國立臺北科技大學

資訊工程系碩士班

碩士學位論文

以 DQN 演算法訓練 ETF 之交易代理人

Training an ETF Trading Agent with Deep Q-Learning

研究生：洪紹晏

指導教授：陳偉凱、尤信程

中華民國一百零八年八月

摘要

論文名稱： 以 DQN 演算法訓練 ETF 之交易代理人

頁數：七十三頁

校所別：國立臺北科技大學 資訊工程系 碩士班

畢業時間：一百零七學年度 第二學期

學位：碩士

研究生： 洪紹晏

指導教授：陳偉凱、尤信程 教授

關鍵詞：深度強化學習、DQN、ETF

近來 AlphaGo 問世，留下輝煌的戰績，使得深度強化學習(Deep Reinforcement Learning, DRL)聲名遠播，眾多 DQN 的應用相繼迎來。強化學習具有最大化累積獎勵(Accumulate Rewards)的特性，與交易員的夢想可以說是一拍即合，本論文嘗試將深度強化學習(Deep Reinforcement Learning, DRL)應用於 ETF (Exchange Traded Funds)的交易，打造一個自動交易的代理人(Agent)。本論文針對強化學習的框架加以定義狀態、動作、獎勵，將獲利的交易模式套用之，利用強化學習的特性，訓練出具有最大化報酬能力的代理人。依據價位等資訊作為狀態，執行買入、持倉、賣出等交易動作，以獲利或虧損幅度給予獎勵反饋。最終訓練出優秀的代理人，依據各種價位資訊，做出最優的決策，為交易員帶來豐腴的報酬。此代理人以每 5 個交易日作一次交易決策，基於收盤價買入、持倉、賣出等交易行為，期望能帶來可觀的報酬。並且本論文以為期 4 年的歷史資料作訓練，隨後與買進持有策略及隨機策略於為期 1 年的資料中進行績效比較，以審視此 DQN 代理人的適用性。根據實驗結果顯示，該代理人具有不錯的獲利能力，並且對於泛化多變的交易市場具有發展前景。

ABSTRACT

Title: Training an ETF Trading Agent with Deep Q-Learning
Pages: 73
School: National Taipei University of Technology
Department: Computer Science and Information Engineering
Time: August, 2019
Degree: Master
Researcher: Shao-Yan Hong
Advisors: Shing-Chern You, Ph.D., Woei-Kae Chen, Ph.D.

Keywords: Deep Reinforcement Learning, Deep Q-Learning, Exchange Traded Funds

AlphaGo recently received a lot of attention after beating professional Go players. Thanks to AlphaGo, deep reinforcement learning is now well known to the academia. Consequently, the deep Q-learning, one of the reforcement learning methods, has been applied to more and more problems. As reinforcement learning trains the agent to maximize accumulated rewards, the agent may act as a successful trader (speculator). In this thesis, we study how to apply reinforcement learning to train an agent for trading ETFs. To use the reinforcement learning, we need to determine states, actions, and rewards in the problem. In our setting, the agent takes the prices as states to make decisions, such as buy, sell or no action, and use portfolio returns as rewards. In this work, the agent performs one action per five days based on the closing prices (and others) of the prior five days of the ETF to be traded. The agent is trained with 4 years of trading records, and is tested on the interval of one year. The performance of the agent is compared with the buy-and-hold strategy. The experimental results show that the proposed trading agent using double deep Q-learning can beat the buy-and-hold in the bear market. Therefore, this approach is suitable for diverse market situations.

致謝

在此論文完成之際，筆者有話要說。首先由衷感謝論文指導教授，陳偉凱教授、尤信程教授、劉建宏教授。在這段期間內，每週貢獻一定時間指導並協助撰寫論文。並且感恩各位教授及口試委員對於本論文的實驗提供幫助以及指導，針對實驗問題的解決提供指引方向。以及寫作過程中提供改善意見的全體 SELAB 實驗室成員，林照晟、何威杭、陳日揚、郭建陞、李修豪、林亮勳、劉宸宗、唐振嚴、陳郁欣、林佳緯、鄭鴻仁、林冠璋、陳巧宜、洪子軒、周士禾、邱楹傑等人。最後感恩在此碩士生涯盡全力支持的父母，以及一同攻略各行情的諸位 ITF 成員。承蒙各位的鼎力相助，在此筆者獻上十二萬分的肝臟以表達感謝之意。

目錄

表目錄

圖目錄

第一章 緒論

1.1. 研究動機與目的

近來深度學習(Deep Learning, DL)的發展越發成熟且盛行，已成家喻戶曉的熱門話題，被廣泛應用於眾多領域，如：影像辨識、語意分析、回歸分析、以及遊戲應用。結合深度學習(DL)及強化學習(Reinforcement Learning, RL)的 AlphaGo 問世，於圍棋遊戲上戰勝各路頂尖好手，留下輝煌的戰績，為深度強化學習(Deep Reinforcement Learning, DRL)打響了名號，在人工智慧(Artificial Intelligence, AI)界種下嶄新的里程碑，開啟後續的新篇章。

