置，代理人可選擇買多或做空標普 500 指數，其中每當做空標普 500 指數就會賺取美國國庫券的利息，資料集使用 1950 年至 1994 年的月資料，前 20 年為訓練資料集及後 25 年為測試資料集， 0.05%的手續費將在每次交易中產生，其中 Q-learn 使用兩層的 FFN (FeedForward Network)，獎勵函數為下降偏差比率(Downside Deviation ratio)[2]。學者分別應用兩種不同的強化學習演算法並使用做多或賣空作為離散動作，並對 RRL、Q-learn 模型及買入並持有策略(buy-and-hold strategy)做一個績效比較，其結果顯示 RRL 的強化學習方法的報酬

註[1] 循環強化學習模型(RRL, Recurrent Reinforcement Learning)為 Moody et al. (1997)所提出。

註[2] 下降偏差比率(Downside Deviation ratio)為 Moody et al.(1999)所提出，它類似於夏普率變形的索提諾比率(Sortino ratio)，主要功能為控制投資利潤下行波動的風險。

DOI:10.6814/NCCU202100964

率顯著優於 Q-learn 的學習方法及其他策略，但其缺點是僅考慮單一資產進行實驗，而本研究則應用同時存在 Actor 跟 Critic 的演算法並針對其環境進行策略搜索，其演算法細節將在第二章第一節中介紹，本實驗對台灣股票進行篩選後得到的 15 檔股票對此進行實驗，同時使用連續動作(Continuous Action)作為資產在每一期的權重配置。

Gabrielsson et al. (2015)參考了 Moody et al. (1997)所提出基於策略(policy-based)的循環強化學習模型(RRL, Recurrent Reinforcement Learning)，期望建立降低投資成本及投資風險的交易策略，觀察其長期投資報酬及週期性交易訊號用來作為交易系統，學者利用循環強化學習模型與 Nison(1991)所提出的日本燭台(Candlesticks)作為技術指標來做一個結合，並應用在美國標準普爾 500 指數的期貨市場，以每分鐘作為時間單位，在強化學習上建立高頻交易的策略算法，至於缺失值則利用前一期的數據來填補。學者使用報酬率及夏普比率作為該實驗的績效基準，並分別在含有手續費及無手續費的情況下，跟不使用日本燭台的一般循環強化學習模型、投資並且持有的策略(buy&hold strategy)以及隨機投資策略來做一個績效比較。其結果發現，當不考慮手續費的情況下，運用日本燭台的強化學習模型明顯優於一般模型，但當包含手續費的時候，其模型沒有特別優勢且報酬率為零。本研究雖沒有像 Gabrielsson et al. (2015)學者使用日本燭台，日本燭台的原理是由該檔股票當期的最高、最低、開盤價及收盤價所組成，由於月報酬率只反映了當月月初及月尾之間的差距，並無法呈現該檔股票股價在現實中的真實趨向，為了考慮投資股票市場的真實狀況，在模擬投資人進行投資情境時，本研究在資料集中將使用月平均值(見式 3.4)的原理應於實驗環境中。此外，本文將對臺灣股票進行篩選後得到多檔股票並對此進行實驗訓練，並同時使用連續動作(Continuous Action)作為資產在每一期的權重配置，本研究使用複合年均增長率(CAGR)、夏普比率及 Calmar 比率(見式 3.7)作為績效基準。

Pendharkar et al. (2018) 應用強化學習於個人退休類型的投資組合，專注於調整兩種資產的權重以求取最大報酬率，分別對不同的演算法(SARSA 和 Q-learning)、不同的獎勵函數(Reward function)及動作類別(離散及連續動作)進行實驗，作者的投資標的為美國的標準普爾 500 指數(S&P 500 Index)以及美

DOI:10.6814/NCCU202100964

國的巴克萊資本綜合債券指數(AGG)或者是美國十年期國債，讓代理人在兩種投資商品上做權重配置的訓練，分別以季度、半年度及年度為時間單位。在訓練過程中發現以Q-learning演算法的學習規律，將無法在下一期投資做到獎勵最大化，在實驗結果發現，使用連續動作為輸出所獲的報酬率比離散動作高，離散動作無法將投資報酬最大化。一般的實驗在不考慮手續費的情況下，高頻交易的報酬率應該會比低頻交易高，但在學者的實驗中卻發現年交易比每季或半年交易一次的總計報酬率來得高，其意味著以金融市場作為實驗環境的模型即使使用大量的數據也並不代表能有效的訓練模型。本文並沒有使用投資金額來衡量投資所帶來的虧盈狀況，而是像 Pendharkar et al. (2018)那樣使用概率百分比來作為本研究的衡量方式(見第四章第二節)，本研究將使用 Calmar ratio 作為回合獎勵函數及月資料作為時間單位，使用低頻的交易規律可以避免因交易頻率提高而導致交易成本隨之增加的局面。

Kanwar(2019) 應用無模型(Model-free)的強化學習在股票市場，從美國股票市場指定選取出十檔股票，讓代理人在每一期作出投資 M 檔股票的權重配置的決策，用來作為管理財務組合，以獲取長期回報為目的，其回合獎勵為報酬率，以期望在訓練過程中獲得最大報酬率，學者使用 1986 年至 2010 年作為訓練資料集，2010 年至 2019 年作為測試資料集，應用 DDPG(Deep Deterministic Policy Gradients) 演算法，分別測試前饋類神經網路(Feedforward Neural Networks，FNN)和卷積神經網路(Convolutional Neural Network，CNN)的神經網絡，前者當中含有兩層的隱藏層而後者有三層 Conv(Convolution layer)，在最後一層(Output layer)皆使用 Softmax 作為輸出，然後檢測在不同學習率(Learning Rate)的情況下所獲得的報酬。學者在實驗中發現應用強化學習在股票市場環境，其實驗結果的損失值(Loss)跟獎勵值(Reward)是不相關的，損失值越低其獎勵值並不會越高，另外，實驗結果發現在不同的學習率之下 FNN 的夏普比率(Sharpe ratio)跟 CNN 的結果不相上下。本研究應用的是作為 DDPG 的一個延伸的 TD3 演算法，本研究的資料來源將會從臺灣股票市場指定選取出十五檔股票(見第三章第二節)，在動作輸出的設定中共有 15 個動作$A_{i,t}$(見第三章第一節)，其中將代理人的動作設定為在每一期的股票當中選取 6 檔股票作為當期投資股票權重。

DOI:10.6814/NCCU202100964

第二節 研究動機

本研究將討論如何應用強化學習在投資選擇的問題，在應用強化學習(Reinforcement Learning)作為模型基礎於股票市場之前，已有不少研究員已經在應用機器學習(Machine Learning)或深度學習(Deep Learning)於股票市場，應用機器學習於預測股票市場的方法可以分為兩類，過去已有許多學者設計了各種股票市場預測問題的模型分析的方案，機器學習的應用如監督式學習(Supervised learning)的決策樹(Decision Tree)、支援向量機 (Support Vector Machines)、XGBoost(Extreme Gradient Boosting)等演算法，皆可以利用分類分群的方式來預測股價漲跌，對於無監督式學習有助於識別股票市場等不相關數據集中的相關性，許多學者利用無監督式學習的方法對於股票進行預測，這些已發表的作品大多針對海外市場，例如 Powell et al. (2008) 分別使用 無監督技術的 k 均值聚類 K-means 和監督技術的支持向量機（SVM）將 S&P 500 股票每週分為上漲或下跌的價格並且對此進行比較及解釋了兩者之間的差異，對於未來價格的漲跌幅無法準確有效的進行預測，股票價格數據資料會隨著時間而變動，我們很難從舊的數據中預測未來的行為，數據樣本的選擇也會影響到其訓練結果。

在 90 年代有關於神經網絡的部分，人工智能(A.I.)技術常用於股票市場分析等非線性模式分析，多項研究文獻表明，許多學者在研究早期將研究方法側重於人工神經網絡（Artificial Neural Network，ANN）在股票市場預測中的應用，雖然大多數研究主要集中在時間序列數據的預測上，這反映出由於巨大的噪聲和數據的非平穩性致使無法表現出色的預測準確性。H. Ahmadi(1990)嘗試利用人工神經網絡對套利定價理論(Arbitrage pricing theory，APT)進行測試，套利定價理論為金融股票市場在傳統的資產定價模型方面提供了一種可替代的方案，在早期大部分的研究文獻中都使用因素分析 (Factor analysis)的統計方法來測試 APT 模型，在文獻當中，學者使用梯度下降學習規則的反向傳播神經網絡來學習股票收益與股票市場相關因素之間的關係。深度學習會利用神經

DOI:10.6814/NCCU202100964

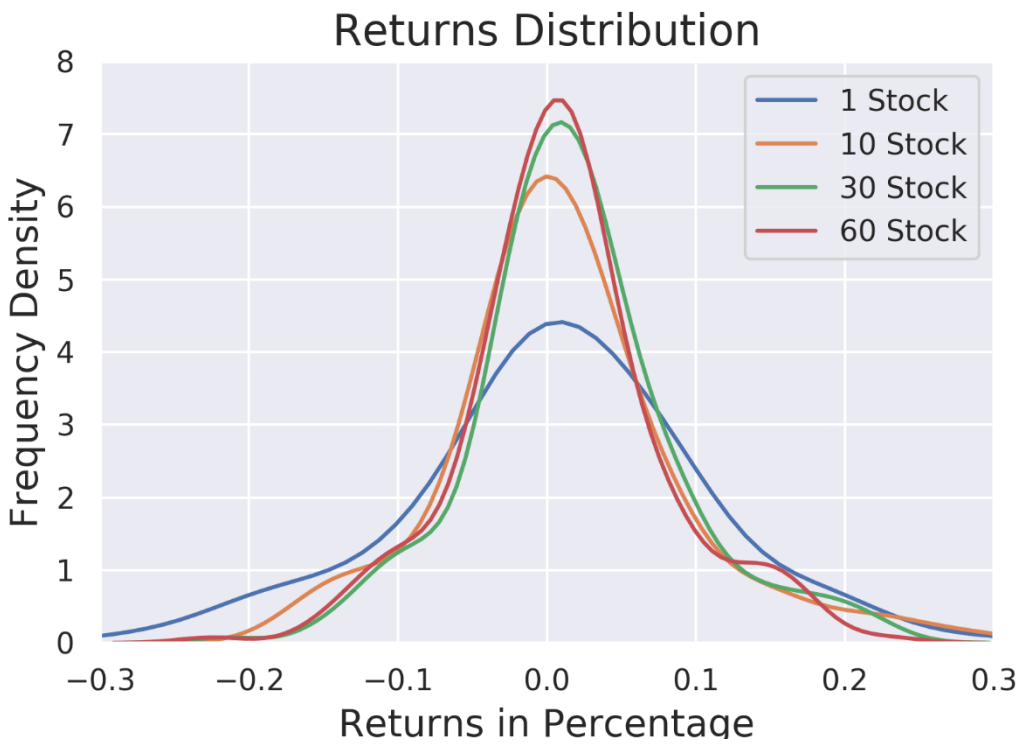
網絡的優勢，得以在股票市場擬合最佳的預測值，深度學習的實際應用較常見的有 LSTM(Long short-term memory)和 GRU(Gated Recurrent Unit)等時間序列(Time Series)相關的演算法，也有研究員應用擁有圖像辨識能力的卷積神經網路(Convolutional neural network)透過股票的趨勢圖來預測股價，例如 Chang(2018)應用神經網絡於上海綜指及深圳成指上，藉著移動平均、相對強弱指標及隨機指標等技術指標來判斷股價的進場及出場的時機點來預測未來一年的股價走勢。劉俞含(2018)使用 XGBoost 模型利用總體經濟、技術指標及國際股市作為特徵來尋找股價上漲的訊號，並分別對台灣加權股價指數、日經 225 指數及標準普爾 500 指數進行股價之預測。