由於強化學習具有最大化累積獎勵(Accumulate Rewards)的特性，對於眾多交易員的目的：最大化報酬，應該可以說是一拍即合，倘若應用得體將是夢寐以求的一大利器。本論文嘗試將深度強化學習應用於 ETF (Exchange Traded Funds)的交易，以打造一個自動交易的代理人(Agent)，期盼該代理人能因應交易市場上各狀況，做出較佳的交易決策，為交易員帶來豐腴的獲利。

一般而言，在交易市場中獲利的主要操作模式為：(1)在相對低點的價位買進，即買的相對便宜；並賣在相對高點的價位，即賣的相對昂貴；(2)當價位不優時則不動作，以等待較佳的時機。倘若每次下決策皆很精確，反覆如此操作，則能賺取高低的價差，創造可觀的報酬。

上述交易市場的操作模式可以透過強化學習的框架來實現，依據強化學習的框架，只要能清楚定義 3 個元素：狀態(State)、動作(Action)、獎勵(Reward)，則有機會利用強化學習的特性，訓練出具有最大化報酬能力的代理人，依據各狀態做出較優的動作。如上述操作模式，依據價位的高低，做出交易決策，究竟是要(1)買、(2)賣，或(3)不動作等 3 個動作，以賺取其中的價差，ETF 價位等資訊視為狀態，獎勵則依據交易結果為賺或賠給予反饋，透過設計合適的神經網路架構訓練和獎勵機制，將可訓練出一個可以提高獲益的 ETF 交易代理人。

因此本論文的主要目的為嘗試探討如何妥當定義強化學習框架所需的元素：狀態、動作、獎勵，以期盼利用強化學習特性，訓練出優秀的 ETF 交易代理人，能依據各種 ETF 價位資訊，做出買、或不動作的決策，以創造整體最大化的報酬，獲取豐腴的獲利。

1.2. 論文的組織架構

本論文的組織架構如下：本論文第二章探討背景介紹與相關知識；第三章詳細介紹系統設計；第四章介紹實驗與結果；最後為結論描述本論文的貢獻以及未來的研究方向。

第二章 背景介紹與相關知識

於本章中，將介紹何謂 ETF、評估績效的工具為何、強化學習(Reinforcement Learning, RL)又是何方神聖，以及相關文獻。

2.1. ETF (Exchange Traded Funds)

ETF 為一種金融商品，主要概念如圖 1 所示，結合三要素：指數(Index)、基金(Fund)、股票(Stock)。於本節中，首先將介紹各要素所扮演的角色，接著介紹本論文所使用的 ETF 及歷史資料。

- 指數(Index)：

  ETF 主要以追蹤指數(Index)為原則，不主觀判斷，純被動依比例投資該指數(Index)的成分標的，進而貼近該指數的漲跌幅行情。

- 基金(Fund)：

  投資人若持有 ETF 如同投資該追蹤指數(Index)，意涵持有該指數(Index)的成分標的，具有基金(Fund)的優勢：一籃子標的物，達穩定分散投資風險。

- 股票(Stock)：

  ETF 可於股票(Stock)市場上進行交易，也就是說諸位股民們可以無縫接軌投資 ETF。

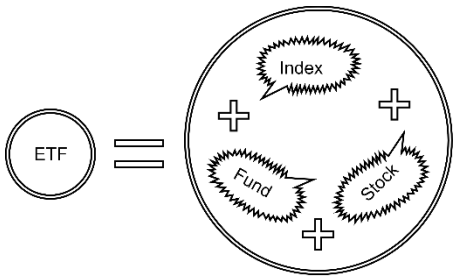


圖 1 ETF 示意圖

以台灣 50ETF (0050)為例，如圖 2 所示，該 ETF 追蹤的指數(Index)為台灣 50 指數(Index)，成分標的為台灣最大的 50 檔股票，倘若股民們持有台灣 50ETF(0050)，意即投資台灣 50 指數(Index)，持有台積電(Taiwan Semiconductor Manufacturing Co., Ltd.) (2330)、鴻海(HON HAI PRECISION IND. CO., LTD.) (2317)、中華電(Chunghwa Telecom Co., Ltd) (2412)等，台灣最大的 50 檔股票。

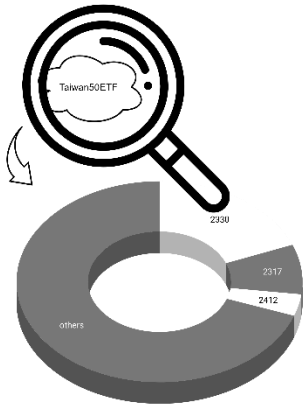


圖 2 台灣 50ETF 示意圖

本論文採用 Chen 於論文 [1]中提及的 ETF，請參閱附錄 A 實驗相關圖表，如表 23 所示包含 38 檔 ETF，各 ETF 取自 MoneyDJ 理財網 [2]。各投資標的分布如圖 3 所示，股票型 28 檔，占比 73.7%；做多型 2 檔，占比 5.3%；債券型 3 檔，占比 7.9%；匯率型 2 檔，占比 5.3%；反向型 1 檔，占比 2.6%；期貨商品 2 檔，占比 5.3%。