Pendharkar et al.（2018）使用報酬率作為其資料輸入，但由於使用一般的報酬率進行觀察無法讓投資人在真實情景中進行應用，因為其公式為開盤價跟收盤價格之間的差距，當時間軸為一個月甚至更長的時候，譬如在開盤時波動幅度很大，但在收盤時幅度卻變得趨緩，那麼觀察報酬率就無法呈現當月的真實狀況，故此本文將使用月平均值為狀態值，可以顯示當月的平均水平，雖然月平均值會比報酬值來的小，但月平均值可以分辨出正負數，以及在跟其他績效策略比較時顯示出他們之間的差距。在回合獎勵的部分，Kanwar（2019）報酬率作為其回合獎勵，蔡岳霖（2013）以 40 年的台灣加權指數作為樣本，並針對不同的績效指標(其中包含Calmar比率、夏普比率、報酬率等績效指標）進行研究與比較，其研究指出 Calmar 比率的分析結果顯然優於其他績效指標，因此本文將參考蔡岳霖（2013）的研究成果選擇 Calmar 比率作為本文的回合獎勵，Calmar 比率考慮到了投資者的投資風險，即最高獲利與最大虧損之間的差距，使用 Calmar 比率期望代理人在訓練的時候可以在控制投資風險的情況下，最求獲利的目的。

DOI:10.6814/NCCU202100964

第三節 研究目的

本研究將應用強化學習與深度學習結合去擬合投資決策，並期望能研究出能獲得利潤且未來可實際運行的 A.I.（Artificial Intelligence）投資智能體，讓藉由代理人在股票市場從中自行學習以達到投資獲利的投資方式。從上圖一可以看到當投資者持有更多支股票的時候，其收益分佈會逐漸變得穩定，因此相較於持有單一資產，投資者可以嘗試持有多元化的資產以組成一個投資組，得以分散投資風險，因此本研究之實驗將以臺灣股票市場為研究對象，從兩千多支股票當中挑選出數檔股票，然後使用強化學習來訓練模型，代理人每一期將會從中選擇投資若干支股票，以期望達到收益最大化。



圖一：組合的股票越多就越有較穩固的投資報酬

DOI:10.6814/NCCU202100964

本研究將從奇摩財經(Yahoo 奇摩股市)、TEJ+台灣經濟新報資料庫及其他網站收集相應股票資訊之資料，並進行資料預處理 (Data pre-processing)，設計強化學習代理人及其學習環境框架，應用強化學習演算法於臺灣股票金融市場環境中。本實驗的模型將以每月為時間單位，資料集的總時間長度為 21 年，訓練集 17 年 (2000 年 1 月 至 2016 年 12 月)，以及測試集 4 年 (2017 年 1 月 至 2020 年 12 月)，除了運用股價資料外，本研究會將股票資料轉換為月平均值(詳見本文第三章實驗設計)，這樣能在模擬投資人進行投資的情境進行訓練，同時也使得訓練環境更加的合理化，在圖一可看出在持有多檔股票比持有單一股票的投資報酬相較為穩定，本研究代理人做出動作輸出的時候將會在每期 15 檔股票當中選取 6 檔股票作為當期的投資股票，以期望能加強其投資的穩定性及降低投資風險的能力。最後本研究將評估在進行多個不同的實驗來調整其超參數以進行演算法的優化，並把分析結果拿來跟台灣股票市場的大盤及兩檔在 2016 年排名前十的基金進行投資報酬績效之比較。

DOI:10.6814/NCCU202100964

第四節 論文架構

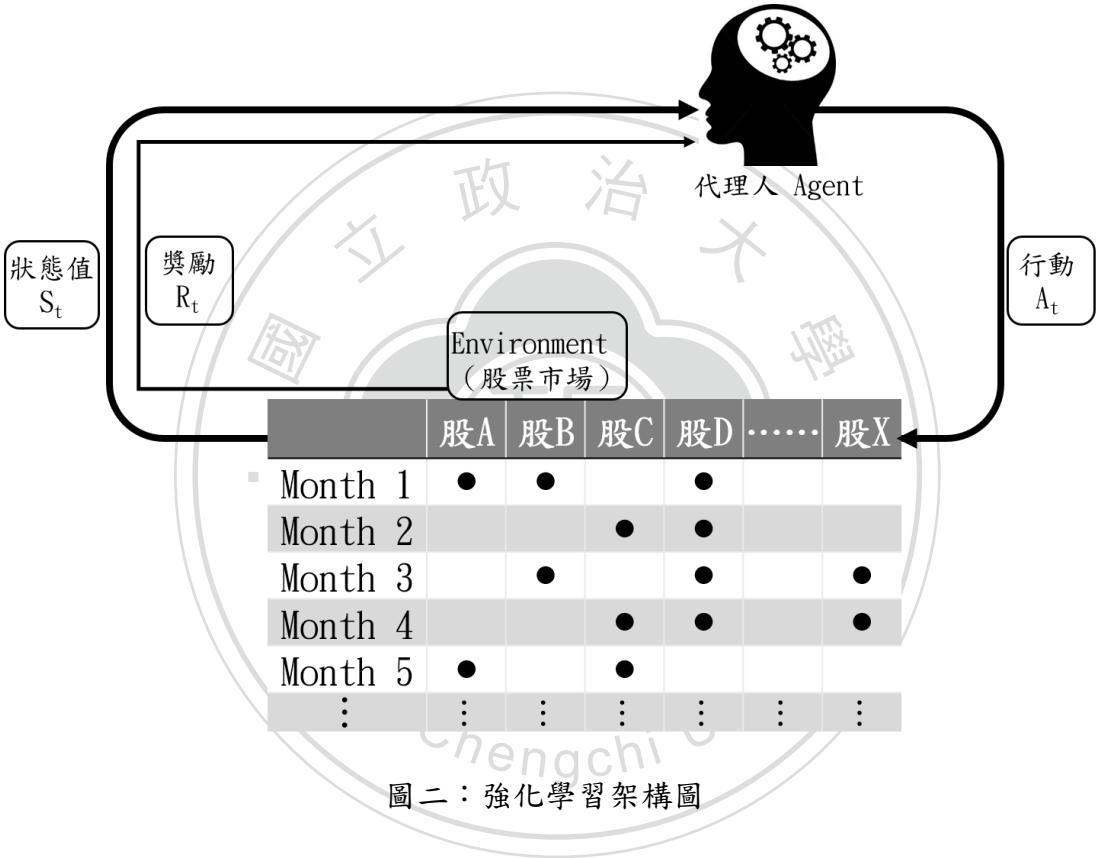
本研究的內容一共分為五個章節，接下來將說明其論文架構的排序，第一章為本文的研究背景、研究動機以及研究目的來彙整出研究的架構，並對國內外相關的研究加以深入的文獻分析，並進行探討與整理，參考相關文獻的方法與結果，引導本文的研究方向與研究目標。第二章將延續前章節之文獻回顧所獲得之論文思路及研究方法，闡述本文所應用的演算法。

本文將在第三章說明本文的變數設定、所研究之範圍其資料、資料蒐集及資料前置處理、金融工具及其衡量投資策略績效的指標，解釋在本文所使用強化學習的應用及設定方式，還有說明超參數的定義及整體的硬體環境與程式工具。第四章將進行本文的實驗研究並對其做出分析，觀察其在不同實驗下的變化結果，然後對結果分析作出比較。最後本文將在第五章將進行總結，根據分析出來的各個結果彙整成本文的結論及提出建議，說明本文的不足之處及討論未來可研究的方向供後續研究作為參考。

DOI:10.6814/NCCU202100964

第二章　文獻回顧

第一節 強化學習



圖二：強化學習架構圖

強化學習就像一個懵懂的孩童透過不停地從錯誤中學習，逐漸找到正確的方向，它不需要像監督式學習那樣，使用已標籤(Label)的數據進行學習，而是可以在沒有人提供指示的情況下自行學習，選擇要採取的行動，最後通過獎懲制度去判斷該行動的適宜程度。在強化學習中需設置一個學習環境，由代理人(Agent)在該環境中進行操作，並根據操作來獲得獎勵或懲罰。在強化學習的訓練過程中需要了解到馬可夫決策過程（Markov Decision Process，MDP），馬可夫決策過程可以用在處理擁有隨機性質且需連續做決策的問題上，而在這

DOI:10.6814/NCCU202100964

過程中其歷史或未來狀態都是互不相關且獨立的，就像加利福尼亞大學 UC Berkeley CS188 的課程教學當中以機器人在迷宮中行走的遊戲為例子，遊戲的設計框架包含了每一格迷宮的狀態、可以選擇的行動、從行動中路徑所轉換的概率及作出動作後所獲得的獎勵或虧損，然後從當前狀態到下一步狀態。像 2015 年舉世聞名的 AlphaGo[1]，便是其中一個強化學習的例子，應用於圍棋上，棋盤作為環境，動作則是下一步要移動的棋子，通過一盤棋局的勝負情況來給予代理人獎勵或懲罰，而代理人所做出的一系列的動作(獲得獎勵的行為)稱為策略(Policy)。強化學習可以應用於解決動態決策及優化控制等問題，而在股票市場，股票的投資買賣則是一種決策問題，是否進行投資或該投資多少都會改變最終所獲得的報酬，本研究希望將深度學習與強化學習結合，並應用於股票市場，用以降低投資風險、優化投資選擇。

2.1.1. 行動

行動(Actions)是代理人在每種狀態下可以執行的操作或選擇，在該時間段的金融環境及狀態下，代理人可以在指定的範圍內採取既定措施或操作來更改當前的狀態，例如學習步行的機器人可選擇或執行前往哪個方向前進。在早期的研究如 Moody et al. (1997) 多以離散動作作為輸出空間，讓代理人在單一投資資產上的輸出動作選擇為買入、持有或賣出，因於神經網絡的運算能力受到限制，但隨著 GPU 硬件的技術改進，學者的研究方向多以著重於連續動作來進行行動操作。本研究將交易時間設定為每個月進行一次交易，即在當個月第一天買進一定比例的股票，然後在該月最後一天賣出該股票以進行獲利結算，但是這樣的方式是有缺陷的，投資人被規定需要在這樣的條件下進行投資操作，且無法表現該時間點的真實數值，於是本研究會將股票資料轉換為月平均值（詳見第三章第一節 變數設定），這樣能在模擬投資人進行投資的情境進行訓練，同時也使得該訓練環境更加的合理化，其中該環境框架則以連續動作作為輸出空間，代理人所做出的行動在於確定其投資選擇，以預期達到利潤最

註[1] 2016 年以 David Silver 帶領的 AlphaGo 研發團隊應用強化學習在圍棋中，並擊敗了著名的世界圍棋界的冠軍李世石(Lee Sedol)。

DOI:10.6814/NCCU202100964

大化，讓代理人在每一期從特定股票（見第三章第二節 選股標的）當中選擇所投資的六檔股票。

2.1.2. 獎勵

當代理人在當下的狀態(S_t)採取行動(A_t)時，它將會獲得獎勵(R_t)，而該獎勵則描述了來自當前環境的反饋，獎勵可以是正或負值，當獎勵為正數，表示代理人執行正確的行動，該獎勵指示應該嚮往的狀態，然而當代理人作出了錯誤的行動，則給予負數的獎勵作為懲罰，即代理人所採取的每個行動(A_t)都與獎勵(R_t)有關聯。在本研究中，代理人將在訓練或測試的期間進行交易以及資產重配置，而獎勵則為該段期間內所投資獲得的利潤作為回報，但在最求利潤最大化的同時，也應該考慮到其投資的風險，因此本研究將回合獎勵函數設計為 Calmar 比率，詳細內容將在第三章第一節的變數設定中介紹 Calmar 比率的計算方式。

2.1.3. 狀態和環境

本研究的狀態值包含了給定股票的每月獲利百分比信息，代理人從狀態值中獲得整體環境的資訊回饋，並從環境中執行行動，然後從環境回饋其獎勵及接到下一個狀態，代理人在其金融環境中，在給定狀態下，透過收集到的初始狀態值，在每一期的時間軸中，代理人必須採取相對應的行動也就是做出當期的投資決策，代理人必須根據狀態輸入來選擇操作，其中也定義為環境的狀態。在環境中，代理人將根據每次的策略選擇對應的動作，該策略是一種代理人行為的描述，以此告訴代理人應為每次不同的狀態選擇最佳的操作，而作為每個動作的結果，代理人會獲得獎勵(R_t)或者是懲罰，隨之狀態轉換也會更新為下一個狀態(S_{t+1})，因此在環境中採取最佳的措施以求達到利潤最大化。本研究在其代碼實現參考了來自 Github 網站 Liangzp[1]、Wassname[2]以及

註[1] GitHub - liangzp/Reinforcement learning algorithms in portfolio management
註[2] GitHub - wassname/RL portfolio management: Attempting to replicate "A Deep Reinforcement Learning Framework for the Financial Portfolio Management Problem"

DOI:10.6814/NCCU202100964

Jhdemendoza[1]對於應用強化學習於股票金融市場環境所分享的代碼，利用強化學習的學習框架來訓練代理人應用於股票投資市場上。

2.1.4. TD3 演算法

由於以價值(value-based)為基礎的 DQN 等演算法，將導致其 Q 值被過於高估計以及無法取得最佳策略[2](Policy)的計算值，Merger et al.(2018) 發表了 TD3，其代碼實現參考了來自學者在 Github[3]的代碼分享，它是屬於 DDPG 延伸版本的演算法，在 DDPG 中採用了 Double Q-learning 的 Twin Critic，在其基礎上透過兩個 Critic 中的最小值函數以防止過高估計值，讓 Critic 有一個更平滑的 Q 函數值，以此解決過高估計值的問題，並且在代理人的行動上引入一個高斯噪音增加探索範圍，除此之外，學者設立延遲其策略的更新，減少每次更新所發生的錯誤用以提高其性能。本研究的資料形態在觀察空間（Observation space）為離散，動作空間（Action space）為連續動作。

Algorithm 1 TD3

Initialize critic networks Q_{θ_1}, Q_{θ_2}, and actor network π_ϕ
with random parameters θ_1, θ_2, ϕ
Initialize target networks $\theta'_1 \leftarrow \theta_1$, $\theta'_2 \leftarrow \theta_2$, $\phi' \leftarrow \phi$
Initialize replay buffer $\mathcal{B}$
for $t = 1$ **to** T **do**
  Select action with exploration noise $a \sim \pi_\phi(s) + \epsilon$,
  $\epsilon \sim \mathcal{N}(0, \sigma)$ and observe reward r and new state s'
  Store transition tuple (s, a, r, s') in $\mathcal{B}$

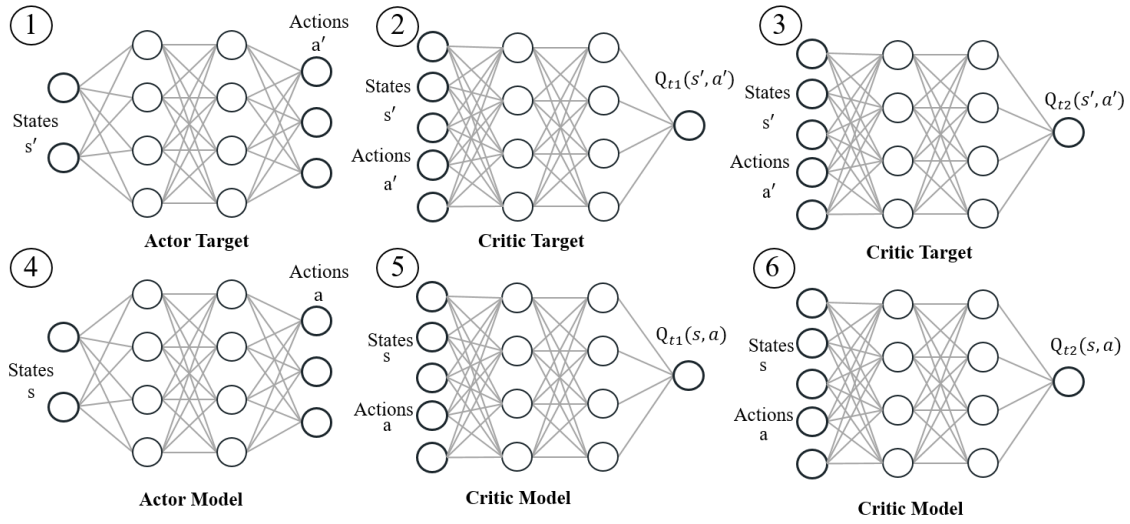
  Sample mini-batch of N transitions (s, a, r, s') from $\mathcal{B}$
  $\tilde{a} \leftarrow \pi_{\phi'}(s') + \epsilon, \quad \epsilon \sim \text{clip}(\mathcal{N}(0, \tilde{\sigma}), -c, c)$
  $y \leftarrow r + \gamma \min_{i=1,2} Q_{\theta'_i}(s', \tilde{a})$
  Update critics $\theta_i \leftarrow \text{argmin}_{\theta_i} N^{-1} \sum (y - Q_{\theta_i}(s, a))^2$
  if t mod d **then**
    Update ϕ by the deterministic policy gradient:
    $\nabla_\phi J(\phi) = N^{-1} \sum \nabla_a Q_{\theta_1}(s, a)|_{a=\pi_\phi(s)} \nabla_\phi \pi_\phi(s)$
    Update target networks:
    $\theta'_i \leftarrow \tau\theta_i + (1 - \tau)\theta'_i$
    $\phi' \leftarrow \tau\phi + (1 - \tau)\phi'$
  end if
end for

圖三：強化學習 TD3 演算法內部示意圖 A

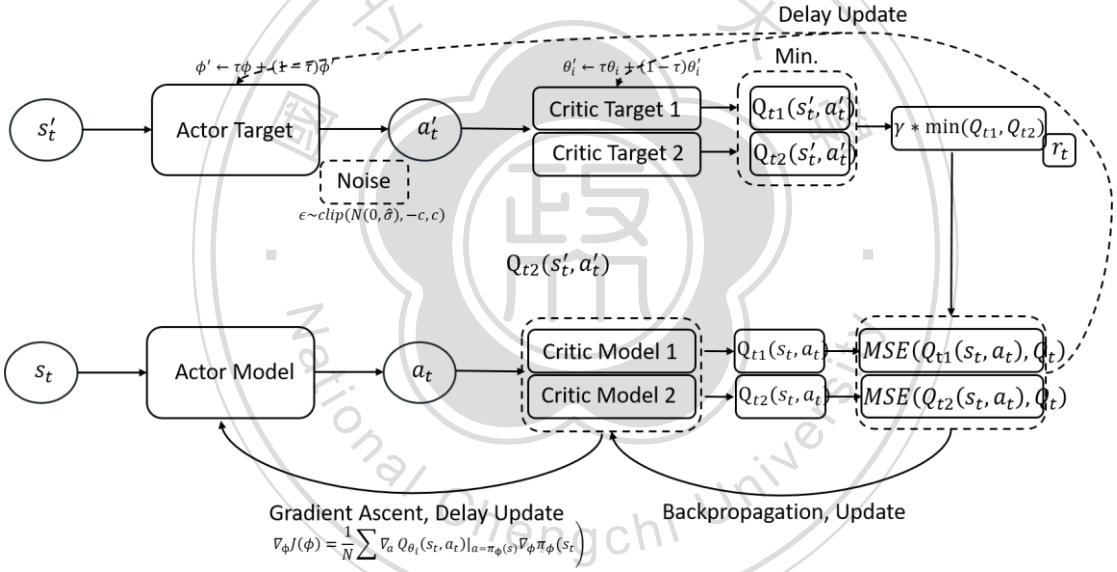
註[1] GitHub - Jhdemendoza/Stock exchange pytorch
註[2] 策略是決定代理人在一個完整的狀態裡所要執行的具體操作，即代理人用來決定要採取哪些行動的規則。

註[3] GitHub - sfujim/TD3: Author's PyTorch implementation of TD3 for OpenAI gym tasks

DOI:10.6814/NCCU202100964



圖四：強化學習 TD3 演算法內部示意圖 B



圖五：強化學習 TD3 演算法內部示意圖 C

以下將說明強化學習 TD3 演算法的演算過程：

以上圖四為其內部構造的示意圖，首先將建立兩組相同的初始 Actor-Critic 的神經網絡並額外建立兩個 Critic Target 的神經網絡以及一個經驗池（Experience replay memory），然後讓其經過一個完整步驟後，從經驗池中隨機抽取 100 個 Transitions 樣本用於訓練，$(s_t, a_t, r_t, s_{t+1}) \leftarrow \text{sample}(\text{batch} - \text{num})$。

DOI:10.6814/NCCU202100964

整體網絡的訓練過程為：首先是從①Actor target，再到兩個②③Critic target和兩個⑤⑥Critic model，最後是④Actor model。

在 Actor Target 裡，s_t'將作為 Actor Target 的狀態值並且用來執行其行動 a_t'。學者為了讓 Actor Target 神經網絡的策略更加的正規化，故在其策略函數π_ϕ引入一個高斯噪音(Gaussian Noise)作為下一次行動 a'的輸出，即加入一個高斯分佈[1]所產生的隨機值，故在原始環境做出行動時增加其搜索範圍並擴大其探索能力，同時也能降低在更新時過擬合所造成的高方差估計值，使其方差最小化，其式子為：

$$a_t'(s_t') \leftarrow y = r_t + \gamma Q_{\theta'}(s_t', \pi_{\Phi'}(s_t') + \epsilon) \ , \epsilon \sim clip(N(0, \hat{\sigma}), -c, c) \qquad (2.1)$$

Merger et al.(2018)學者在這裡引用了 Silver et al.(2016)Double Q-learning 用來處理過高估計問題的運作方式，Critic 需要 Actor 神經網絡做出行動來估計 Q 函數，故需要在 Actor Target(圖四的①)得到行動(a_t')，然後在狀態(s_t')及獲得的行動(a_t')來計算出兩個 Critic Target(圖四的②和③)的 Q 值，即狀態(s_t')透過策略所採取的動作(a_t')所獲得的預期報酬，即得出$Q_{t1}(s_t', a_t')$和$Q_{t2}(s_t', a_t')$，從當中選用較小值作為其輸出結果，以避免過高估計的同時也能帶了訓練的穩定性。

Q 值式子為：

$$Q_t = \Upsilon^0 r_1 + \Upsilon^1 r_2 + \cdots + \Upsilon^{t-1} r_t \qquad (2.2)$$

Υ: 折扣因子 Discount factor

在兩個 Critic Target 間取較小的 Q 值式子為：

$$Q_t = r_t + \gamma^* min(Q_{t1}, Q_{t2}) \qquad (2.3)$$

Q 值的更新式子為：

$$Q_t^* = (1-\alpha)Q_t + \alpha(r_{t+1} + \gamma^* min(Q_{t1+1}, Q_{t2+1})) \qquad (2.4)$$