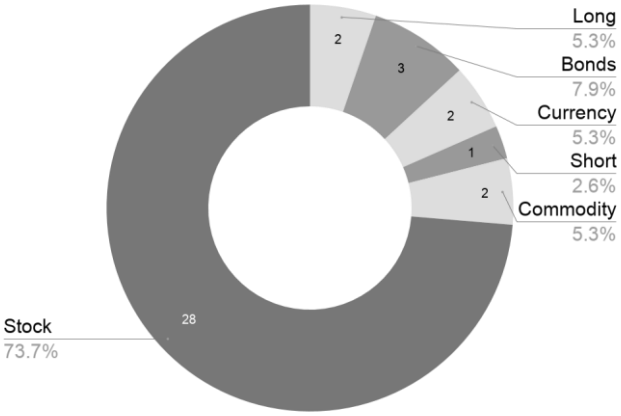


圖 3 ETF 投資標的分布圖

各 ETF 均取用將近 7 年的歷史，2012/02/14~2019/01/29，共計 1750 個交易日，取自 Yahoo Fiance [3]。如表 1 所示，每個交易日皆含 6 個特徵值：開盤價(Open)、最高價(High)、最低價(Low)、收盤價(Close)、調整後的收盤價(Adj Close)、成交量(Volume)，簡易記錄市場上在該交易日的漲跌幅，以下將簡單介紹此 6 特徵的含意。

- 開盤價(Open)：

  如圖 4 所示，開盤價(Open)為當日市場上第 1 筆成交的價位，即該區間走勢最左的價位。

- 最高價(High)：

  如圖 4 所示，最高價(High) 為當日市場上開盤至收盤時間內最高的成交價位，即該區間走勢最高的價位。

- 最低價(Low)：

  如圖 4 所示，最低價(Low) 為當日市場上開盤至收盤時間內最低的成交價位，即該區間走勢最低的價位。

- 收盤價(Close)：

  如圖 4 所示，收盤價(Close) 為當日市場上最後 1 筆成交的價位，即該區間走勢最右的價位。

- 調整後的收盤價(Adj Close)：

  調整後的收盤價(Adj Close)為當日市場收盤後因應各種情況所調整後的價位，如現金分紅、股票分紅等。

- 成交量(Volume)：

  成交量(Volume)為當日市場上開盤至收盤時間內累計成交的單位。

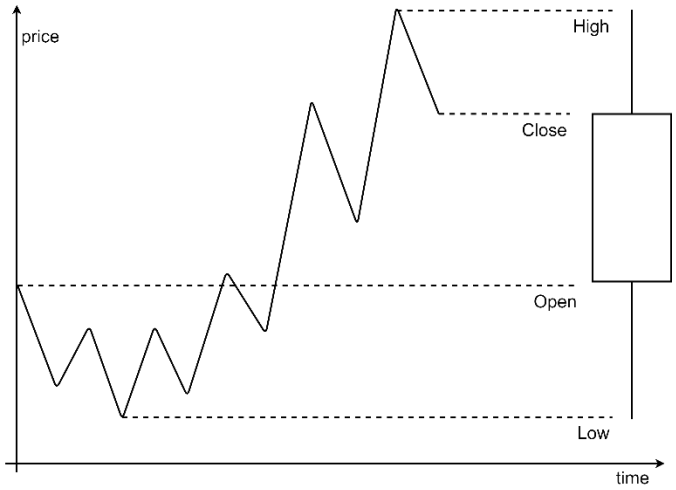


圖 4 K 棒示意圖

表 1 ETF 範例資料

Date	Open	High	Low	Close	Adj Close	Volume
2/14/2012	25.32	25.5	25.23	25.23	21.56118	1500
2/15/2012	25.83	26.02	25.61	25.65	21.92011	18400
2/16/2012	25.75	25.95	25.74	25.95	22.17648	3500
2/17/2012	26.41	26.41	25.75	25.85	22.09102	1200
2/21/2012	25.9	25.95	25.76	25.95	22.17648	1300
2/22/2012	25.99	25.99	25.73	25.91	22.1423	11900
2/23/2012	25.91	25.91	25.73	25.73	21.98847	300
2/24/2012	25.99	26	25.97	25.97	22.19357	1600
2/27/2012	25.9	26.01	25.7	26	22.21921	1800
2/28/2012	26.16	26.45	26.16	26.37	22.5354	1500
2/29/2012	26.49	26.49	25.99	25.99	22.21066	7700
3/1/2012	26.07	26.2	26.02	26.2	22.39013	2200
3/2/2012	25.94	26.05	25.94	26.02	22.2363	4400
3/5/2012	25.76	25.87	25.55	25.6	21.87738	3000
3/6/2012	24.99	25.42	24.85	24.85	21.23643	700
3/7/2012	25.22	25.43	24.65	25.27	21.59536	5500
3/8/2012	25.57	25.82	25.4	25.45	21.74919	2500
3/9/2012	25.81	26.2	25.69	25.87	22.10811	8000
3/12/2012	25.83	25.86	25.62	25.66	21.92865	6000
3/13/2012	25.76	26.12	25.76	26.11	22.31321	5600