α: 學習率

注[1] 高斯分佈亦可稱為常態分佈是一種概率分佈，考量均勻變動之下的集中性。

DOI:10.6814/NCCU202100964

Critic Target 的最小化 Q 值跟在兩個 Critic Model(圖四的⑤和⑥) 的 Q 值計算出 Critic 的損失函數（Loss function），式子為：

$$Loss = MSE(Q_{t1}(s,a), Q_t) + MSE(Q_{t2}(s,a), Q_t) \quad (2.5)$$

在得到損失函數後，學者利用隨機梯度下降(SGD，Stochastic Gradient Descent)優化器進行反向傳播(Backpropagation) 的方式對 Critic Model 的參數進行更新。

Actor 又稱為策略，目的為透過策略權重參數(Φ)來其優化策略路徑(π)，最大化預期收益(maximizes the expected return) ，對於 Actor Model 的策略權重更新，透過來自 Critic Model 的 Q 值(見式 2.3)。為了獲得更穩定的策略函數，學者在接下來的步驟提出了延遲更新的做法，學者在每經歷了兩個完整訓練才進行更新，使用 Critic model 較小的 Q 值，透過梯度上升(Gradient ascent) 的方式對 Actor Model 的策略權重進行更新(圖四的④)，其式子為：

$$\nabla_\phi J(\phi) = \frac{1}{N}\sum \nabla_a Q_{\theta_i}(s_t, a_t)|_{a=\pi_\phi(s_t)} \nabla_\phi \pi_\phi(s_t) \quad (2.6)$$

Φ: Actor model 的策略權重

θ_i: Critic model 的權重

當更新其目標網絡的時候，在每經歷兩個完整訓練，利用 Polyak averaging 的方式對 Critic target(式 2.7)及 Actor target(式 2.8)的參數進行緩慢更新:

$$\theta_i' \leftarrow \tau\theta_i + (1-\tau)\theta_i' \quad (2.7)$$

$$\phi' \leftarrow \tau\phi + (1-\tau)\phi' \quad (2.8)$$

τ: 接近於 1 的參數

DOI:10.6814/NCCU202100964

第二節 優化器與激勵函數

2.2.1. 優化器

DP Kingma et al.（2014）提出了 Adam 優化器之後被廣泛的運用，它結合了 RMSprop 和 Momentum 的優勢，適用於大部分的環境任務，但在訓練初期它的方差(variance)會很大且可能會陷入成局部最優（local optima），於是 Liu, et al.(2019)提出了 Radam 改善了網絡優化的過程，它在 Adam 一開始訓練的時候加一個預熱效果(warm-up），當自由度大於 4 的時候它主要的改進是在訓練初期的時候設置一個較小的自適應學習率(r_t，adaptive learning rate)，減少了初期的方差使其能更加穩定的收斂，當方差變小後，其優化器就跟 Adam 一樣。Radam 在進行更新的式子：

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t r_t \tag{2.9}$$

$$r_t = \sqrt{\frac{(p_t - 4)(p_t - 2)p_\infty}{(p_\infty - 4)(p_\infty - 2)p_t}} \tag{2.10}$$

p_t: 自由度

Hinton et al.(2019)發表了 LookAhead 的架構，它並不算是優化器，而是是一個 wrapper 且可以跟其他任何優化器組合一起使用，能讓優化器更穩定的探索，它有兩組權重，快權重(fast weight)及慢權重(slow weight)進行同步，然後更新快權重，其主要是向前探索，快權重走到 k 步後慢權重會用前者來作為更新，快權重每次往前進行探索的時候都會往回看以避免探索太遠以致於走到錯的方向，所以每走 k 步就會往回檢查看自己的方向正確與否，每當走到最新的點 $\theta_{t,k}$ 表示為心得權重，比較與上一點的差距乘上 α 作為一個更新的距離，學者將 α 設置為 0.5，其效果會讓模型更穩定，然後再繼續不斷地探索。

LookAhead 的式子為：

$$\phi_{t+1} = \phi_t + \alpha(\theta_{t,k} - \phi_t) \tag{2.11}$$

DOI:10.6814/NCCU202100964

2.2.2. 激勵函數

線性整流函數(Rectified Linear Unit, ReLU)，在深度學習裡作為非單調激勵函數的 ReLU 跟 Adam 優化器經常的搭配在一起使用，ReLU 的計算方式比 Sigmoid 和 Tanh 來得簡單，能讓神經網絡快速的收斂，但是它會出現神經元死亡的現象，當輸入值為負的時候都會以 0 來計算，於是就有梯度消失的問題。學習率的設置不能太大，這樣才能稍微緩解在 ReLU 梯度消失的問題。

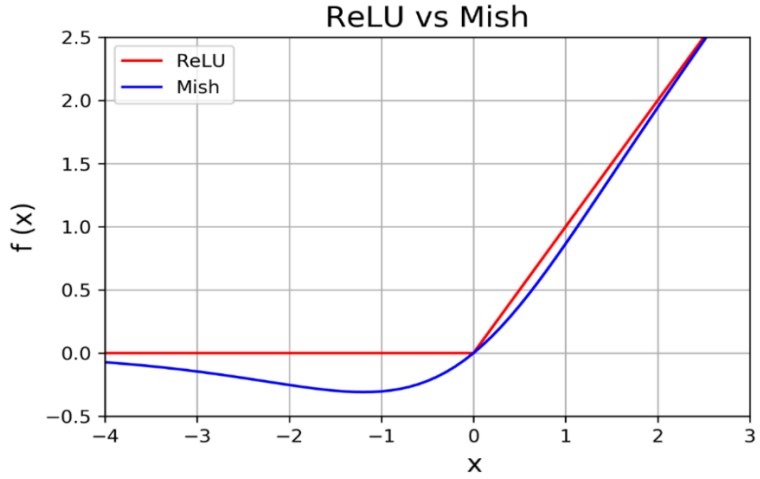
ReLU 函數公式：

$$f(x) = \max(0, x) \tag{2.12}$$

D Misra(2019)發表了 Mish 的激勵函數，跟 ReLU 一樣也是非單調激勵函數，它類似於 Swish，會保留一部分負值的神經元，然後在多大的極端值的負數會接近零，解決了 ReLU 梯度消失的問題，以下圖三為 ReLu 和 Mish 兩種激勵函數對比圖，在 Misra 的論文對 CIFAR10(圖像分類)做了實驗，其結果顯示使用 Mish 激勵函數在其測試集的準確率比 ReLU 高出了 1.671%，同時損失值也比 ReLU 來得低，另外，Misra 還使用 Mish 激勵函數在 75 個基準測試中做了實驗，其中在 55 個的測試結果勝於 ReLU。

Mish 函數公式：

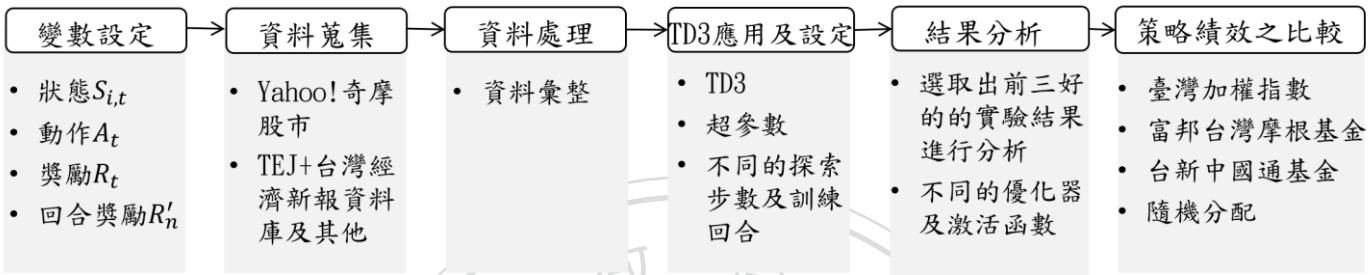
$$f(x) = x * tanh(log(1 + e^x)) \tag{2.13}$$



圖六：激勵函數對比圖

DOI:10.6814/NCCU202100964

第三章　實驗設計



圖七：實驗操作圖

本章第一節將說明本實驗所設定之變數，其中包含狀態$S_{i,t}$、動作$A_{i,t}$、獎勵R_t以及回合獎勵R'_n。在本章第二節將說明本實驗的資料蒐集，本研究在3.2.1.小節說明本文的股票篩選方式，然後將從奇摩財經(Yahoo 奇摩股市)、TEJ+台灣經濟新報資料庫及其他網站收集相應股票資訊之資料，然後進行資料前置處理 (Data pre-processing)，並在 3.2.2.小節做本實驗敘述統計的說明。本文將在本章第三節說明 TD3 的應用、超參數的設定以及本實驗所應用的硬體環境與程式工具。

本實驗將使用同為策略績效衡量指標的年均複合增長率(CAGR)、夏普比率及 Calmar 比率(見式 3.7)作為本實驗在結果的顯示，本研究將使用兩個衡量指標用以評估實驗結果績效的標準，其中一個為年均複合增長率，由於平均報酬率(Average Rate of Return)沒有考慮每年該投資項目的實際金額，故當投資時間越長，其實際價報酬率則越不準確，而年均複合增長率 (Compound Annual Growth Rate, CAGR)能看出在一定時間之內該投資在沒有撤資且更新每年資產額的情況下，算出平均一年所賺取的投資報酬率，故 CAGR 很常被用來評估同產業類別股票報酬率的表現能力。

DOI:10.6814/NCCU202100964

年均複合增長率的公式為：

$$CAGR = \left(\frac{Value_{t_n}}{Value_{t_0}}\right)^{\frac{1}{t_n - t_0}} - 1 \quad (3.1a)$$

$Value_{t_n}$ ：最終期價值

$Value_{t_0}$ ：初始價值

$t_n - t_0$ ：共投資年數

由於本實驗所測試的資料為月資料,且不使用投資金額來計算其報酬而是使用報酬率來計算，故將公式調整為：

$$CAGR = \left(1 + \sum_{t=1}^{n} R_t\right)^{\frac{12}{m_n - m_0}} - 1 \quad (3.1b)$$

R_t：報酬率(見式 3.6)

$m_n - m_0$ ：共投資月數

t：時間單位

i：個股表示

由於 CAGR 只能用來評估該公司的賺錢能力，但它不能反映其投資風險，故有鑑於此本研究將同時應用 Sharpe(1994)所提出的夏普比率(Sharpe ratio)納入評估標準之一，其計算方式為收益中減去無風險利率，再除以其超額收益的標準差。夏普比率能算出在每一單位的風險波動下能得到的報酬，高夏普比率意味著在比較小的風險波動之下的報酬率穩定的成長，由於本研究不考慮無風險利率，故在計算夏普比率時它的值為零。

DOI:10.6814/NCCU202100964

夏普比率的公式為：

$$\text{夏普比率} = \frac{R_p - R_f}{\sigma_p} \tag{3.2}$$

R_p：該投資組合的預期報酬率

R_f：無風險利率(risk-free rate)

其中標準差為：

$$\sigma_p = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{3.3}$$

x_i：在投資期間當期的投資報酬率

μ：在投資期間的平均投資報酬率

σ_p：該投資組合的標準差(standard deviation of the portfolio)