2.2. ETN (Exchange Traded Node)

ETN 為一種金融商品，它的特性與 ETF 相近，必須追蹤指數(Index)，並且可於股票(Stock)市場上進行交易。較明顯的差異為，ETF 必須被動投資該追蹤的指數(Index)的成分標的物，而 ETN 則不受此約束，可以將資金挪做其他用途。此外投資人需承擔信用風險，意即若投資的該檔 ETN 不幸下市，落得求償無門的機率頗高。

2.3. 交易策略及評估工具

於本節中，將介紹兩項交易策略以及一項評估績效的工具。何謂波段交易(Swing Trade)，而買進持有(Buy and Hold)策略又為何物，投資報酬率(Return on Investment, ROI)又該如何計算。最後，本論文以買進持有(Buy and Hold)策略作為實驗的對照組。

- 波段交易(Swing Trade)：

  波段交易(Swing Trade)策略如圖 5 所示，在該坐標系上，橫軸代表時間，縱軸代表價位。判定價位走勢後，期望買在低點，賣在高點。

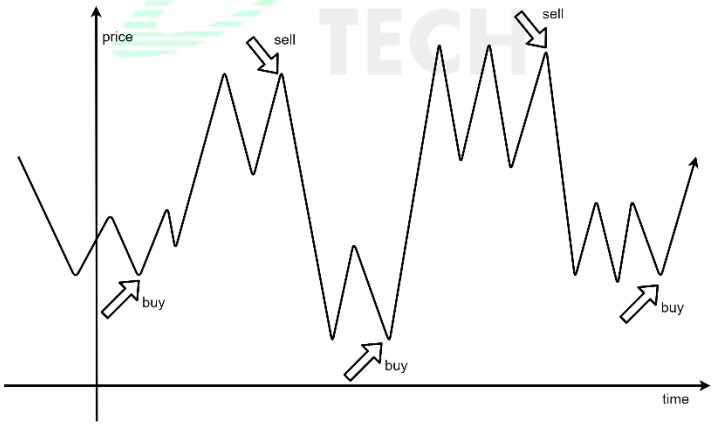


圖 5 波段交易示意圖

- 買進持有(Buy and Hold)：

  買進持有(Buy and Hold)策略如圖 6 所示，在該坐標系上，橫軸代表時間，縱軸代表價位。相較於波段交易(Swing Trade)，買進持有(Buy and Hold)策略，不作頻繁交易，一旦買進之後，則不賣出。

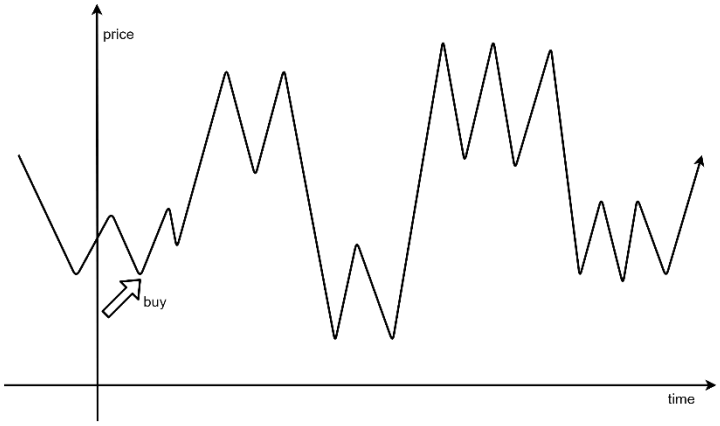


圖 6 買進持有策略示意圖

- 投資報酬率(Return on Investment, ROI)：

  本論文評估績效的工具：ROI，如圖 7 所示，在該坐標系上，橫軸代表時間，縱軸代表資產。在初始時間點上，資產記為$asset_{initial}$，隨著時間的推移，因投資有時賺有時賠，資產亦起起伏伏。經過一段時間後，該資產記為$asset_{final}$，ROI 為終資產扣除本金後對初始資產的比值，如公式(1)。其中資產(Asset)包含現金(Cash)及持股(Position)，計算資產(Asset)時，本論文以收盤價(Close)依據，如公式(2)，現金(Cash)加上持股數(Position)乘上當前價位(Close)：