DOI:10.6814/NCCU202100964

第一節 變數設定

表一：變數設定表

符號	屬性	名稱	描述
$S_{i,t}$	狀態	月平均值	i：個股表示(股票代號為 1101、1216、1301、1303、1326、2002、2207、2303、2308、2317、2330、2357、2382、2603 及 2912) 在狀態欄裡，Pendharkar et al.（2018）使用報酬率作為其資料輸入，本文將使用 15 檔股票的月平均值（式 3.4）作為其狀態的資料輸入。
A_t	動作	股票權重	Kanwar（2019）使用 Softmax 於股票中的配置作為其資料輸出，本文將參考學者輸出的設定並將在每一期股票當中選取 6 檔股票作為當期投資股。
R_t	獎勵	報酬比率	獎勵R_t為當期的報酬率（式 3.6）。
R'_n	回合獎勵	Calmar 比率	n：投資期間的時間長度 Kanwar（2019）使用報酬率作為其回合獎勵，以期望在訓練過程中獲得最大報酬率。蔡岳霖（2013）針對不同的績效指標（如：報酬率、最大回撤率、夏普比率、Sortino 比率及 Calmar 比率等績效指標）進行分析與比較，其中發現 Calmar 比率的研究結果跟其他的績效指標結果比起來較為優異，Calmar 比率的分母為最大回撤比率，在最求最大報酬的情況下，同時期望能把風險降到最低，故本文將使用 Calmar 比率（見式 3.7）作為本文的回合獎勵函數。

由於月報酬率的公式解釋為當月月初買入該檔股票並在當月月尾賣出所賺取的報酬率，在模擬投資人進行投資情境時，以這種方式無法呈現該檔股票股價在現實中的真實趨向，故本研究在狀態欄的資料形態上將以當月平均值（式 3.4）的方式呈現，希望藉此讓整體訓練環境的情境更加的合理化。

其中報酬率的式子為：

$$報酬率_{i,t} = \frac{收盤價_{i,t} - 開盤價_{i,t}}{開盤價_{i,t}} \quad (3.4a)$$

t：時間單位

i：個股表示

DOI:10.6814/NCCU202100964

$S_{i,t}$的式子為：

$$S_{i,t} = \frac{每日報酬率加總_{i,t}}{當月開盤天數} \tag{3.4b}$$

Kanwar (2019)使用 Softmax 於股票中的配置作為其資料輸出，本文將參考學者輸出的設定並將在每一期股票當中選取 6 檔股票作為當期投資股。代理人在 Actor Model（如圖八的④）透過狀態$S_{i,t}$做動作輸出$A_{i,t}$，選出當期 6 檔股票作為其欲投資的股票權重。

A_t的式子為：

$$A_t = \sum_{i=1}^{6} A_{i,t} = 1 \tag{3.5}$$

t：時間單位

i：個股表示

獎勵R_t為當期的報酬率，代理人在 Actor Model（圖八的④）做出動作$A_{i,t}$對應到權重，然後乘以月平均值即為當期能獲得的獎勵R_t，本研究將獎勵R_t設定為當期代理人所選擇股票的投資權重乘以當期的報酬率（即月平均值），作為代理人在當期所獲得的投資報酬率。

R_t的式子為：

$$R_t = \sum_{i=1}^{6} r_{i,t} w_{i,t} \tag{3.6}$$

$w_{i,t}$：代理人在時間t對該股票所做出的投資權重

$r_{i,t}$：在時間t該股票的投資報酬

DOI:10.6814/NCCU202100964

蔡岳霖（2013）針對不同的績效指標（如：報酬率、最大回撤率、夏普比率、Sortino 比率及 Calmar 比率等績效指標）進行分析與比較，其中發現 Calmar 比率的研究結果跟其他的績效指標結果比起來較為優異，Calmar 比率的分母為最大回撤比率，在最求最大報酬的情況下，同時期望能把風險降到最低，故本文參考了蔡岳霖（2013）的實驗研究成果選取 Calmar ratio 作為本文的回合獎勵函數。回合獎勵$R_n^{'}$為一個完整回合的總和報酬，本研究中回合獎勵$R_n^{'}$的設定參考 Acar(1997)所提出的 Calmar 比率，將其作為衡量投資策略績效指標的工具之一，即年均複合增長率除以所發生的最大回撤率，Calmar 比率可以識別出投資之間的最大風險幅度，用來衡量投資期間所面臨的最大幅度波動，可以理解為在投資期間最大一筆的投資損失下，盈利與虧損的關係，在投資期間利用 MDD 作為其風險測量的準則，Calmar 比率反映了投資人在虧損幅度的波動程度。

R_n'的式子為：

$$R_n' = \text{Calmar 比率} = \frac{CAGR}{MDD} \tag{3.7}$$

$CAGR$：年均複合增長率(見式 3.1)

MDD：最大回撤率

n：投資期間的時間長度

其中最大回撤率(Maximum Drawdown, MDD)作為一個投資策略評估的指標之一，為投資人在過去的投資期間虧損最大值的差距，透過 MDD 可以得知投資人投資人在投資期間最大一筆的投資損失。

MDD的式子為：

$$MDD = \frac{R_H - R_L}{R_H} \tag{3.8}$$

R_H：獎勵R_t期間的最高值

R_L：獎勵R_t期間的最低值

DOI:10.6814/NCCU202100964

第二節 資料收集及資料前置處理

3.2.1. 選股標的

本研究將從臺灣股票市場中依據篩選原則選出部分股票作為研究對象，接下來將簡單說明作為本研究之研究對象的篩選標準。將參考不同論文的選股原則來擬訂合適的篩選標準，如：Kanwar（2019）的選股方式為自行指定 10 檔美國股票作為其訓練對象，陳昱安（2020）的選股方式為股票市場中市值前十大的股票作為其訓練對象。經審慎考量後本研究採以陳昱安（2020）的選股方式，以臺灣股票市場市值的總值來作為篩選條件來進行股票選取，並期望作為研究對象的股票擁有長期的成長趨勢。

在臺灣股票市場中市值最大的前十五檔股票，其名稱為台積電、聯發科、鴻海、台塑化、中華電、富邦金、台達電、國泰金、南亞、台塑、聯電、中鋼、台化、日月光投控及中信金，由於本實驗在整體的研究時程為 2000 年 1 月至 2020 年 12 月，故本研究將篩選範圍聚焦於已上市至少 21 年且佔股票市場總市值前十五檔之股票，其分別為台積電(2330)、鴻海(2317)、台達電(2308)、南亞(1303)、台塑(1301)、聯電(2303)、中鋼(2002)、台化(1326)、統一(1216)、長榮(2603)、廣達(2382)、和泰車(2207)、台泥(1101)、華碩(2357)及統一超(2912)，本研究將從奇摩財經(Yahoo 奇摩股市)及 TEJ＋(台灣經濟新報資料庫)收集臺灣股票資訊的相應資料。

DOI:10.6814/NCCU202100964

3.2.2. 敘述統計

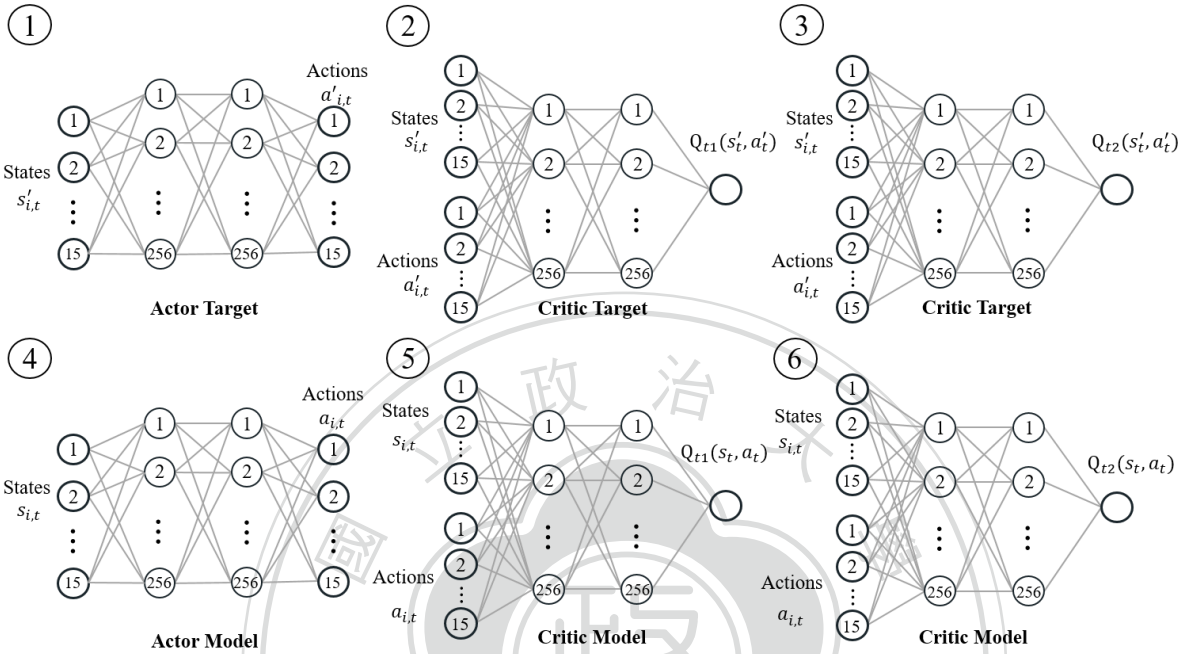
表二：敘述統計表

屬性	描述
總時間長度	2000/1~2020/12
總時間長度數量(月)	252
訓練時間長度	2000/1~2016/12
訓練時間長度數量(月)	204
測試時間長度	2017/1~2020/12
測試時間長度數量(月)	48
股票數量	15
數據總數	3780
平均值	-0.000373243
標準差	0.004273631
極小值	-0.030190652
極大值	0.067954207

以上為敘述統計表(詳見表 2)，本研究時程為自 2000 年 1 月起，至 2020 年 12 月，共計 252 個月，分為兩個階段，訓練階段及測試階段，前 17 年(2000 年至 2016 年)為訓練時間；後面四年(2017 年至 2020 年)用來測試。共選擇十五檔股票作為研究對象，數據總量為 3780 個數據資料，其平均值為-0.000373243，標準差約為 0.0043，極大值大約為 0.068，極小值則約為-0.03。

DOI:10.6814/NCCU202100964

第三節 TD3 應用及設定



圖八：強化學習 TD3 應用實驗設計圖

本研究應用了來自學者在 Github[1]的 TD3 代碼分享，本研究在 TD3 的實驗設計，在狀態設定中共有 15 個狀態$S_{i,t}$(即 15 檔股票，其股票代號為 1307、1452、1521、1527、1707、2108、2316、2345、2374、2377、2383、2404、2485、9910 及 9924)，在動作輸出設定中共有 15 個動作$A_{i,t}$，其中將代理人的動作設定為在每一期的股票當中選取 6 檔股票作為當期投資股票權重，在獎勵變數的設定中獎勵R_t為當期的報酬率，一個完整的回合獎勵R'_n設定為 Calmar 比率，以 40 個月為代理人投資期的時間長度，而每月投資所獲得的報酬率的計算方式則以每月報酬平均值來進行計算。本研究使用的測試方式如下，在實驗當中使用 17 年(2000 年 1 月至 2016 年 12 月)的歷史資料作為訓練資料集，在訓練的時候在每個訓練回合將以隨機抽樣的方式作為時間起點，其次，本研究將使用 4 年(2017 年 1 月至 2020 年 12 月)作為測試資料集，而總測試時間長度為 48

註[1] GitHub - sfujim/TD3: Author's PyTorch implementation of TD3 for OpenAI gym tasks