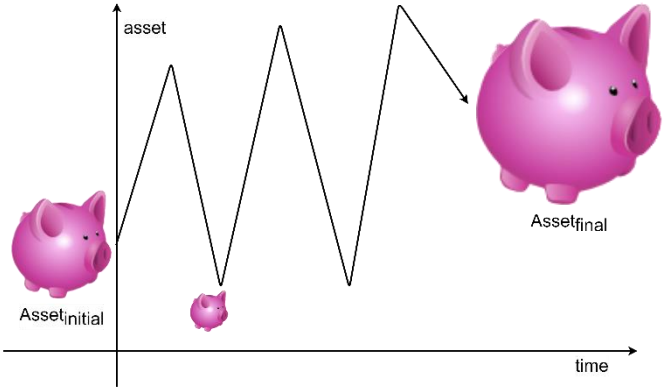


圖 7 ROI 示意圖

$$ROI = \left(\frac{asset_{final}}{asset_{initial}} - 1\right) \times 100\% \quad (1)$$

$$asset = cash + position \times price_{close} \quad (2)$$

2.4. Reinforcement Learning

2.4.1. 強化學習簡介

在強化學習(Reinforcement Learning, RL) [4]，代理人(Agent)透過學習狀態(State)與動作(Action)的對應，以達最大化報酬(Reward)為目的。代理人(Agent)並無報馬仔告知該採取什麼動作(Action)，相反地，僅透過與環境(Environment)多次互動，嘗試各種動作(Action)，不斷的嘗試錯誤，以習得何種動作(Action)足以觸發最多獎勵(Reward)回饋。

該互動如圖 8 所示，代理人(Agent)將在環境(Environment)給出獎勵(Reward)及狀態(State)後，對環境(Environment)做出下一個動作(Action)，而環境(Environment)則針對代理人(Agent)所做出的動作(Action)給予獎勵(Reward)及狀態(State)，如此不停演化，代理人(Agent)逐漸習得較優的動作(Action)，往最大化報酬(Reward)方向移動。

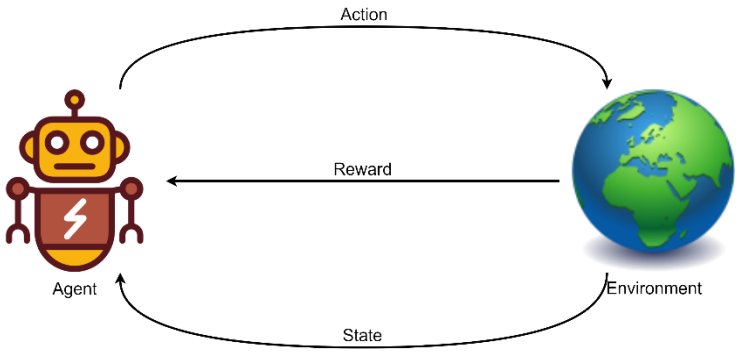


圖 8 代理人與環境的互動

順帶一提，強化學習(Reinforcement Learning, RL)的家族中，存在不少演算法，如：Policy Gradient、SARSA、Q-Learning [5]，及 DQN [6]等。其中以 Q-Learning 較廣為人知，有關 Q-Learning 如何使用表格法(Tabular Method)，以及其所面臨的議題，請參閱 2.4.2，DQN

如何以神經網路(Neural Network)應對此議題，以及 DDQN 又做了什麼優化，請參閱 2.4.3 及 2.4.4。最後，本論文取用 DDQN 實作。

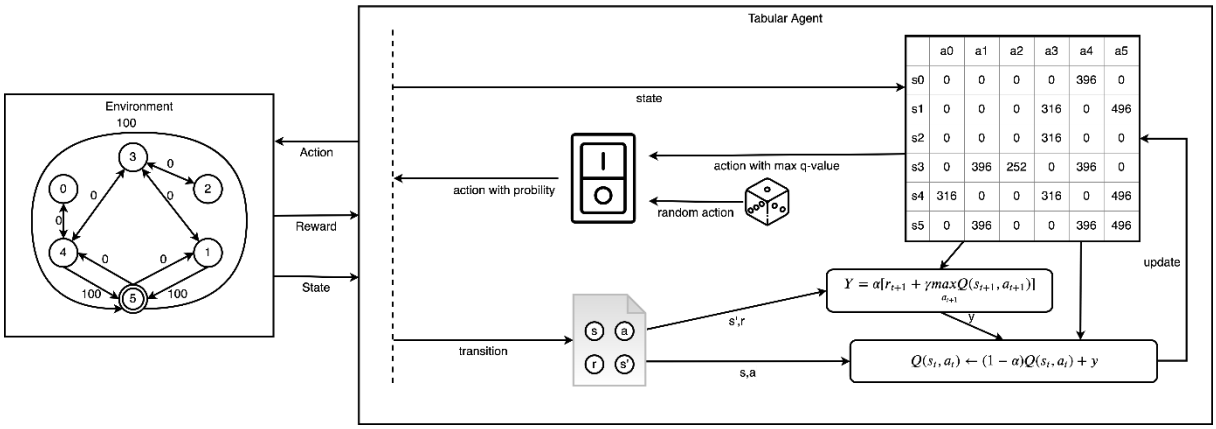
2.4.2. Q-Learning

在強化學習(Reinforcement Learning, RL)，其中一支家族為 Q-Learning，該核心概念為表格法(Tabular Method)，即根據 Q 表(Q-Table)決定動作(Action)。本節將以簡單的房間範例介紹 Q-Learning，並點出表格法(Tabular Method)的瓶頸。