DOI:10.6814/NCCU202100964

個月，在測試起點沒有重複的情況下，其總回合數則一共為 9 次，實驗的測試方式將在總測試時間長度中進行 9 次的測試並以實驗結果的加總平均方式來呈現。

代理人在 Actor Target(圖八的①)的動作輸出做出投資選擇時加入了隨機噪音，其隨機值是來自一個高斯分佈所產生的值，目的是為了在環境中擴大其探索範圍，從①那裡得到$a'_{i,t}$後將會作為兩個 Critic Target(圖八的②和③)的動作輸入值，從而在②和③得到的兩個 Q 值當中取較小的 Q 值(見式 2.3)，然後跟⑤和⑥Critic Model 的 Q 值計算出 Critic 的損失函數(見式 2.5)用來優化兩個 Critic Model，最後在每兩次完整訓練過程利用從兩個 Critic Model 得來的 Q 值對 Actor Model(圖八的④)進行更新一次，在 Actor Model 透過狀態$S_{i,t}$做動作輸出$A_{i,t}$選出當期 6 檔股票作為其要投資股票的權重。

DOI:10.6814/NCCU202100964

3.3.1 超參數設定

表三：TD3 超參數設定表 A

Description	Value				
Experiment number	1	2	3	4	5
Coefficient for updating the target network	0.995	0.995	0.995	0.995	0.995
Discount factor	0.99	0.99	0.99	0.99	0.99
Learning rate for Actor	0.001	0.001	0.005	0.01	0.001
Learning rate for Critic	0.001	0.001	0.005	0.01	0.001
Number of units in hidden layers	(256,256)	(256,256)	(256,256)	(256,256)	(256,256)
Batch size	100	100	100	100	200
Activation function	ReLU	Mish	Mish	Mish	Mish
Optimizer	Adam	Ranger	Ranger	Ranger	Ranger
Action noise for the critic update	0.2	0.2	0.2	0.2	0.2
Noise clip threshold	0.5	0.5	0.5	0.5	0.5
Investment time length (n)	40	40	40	40	40

表四：TD3 超參數設定表 B

Description	Value				
Experiment number	6	7	8	9	10
Coefficient for updating the target network	0.995	0.97	0.995	0.995	0.995
Discount factor	0.99	0.99	0.99	0.99	0.99
Learning rate for Actor	0.001	0.001	0.001	0.001	0.001
Learning rate for Critic	0.001	0.001	0.001	0.001	0.001
Number of units in hidden layers	(256,256)	(256,256)	(256,256)	(256,256)	(256,256)
Batch size	100	100	100	100	100
Activation function	Mish	Mish	Mish	Mish	Mish
Optimizer	Ranger	Ranger	Ranger	Ranger	Ranger
Action noise for the critic update	0.05	0.2	0.2	0.2	0.2
Noise clip threshold	0.5	0.5	0.5	0.5	0.5
Investment time length (n)	40	40	10	20	30

DOI:10.6814/NCCU202100964

本文參考了在GitHub網站[1]所分享以 RAdam 和 LookAhead 合二為一的一個結合體名為 Ranger 優化器的編碼，將 LookAhead 套用在 RAdam 身上，Radam 解決了 Adam 方差大的問題，而 LookAHead 提供了更穩定的探索能力，RAdam 藉著 LookAhead 抵達頂點的時候再次進行批次探索，(k 參數-抵達頂點然後再探索的次數)，然後乘上 α 參數來更新 RAdam 的權重。憑藉 LookAhead 的往後看機制，優化器可以充分地在附近進行探索，不用擔心陷入局部最優的情況。本研究將在第四章分別對Adam和Ranger以不同的優化器來進行實驗測試，然後將兩者結果作以比較。另外，本研究參考了同樣來自 Github 網站[2]分享 Mish 激勵函數的代碼, Mish 的激勵函數跟 ReLU 一樣是非單調激勵函數，它類似於 Swish，會保留一部分負值的神經元，然後在多大的極端值的負數會接近零，解決了 ReLU 梯度消失的問題。本論文的實驗一至四為優化器與激勵函數的比較及其學習率的調整，在實驗一使用 Adam 優化器及 ReLU 激勵函數，而在實驗一至三則使用 Ranger 優化器和 Mish 激勵函數，以及調整其學習率，期望透過學習率的調參能提升實驗結果的性能。

Batch size 作為在每次的訓練回合中從訓練樣本數(Batch size)抽取樣本在整個神經網絡中進行訓練來更新參數，本研究在實驗五將訓練樣本數調大至200用以測試實驗的性能。本文在實驗六將調整TD3超參數之一的隨機噪音，即代理人所做出的動作將加入隨機噪音以此來測試代理人穩定性，隨機噪音調小至 0.05，測試在較小的隨機噪音所做出的的動作是否會帶來更好的實驗結果。本文將在實驗七調整作為更新其目標網絡(Critic target 及 Actor target)的超參數τ，將更新其目標網絡的幅度調大至0.03，測試其實驗的性能。本文將在實驗八至十調整實驗中的投資時間長度，即為代理人在每個訓練回合要做投資的時間長度(n)，在實驗八至十將投資時間長度的調整設置分別為20、80及100期的投資時間，測試在不同的步長在實驗結果上是否會有差異性。

註[1][2] GitHub - lessw2020/Mish: Mish Deep Learning Activation Function

DOI:10.6814/NCCU202100964

3.3.2. 硬體環境與程式工具

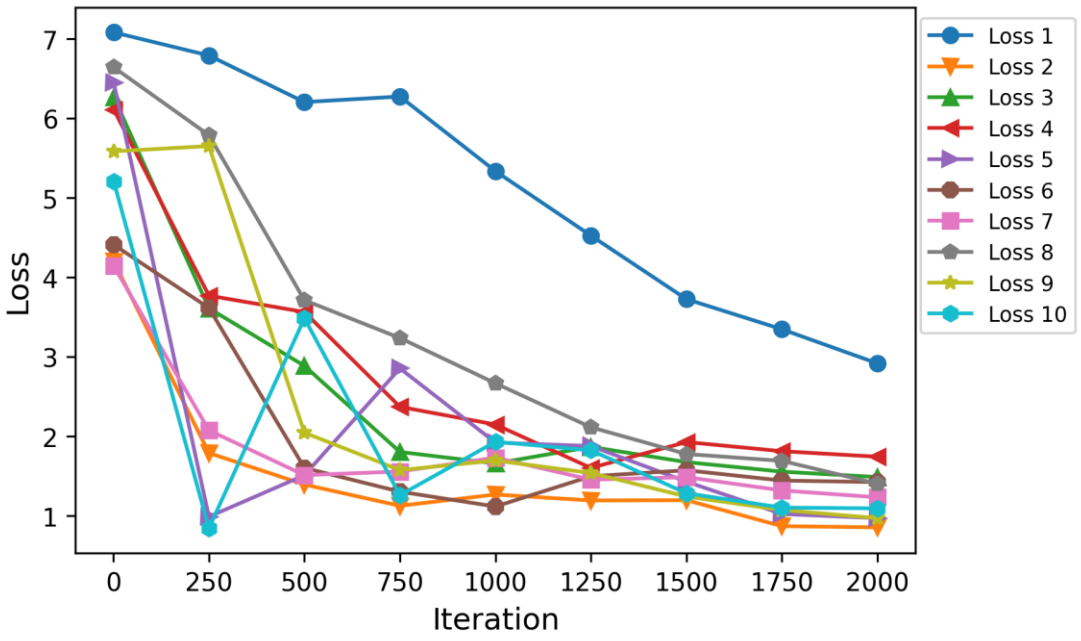
表五：硬體環境與程式工具表

硬體環境與程式工具
Google 雲端 • 作業系統：Ubuntu 20.04 LTS • CPU: n2-standard vCPUs 程式工具： • Anaconda3：2020.02 版 • Python：3.7.6 版 • JupyterLab：2.2.0 版 • Torch：1.5.1 版 • Cuda：10.1.243 版 • Gym：0.17.2 版

DOI:10.6814/NCCU202100964

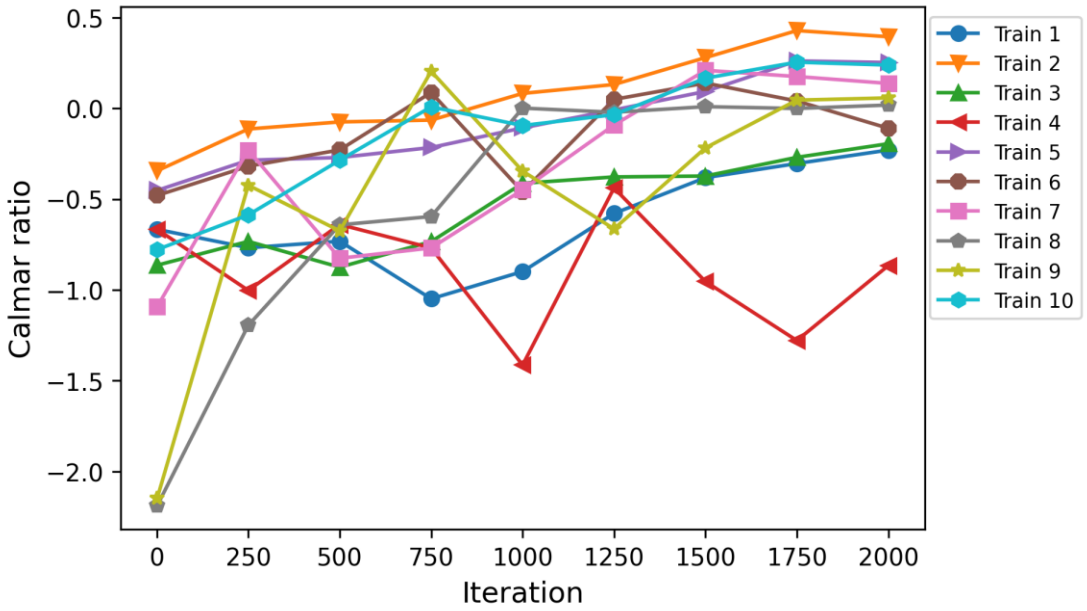
第四章 實驗結果

本研究所使用的資料為臺灣證券交易所(TWSE)的股票，在經過股票篩選後最終選出十五檔股票作為本研究的實驗對象，本研究的資料來源是來自奇摩財經(Yahoo 奇摩股市)、TEJ+台灣經濟新報資料庫及其他網站收集相應股票資訊之資料，本研究的時間區間將以每月為時間單位，資料集的總時間長度為 21 年（2000 年 1 月 至 2020 年 12 月），其中訓練集 17 年（2000 年 1 月 至 2016 年 12 月），以及測試集 4 年（2017 年 1 月 至 2020 年 12 月），而每月投資所獲得的報酬率的計算方式則以每月報酬平均值來進行計算，以下圖九及圖十為本實驗一至十的訓練結果線狀圖。



圖九：實驗一至十的訓練結果 Loss 線狀圖

DOI:10.6814/NCCU202100964



圖十：實驗一至十的訓練結果 Calmar 比率線狀圖

實驗一至四使用了不同的優化器及激勵函數進行實驗訓練，在本研究實驗一至四的實驗結果當中，實驗二至四係使用 Ranger 為優化器及 Mish 為激勵函數，此項實驗的學習率之設定分別為實驗二 0.001、實驗三 0.005 及實驗四 0.01，訓練到第一千七百五十次的時候損失函數已經降至最低點，而實驗一 Adam 為實驗中屬於神經網絡的優化器及以 ReLU 為激勵函數，其學習率的設定為 0.001，即使訓練到第兩千次其損失函數還未達到最低且仍然有下降的空間，使用 Ranger 優化器及 Mish 激勵函數在訓練的時候其損失函數下降的速度比 AdamReLU 來得快，在實驗一 Calmar 比率的訓練結果為-0.23，而實驗二至四 Calmar 比率的訓練結果分別為 0.394、-0.195 及-0.866，在實驗二至四當中發現，當學習率越小則實驗結果越好，透過學習率的調參提升了實驗訓練結果的性能，其中損失函數從實驗二 0.8565 至實驗四 1.7451，兩者相差了接近 1 倍，自實驗四到實驗二的損失函數乃逐漸降低，而 Calmar 比率也從實驗四的-0.866 提升至實驗二的 0.394。

DOI:10.6814/NCCU202100964

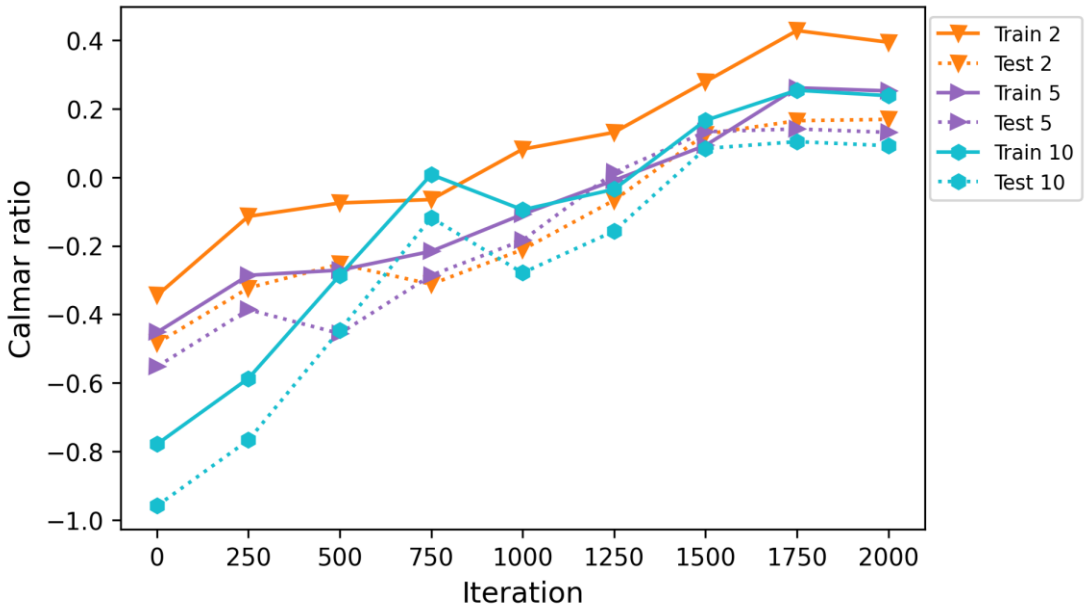
本研究的實驗五為 Batch size 超參數的調參，其實驗在 Batch size 的設定為 200，在訓練到第兩千次的時候損失函數為 0.9717，相較於訓練實驗二的 0.8565 來得高，表現得相對不好，因此在實驗五訓練結果的 Calmar 比率 0.252，也比在訓練實驗二得到的 Calmar 比率 0.394 來得低，由此得知在實驗五當中透過調整 Batch size 並沒有助於優化損失函數的實驗訓練結果，損失函數從實驗二訓練結果的 0.8565 及實驗五的訓練結果 0.9717 相差了約 11.9%。

而本研究的實驗六為對動作輸出隨機噪音超參數進行調參，這項實驗將其超參數設定為 0.05，也就是在動作輸出的時候只有 5%的隨機噪音，比原本預設的超參數設定低了 15%，訓練到第兩千次的時候損失函數為 1.4257，其損失函數比實驗二的損失函數 0.8565 來得高，實驗六訓練結果的 Calmar 比率為-0.11，綜上所述，透過實驗六在本研究當中降低隨機噪音超參數並沒有提升訓練結果的效果。

另外，實驗七為調整神經網絡更新 Target network 的超參數(見式 2.7 和 2.8)，本實驗將其超參數設定為 0.97，即每次更新 Target network 的速度為 3%，比原本預設的超參數設定低了 0.025，訓練到第兩千次的時候損失函數為 1.2346，而訓練結果中實驗七的 Calmar 比率為 0.137，使用原始預設的超參數比實驗七提高 Target network 超參數的損失函數來得低，以最終成果而言，實驗二的訓練結果仍比實驗七的實驗結果來得好。

DOI:10.6814/NCCU202100964

第一節 測試結果



圖十一：實驗二、五及十的測試結果線狀圖

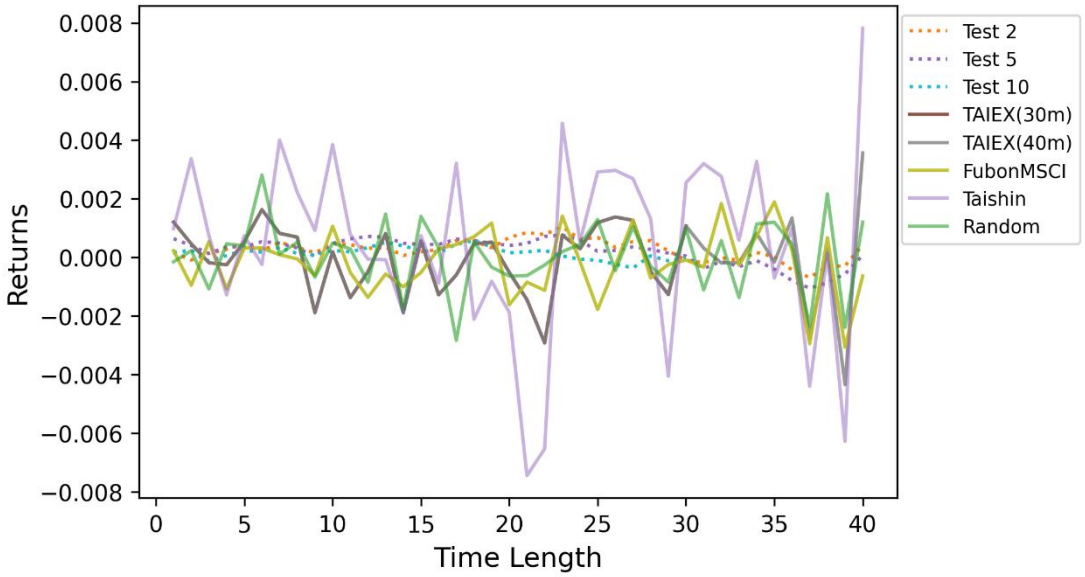
圖十一為本實驗一至十當中前三個最佳實驗，分別為實驗二、五及十的訓練及測試結果線狀圖，本實驗的總測試時間為 2017 年 1 月至 2020 年 12 月，共計 48 個月，在實驗二、五及十的投資時間長度分別為 40、40 及 30 個月，測試結果將以滾動測試樣本的方式來進行結果分析，而模型評估的計算方式為測試起點在沒有重複的情況下，故總回合的投資次數則一共為 9 次，實驗的測試方式將在總測試時間長度中進行 9 次的測試並以實驗結果加總平均的方式來呈現，而在實驗十的投資時間長度為 30 個月，故在總測試時間長度中進行 19 次的測試。在這三項實驗結果當中 Calmar 比率的測試結果隨著訓練次數增加而逐步上升，在訓練到第兩千回合的時候，實驗二、五及十訓練結果的 Calmar 比率分別為 0.394、0.252 及 0.238，而其測試結果的 Calmar 比率分別為 0.169、0.131 及 0.092，雖然實驗二在第兩千回合訓練結果的 Calmar 比率為最高，且比第實驗十的訓練結果高出了 0.8 倍，但在測試結果 Calmar 比率跟實驗五及十卻沒有明顯大的落差。

DOI:10.6814/NCCU202100964

第二節 績效策略比較

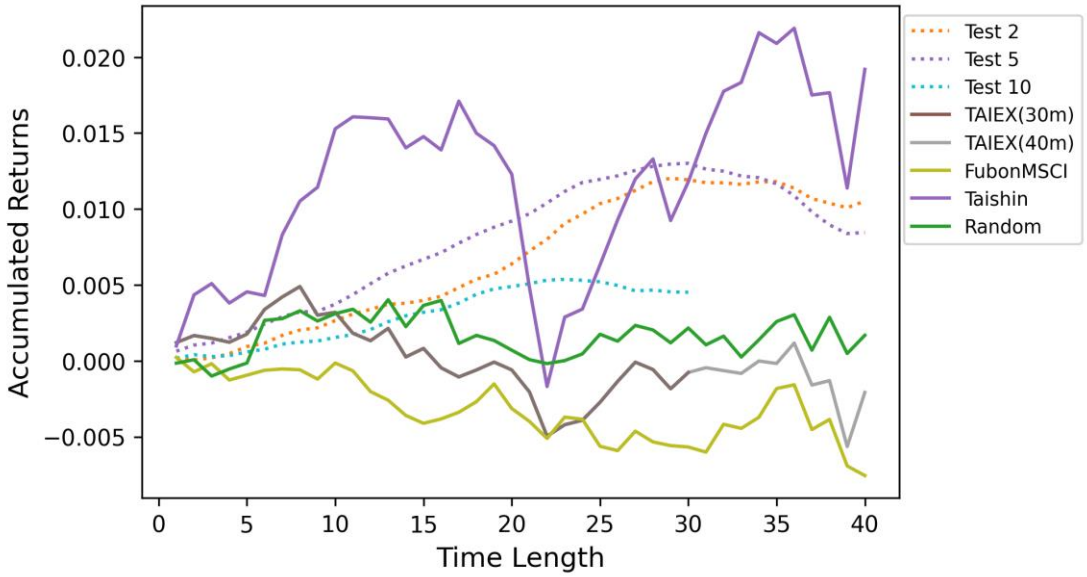
表六：測試結果與其他績效策略比較表

	實驗二	實驗五	實驗十	臺灣加權指數(前30個月)	臺灣加權指數(前40個月)	富邦台灣摩根基金	台新中國通基金	隨機分配
年均複合增長率	0.32%	0.25%	0.18%	-0.03%	-0.06%	-0.23%	0.57%	0.01%
Calmar 比率	0.169	0.131	0.092	-0.011	-0.027	-0.087	0.293	0.008
夏普比率	0.758	0.526	0.445	-0.077	-0.126	-0.594	0.529	0.064
累積報酬率	1.05%	0.85%	0.45%	-0.08%	-0.21%	-0.76%	1.92%	-0.07%
平均每月報酬率	0.03%	0.02%	0.02%	0.00%	-0.01%	-0.02%	0.05%	0.00%
獲利比率	53.89%	49.72%	62.81%	56.67%	55.00%	42.50%	62.50%	47.50%



圖十二：測試結果報酬率線狀圖

DOI:10.6814/NCCU202100964



圖十三：測試結果累積報酬率線狀圖

表六為本實驗測試結果與其他績效策略的比較，其中除了實驗二、五及十之外，還包含了臺股大盤、兩檔投資基金及隨機分配[1]等投資策略，並以年均複合增長率、Calmar 比率、夏普比率、累積報酬率、平均每月報酬率[2]及獲利比率[3]的結果來呈現，圖十二為實驗二、五及十與其他績效策略在整個投資期間的報酬率線狀圖，圖十三則為其累積報酬率線狀圖，圖中臺灣加權指數以 TAIEX 來表示，30m 意指投資 30 個月的行為，富邦台灣摩根和台新中國通這兩檔基金都是國內比較好的基金，在 2016 年其回報率排名都排在前十名[4]，且該基金的前十大持股的股票[5]都是台灣市場上的股票，也就是說這兩支基金都是投資國內的股票，加權指數跟股票型基金[6]不同，加權指數沒有資金上限的問題，而股票型

註[1] 隨機分配策略即為在每期將隨機選擇 10 檔股票進行投資
註[2] 平均每月報酬率 = 累積報酬率 ÷ 總投資時間
註[3] 獲利比率為每次投資所獲利的次數比率，獲利比率 = 獲利次數 ÷ 總投資時間
註[4] 資料來源源自：台股基金 2016 年績效回顧-綠角財經筆記
註[5] 中華民國證券投資信託暨顧問商業同業公會可查找該檔基金所持有前十大股票的股份比例。
註[6] 證券投資信託基金管理辦法指出，股票型基金指投資股票總額達基金淨資產價值百分之七十以上者。