該範例如圖 9 所示，狀態數為 6 如各房間號碼：0、1、2、3、4、5。動作數為 6 代表前進至下一房間的號碼：0、1、2、3、4、5。環境(Environment)的獎勵(Reward)反饋如連接線段所示：0、100。目標為引導代理人(Agent)走到終點號碼 5 的房間，以獲得最大獎勵。

Q-Table 如圖 9(右)所示，首先，代理人(Agent)於各房間查詢 Q 表(Q-Table)。接著，依機率挑選 Q 值分數最優的動作(Action)執行，或者執行隨機動作(Random Action)，走到下一房間(Next State)並獲及獎勵(Reward)。最後，挑選最優 Q 值以更新 Q 表(Q-Table)上一狀態對動作(State-Action)的欄位，如公式(3)。經過多番嘗試後，Q 表(Q-Table)將更成熟，引導代理人(Agent)走向較優路徑。

$$Q(s_t, a_t) \leftarrow (1-\alpha)Q(s_t, a_t) + \alpha[r_{t+1} + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1})] \tag{3}$$



	a0	a1	a2	a3	a4	a5
s0	0	0	0	0	396	0
s1	0	0	0	316	0	496
s2	0	0	0	316	0	0
s3	0	396	252	0	396	0
s4	316	0	0	316	0	496
s5	0	396	0	0	396	496

圖 9 Q-Learning 示意圖

試著根據如上之範例推測，一旦狀態空間(State Space)或動作空間(Action Space)規模擴大，則難以使用表格法(Tabular Method)，為 Q-Learning 所須面臨的議題。面對此議題，DQN 誕生了，其以神經網路(Neural Network)取代表格法(Tabular Method)，有關 DQN 的深入細節，請參閱 2.4.3 節。

2.4.3. Deep Q-Learning

由 Q-Learning 所衍伸的議題：狀態爆炸，即當狀態(State)多到難以使用有限的表格記錄 Q 值。針對此議題，一種應對方法為套用神經網路，於是 DQN [6]誕生了，以下將介紹 Mnih 等人於論文[6]中提出的 DQN 的演算法及主打兩要素：

- Experience Replay
- Target Network

DQN 架構如圖 10 所示，該代理人(Agent)具有兩套相同架構的神經網路：估計網路(Evaluate Network)及目標網路(Target Network)，一個經驗池(Replay Memory)。

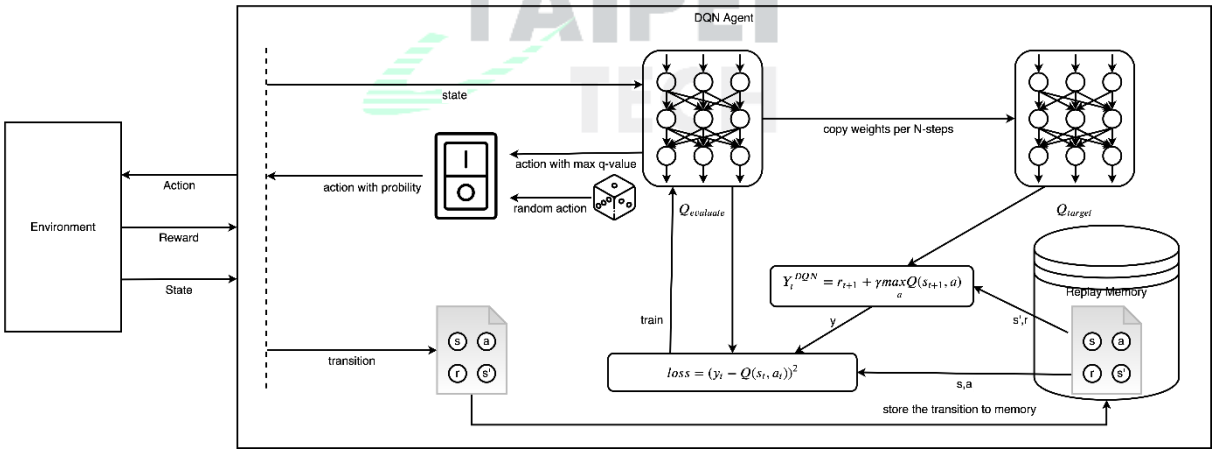


圖 10 DQN 架構圖

- Experience Replay：

  使用經驗池(Replay Memory)存放代理人(Agent)與環境(Environment)互動的經驗，以便代理人(Agent)回顧做使用，提高各樣本重用性。代理人(Agent)回顧時，將從經驗池(Replay Memory)中隨機生成樣本。此舉打散了各樣本的時間序，減

少相依性，有助於神經網路的訓練。經驗所需情報如：現態(State)、動作(Action)、獎勵(Reward)、下一態(Next State)。

- Target Network：

  使用兩套相同架構的神經網路：以目標網路(Target Network)產生目標Y_t^{DQN}值，如公式(4)，接著訓練估計網路(Evaluate Network)。此外，相隔一定的步數(Steps)，才將估計網路(Evaluate Network)的權重(Weights)拷貝到目標網路(Target Network)，藉此降低兩網路的相關性。

$$Y_t^{DQN} \equiv R_{t+1} + \gamma \max_a Q(S_{t+1}, a; \widehat{w}_t) \tag{4}$$

- 演算法：

該演算法如圖 11 所示：首先初始化記憶池(Replay Memory)及 $Q_{evaluate}$、Q_{target} 兩個網路，以若干集(Episode)訓練代理人(Agent)，每集(Episode)具有若干步(Step)，每一步(Step)有 ε 的機率隨機選擇動作(Action)，或是選擇估計網路 $Q_{evaluate}$ 產出最優 Q 值的動作(Action)，執行此動作(Action)後觀察得到的獎勵(Reward)回饋與下一狀態(State)，將此轉換記錄(Transition)放進記憶池(Replay Memory)M，從記憶池(Replay Memory)M 中隨機取出若干筆轉換計錄(Transition)，使用目標網路 Q_{target} 算出目標 y 以訓練估計網路 $Q_{evaluate}$，每隔一段步數(Step)同步兩網路的權重。

Algorithm 1 DQN with experience replay

Initialize replay memory M to capacity N
Initialize action-value function Q with random weights w
Initialize target action-value function $\hat{Q}$ with weights $\hat{w} = w$
for *episode* = 1 to E **do**
  Initialize sequence $s_1 = \{x_1\}$ and preprocessed sequence $p_1 = p(s_1)$
  for $t = 1$ to T **do**
    With probability ϵ select a random action a_t
    otherwise select $a_t = argmax_a Q(p(s_t), a, w)$
    Execute action a_t in emulator and observe reward r_t and image x_{t+1}
    Set $s_{t+1} = s_t, a_t, x_{t+1}$, and preprocess $p_{t+1} = p(s_{t+1})$
    Store transition (p_t, a_t, r_t, p_{t+1}) in M
    Sample random minibatch of transitions (p_i, a_i, r_i, p_{i+1}) from M
    if episode terminates at step i+1 **then**
      Set $y_i = r_i$
    else
      Set $y_i = r_i + \gamma max_{\acute{a}} \hat{Q}(p_{i+1}, \acute{a}; \hat{w})$
    end if
    Perform a gradient descent step on $(y_i - Q(p_i, a_i; w))^2$ with respect to the network parameters w
    Every C steps reset $\hat{Q} = Q$
  end for
end for

圖 11 Mnih 等人的 DQN 演算法 [6]

2.4.4. DDQN

DQN 具有一些不穩定因素存在，如：過度估計(Overestimate)。針對此議題，DDQN (Double DQN) [7]提出進一步改善，針對目標 y 值的計算公式做出調整。DDQN 架構如圖 12 所示，該代理人(Agent)跟同樣具有兩套相同架構的神經網路：估計網路(Evaluate Network)及目標網路(Target Network)，一個經驗池(Replay Memory)。簡略說明如下：