DOI:10.6814/NCCU202100964

基金則有資金上限的限制。表六的實驗二、五及十將以第兩千次訓練回合的測試結果來表示，加權股價指數、兩檔基金及隨機投資策略報酬率的計算方式將採取跟本實驗一致的月平均率作為其報酬率，兩檔投資基金及隨機策略的投資時間皆為測試期間前 40 個月，而臺灣加權指數的投資時間分別為測試期間的前 30 及 40 個月。

透過表六的結果顯示在訓練期間其實驗結果前三佳的實驗二、五及十，其年均複合增長率皆優於加權股價指數的結果，實驗二的 Calmar 比率為 0.169，其年均複合增長率高達 0.32%，比加權股價指數高出了 0.38%的差距，雖然臺灣加權指數的獲利比率為 55%比實驗二的 53.89%高出了 1.11%，但在年均複合增長率卻比實驗二的來得低，表示在實驗二獲得的報酬率比臺灣加權指數高，而實驗十卻有較高的獲利比率，其獲利比率為 62.81%，比實驗二高出了 8.92%，但平均每月報酬率卻比實驗二的低了 0.01%，表示實驗十有比較高的獲利比率是因為其投資時間只有三十個月遠比其他投資時間少了四分之一。

年均複合增長率、累積報酬率、平均每月報酬率及 Calmar 比率在各個績效策略相當中呈現相同幅度的落差，台新中國通基金在眾多投資策略當中是最優的選擇，但從夏普比率來看，實驗二則比台新中國通基金來得好，這意味著雖然後者的回報率是最高的，但前者所帶來的投資策略卻較為穩定。實驗二的累積報酬率達到了 1.05%優於實驗五的 0.85%和實驗十的 0.45%， 從圖十二來看，雖然台新中國通基金擁有最高的投資報酬率，但實驗二、五及十的波動幅度比其他投資策略都來得小。

DOI:10.6814/NCCU202100964

第五章 結論與未來展望

第一節 結論

本研究在強化學習框架中所訓練之智慧代理人在環境模擬訓練的過程中，在一定程度下能成功捕捉到股票市場上股票價格的變動並且能夠實現有效的自我提升，在實驗二、五及十的測試結果皆優於加權股價指數及隨機分配，在只使用了有限的數據資料集來進行訓練的情況下，已能捕捉股票市場的價格波動，若未來能加強金融股票市場相關資訊的資料納入考量，讀取更有效的市場訊息便有機會能出現大幅度的投資獲利，該章節將整理本研究的實作結果並做出總結，另外探討能予以改善的地方以及提供建議，以利未來相關研究能參考採納。本文將強化學習應用於臺灣股票市場，建立一個金融市場股票投資的環境且模擬投資人於股票市場上所做出股票投資的策略選擇，利用強化學習演算法在該環境進行實作，並判讀該智慧代理人對於本模擬環境的學習能力是否合宜，期望得以透過強化學習演算法不斷地訓練智慧代理人以掌握股票市場的獲利趨勢。

我們從圖九及圖十的訓練研究結果趨勢圖來看，當訓練回合達到約 1,750 回合的時候，訓練結果的 Calmar 比率值已接近於趨緩，在實驗一至二的訓練結果中，我們使用了不同的優化器及激勵函數來做比較，當我們將優化器及激勵函數從 Adam 和 ReLU 更換為 Ranger 和 Mish，其訓練效果得到了大幅度的提升，且 Calmar 比率也由負數轉為正數，可以看到在此實驗中 Ranger 和 Mish 的性能明顯優於 Adam 和 ReLU，本研究期望透過修改及調整超參數來找出最佳的訓練模型，然而從實驗三至七的訓練結果中顯示，但是在嘗試調整演算法超參數的實驗結果並沒有獲得更好改善，其中在實驗八至十的訓練結果中顯示，在調整不同的投資時間長度下，發現訓練代理人在進行越長的投資時間其訓練

DOI:10.6814/NCCU202100964

結果的 Calmar 比率就越高，實驗二在訓練長達 40 個月的投資下比短期投資的訓練效果還要好，而在其中投資時間只有 10 個月的實驗八的訓練效果則最低。

從圖十一實驗二、五及十的測試結果趨勢圖中顯示，雖然實驗二在 Calmar 比率訓練結果的提升幅度比其他兩個實驗的訓練結果高出了約 0.6 倍左右，但在實驗二的測試結果跟實驗五及十的測試結果卻沒有大幅度明顯的差異。從表六實驗驗二、五及十的測試結果與其他績效策略的比較，可以看出實驗二、五及十的年均複合增長率皆為正數且有利可圖，實驗二的年均複合增長率比加權股價指數和隨機分配的投資策略都要好，Calmar 比率、夏普比率及累積報酬率在各個績效策略當中則呈現相同幅度的落差。從圖十二來看，雖然本實驗的報酬率並沒有像台新中國通基金那樣大起大落，並且在最高峰的時候來到了 0.78%，但可以看出實驗二的報酬率來得相對平穩，波動幅度也比較小，代理人透過獎勵函數的學習過程中，不是以最大化報酬率為目的，而是在控制最大回撤率的情況下取得最優化報酬率，這達到了一個相對較理想的結果。

DOI:10.6814/NCCU202100964

第二節 未來展望

本研究將金融領域的學理知識融入於強化學習中，關於訓練時間的設定及測試時間限制的部分，由於本研究將測試時間限制於2017年至2020年之間，雖然在2020年年初有一段下滑的段落，然而這一整年的台灣股票市場仍然是屬於牛市的趨勢，雖然在訓練時間當中涵蓋了2007年至2008年的環球金融危機，但由於訓練時間當中屬於熊市段落的股市時間趨於少數，故本研究在進行測試的時候對於該模型是否能真正的應對現實當中熊市的到來仍存有疑慮，這個部分須經過多個不同的時間點來進行測試及驗證。我們認為在這次的試驗中訓練出來的智慧代理人所作出的投資選擇是一個可行的方案，在本研究的實驗當中證明了應用強化學習於金融股票市場是有利可圖的一種投資思考模式，本研究訓練代理人在每次作出投資策略的時候，本研究的設計是從15檔股票當中選取6檔股票作為投資標的，由於在測試時間的最後一年即2020年出現貿易結構性的變化，故投資人在選股的時候應當考慮上市公司所產生的貿易結構作為選股考量，在一個演算法不斷成長的學術環境裡，我們可以考慮擴大股票市場篩選出來的股票數量，讓代理人在作出選擇的時候有更多的選項可以考慮。

關於代理人投資時間長度的部分，在許多文獻並沒有詳細的提及相關內容，Pendharkar et al.（2018）在研究中提及代理人在每次訓練中將進行k步的投資時間，而參數 k 的設定將取決於訓練跟測試數據集的時間長度來決定。在Corazza et al.（2019）的文獻當中，學者設置了兩個不同投資時間去進行測試，而本研究則是在觀察到測試時間長度下設置四個長度不一的投資時間進行測試，在未來的研究當中，投資時間長度這一部分可以考慮以代理人獲利為首要考量，或許也可以嘗試以不同的角度切入讓代理人達到有效獲利的目的來改善這一部分。除此之外，由於本次實驗只有考慮投資股票所獲得的報酬，並沒有顧及到其他影響股票浮動的重要因素，我們可以引入一些基本面、技術面及消息面等股票相關資訊作為考量因素，找出一個包含更多金融股票市場訊息且更加有效的數據集，從中獲取買進訊號可以對代理人在作出投資決定有著關鍵性的影響，在此情況下代理人或許會有不一樣的效果。

DOI:10.6814/NCCU202100964

參考文獻

中文部分

[1] 蔡岳霖(2013)，一個使用遺傳演算法改良之投資組合保險模型之研究，國立高雄大學資訊工程學系碩士論文。

[2] 施承和(2016)，機構投資人與散戶的投資策略之探討，朝陽科技大學財務金融系碩士論文。

[3] 劉俞含(2018)，XGBoost 模型、隨機森林模型、彈性網模型於股價指數趨勢之預測—以台灣、日本、美國為例，國立中山大學財務管理學系碩士論文。

[4] 陳人豪(2018)，台股股利完全填權息關鍵影響因素之研究，國立政治大學資訊科學系碩士在職專班碩士論文。

[5] 陳昱安(2020)，資產配置基於集成學習的多因子模型－以台灣股市為例，國立政治大學金融學系碩士論文。

英文部分

[1] Markowitz, H. (1952). PORTFOLIO SELECTION. The Journal of Finance 7(1): 77-91.

[2] H. Ahmadi (1990). Testability of the arbitrage pricing theory by neural network, IJCNN International Joint Conference on Neural Networks, 1990, pp. 385-393 vol.1, doi: 10.1109/IJCNN.1990.137598.

[3] Nison, S. (1991). Japanese candlestick charting techniques : a contemporary guide to the ancient investment techniques of the Far East, New York Institute Of Finance.

[4] Sharpe, W. (1994). The Sharpe Ratio. Journal of Portfolio Management 21, No.1, Fall: 49-58.

DOI:10.6814/NCCU202100964

[5] Acar, E. and S. James (1997). Maximum loss and maximum drawdown in financial markets. Proceedings of International Conference on Forecasting Financial Markets.

[6] Hochreiter, S. and J. Schmidhuber (1997). LSTM can solve hard long time lag problems. Advances in neural information processing systems.

[7] Moody, J. and L. Wu (1997). Optimization of trading systems and portfolios. Proceedings of the IEEE/IAFE Computational Intelligence for Financial Engineering: 300-307.

[8] Powell, Nicole, et al. (2008). Supervised and Unsupervised Methods for Stock Trend Forecasting. 203 - 205. 10.1109/SSST.2008.4480220.

[9] Chung, J., et al. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555.

[10] Kingma, D. P. and J. Ba (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[11] Cumming, J., et al. (2015). An investigation into the use of reinforcement learning techniques within the algorithmic trading domain, Imperial College London: London, UK.

[12] Gabrielsson, P. and U. Johansson (2015). High-frequency equity index futures trading using recurrent reinforcement learning with candlesticks. 2015 IEEE Symposium Series on Computational Intelligence, IEEE.

[13] Lillicrap, T. P., et al. (2015). Continuous control with deep reinforcement learning. arXiv preprint arXiv:.02971.

[14] Meger, D., et al. (2018). Addressing function approximation error in actor-critic methods. International Conference on Machine Learning(PMLR): 1587-1596.

[15] Pendharkar, P. C. and P. Cusatis (2018). Trading financial indices with reinforcement learning agents. Expert Systems with Applications 103: 1-13.

[16] Kanwar, N. (2019). Deep Reinforcement Learning-based Portfolio Management, Ph.D. Dissertation, The University of Texas at Arlington: Arlington, TX, USA.

DOI:10.6814/NCCU202100964

[17] Liu, L., et al. (2019). On the variance of the adaptive learning rate and beyond. arXiv preprint arXiv:.03265.

[18] Misra, D. (2019). Mish: A self regularized non-monotonic neural activation function. arXiv preprint arXiv:.08681.

[19] Zhang, M., et al. (2019). Lookahead optimizer: k steps forward, 1 step back. Advances in Neural Information Processing Systems.

[20] Corazza, et al. (2019). A comparison among Reinforcement Learning algorithms in financial trading systems, No 2019:33, Working Papers, Department of Economics, University of Venice "Ca' Foscari".

DOI:10.6814/NCCU202100964