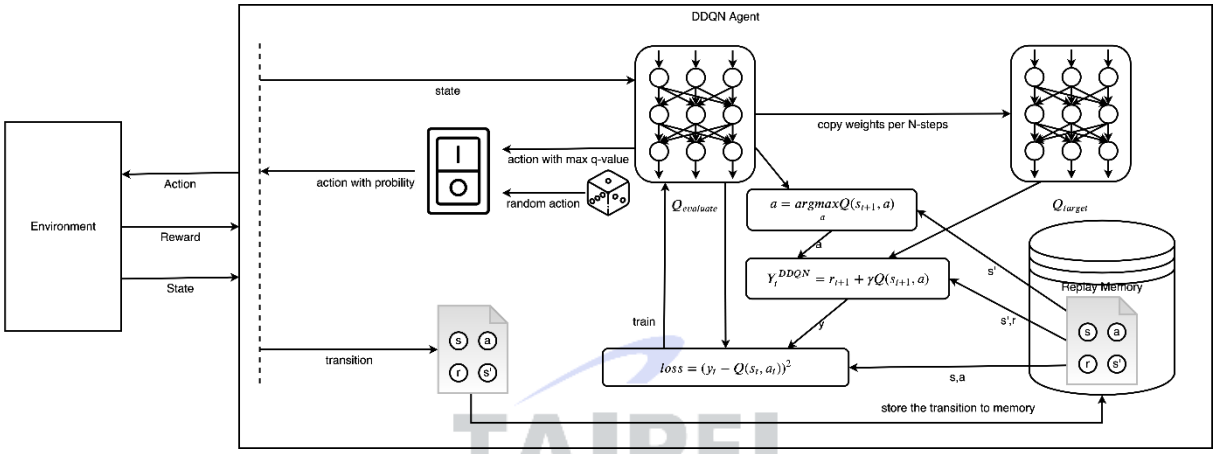


圖 12 DDQN 架構圖

- DQN target：

  在 DQN 中，僅以目標網路(Target Network)參與計算目標Y_t^{DQN}，並訓練估計網路(Evaluate Network)，如 2.4.3 節中的公式(4)

- DDQN target：

  DDQN 做出的改善為使用兩網路參與計算：以估計網路(Evaluate Network)選出最優動作(Action)，爾後將此動作(Action)代入目標網路(Target Network)求得目標Y_t^{DDQN}，並訓練估計網路(Evaluate Network)，如公式(5)。

$$Y_t^{DDQN} \equiv R_{t+1} + \gamma Q(S_{t+1}, \underset{a}{\operatorname{argmax}}\, Q(S_{t+1}, a; w_t), \hat{w}_t) \tag{5}$$

2.5. 相關文獻探討

本論文的訓練方法參考至 Chen 的論文 [1]及劉的論文 [8]，並套用 Gao 於論文 [9]提及的有效動作(Valid Action)概念。各文獻方法的比較如表 2 所示，在狀態(State)方面，Chen 採用價位資訊，劉則採用報酬、價位、持股等資訊。在動作(Action)方面，Chen 採用買(Buy)、賣(Sell)、持倉(Hold)等，而劉則採用調整比例之類的動作。在獎勵(Reward)方面，Chen 使用 ROI 相關的計算方法，劉則使用夏普指數(Sharpe Ratio)。最後，衡量績效方面，Chen 使用 ROI 測量，劉則以夏普指數(Sharpe Ratio)測量。本論文較接近 Chen 的方法，與之主要的差異為：除了價位之外，額外給予現金及持股及兩條交易訊號作為狀態，並限制代理人僅能執行有效的動作，有關詳細的介紹，請參閱 0。

表 2 文獻方法比較表

	This Work	**Chen's Approach [1]**	**Liu's Approach [8]**
State	Price, Cash, Position, Signals	Price	Profit, Price, Position
Action	Valid Buy, Hold, Sell	Buy, Hold, Sell	Adjust Proportion
Reward	ROI Based	ROI Based	Sharpe Ratio Based
Evaluation Tool	ROI	ROI	Sharpe Ratio

2.5.1. Chen 的 DQN 自動交易系統

Chen 於論文 [1]，以 DQN 實做自動交易系統，應用於 ETF 這一類金融商品，每 20 個交易日，參考過去 20 日歷史，做一次買賣交易決策。以下將簡單分為三部分介紹：首先，介紹如何使用資料集。接著，交代強化學習(Reinforcement Learning, RL)三要素如何設計。最後，交代實驗之比較結果。

- 資料集配置：

  該資料集取用為期 5 年的 ETF，共計 40 檔，訓練集為其中 20 檔 ETF，測試集為剩餘 20 檔 ETF。訓練集中僅取前 4 年作訓練，而不包含第 5 年；測試集中僅取第 5 年作測試，而不包含前 4 年。

- 狀態(State)：

Chen 的代理人(Agent)每次作交易決策時，以過去 20 個交易日的 6 個特徵為依據，包含開盤價(Open)、最高價(High)、最低價(Low)、收盤價(Close)、調整後的收盤價(Adj Close)、成交量(Volume)。

- 動作(Action)：

  Chen 假設該代理人(Agent)僅有 5 個動作，如表 3 所示。動作 0 ($Action_0$)為買進 20 個單位的 ETF；動作 1 ($Action_1$)為買進 10 個單位的 ETF；動作 2 ($Action_2$)為持有不動作；動作 3 ($Action_3$)為賣出 10 個單位的 ETF；動作 4 ($Action_4$)為賣出 20 個單位的 ETF。

表 3 Chen 的代理人動作集合

Action	Purchase Quantity
0	20 units
1	10 units
2	hold
3	-10 units
4	-20 units

- 獎勵(Reward)：

  Chen 設計環境(Environment)反饋給代理人(Agent)的獎勵(Reward)為當前資產對前資產的比值再取對數，如公式(6)。

$$reward = \log\left(\frac{asset_{current}}{asset_{last}}\right) \tag{6}$$

- 實驗結果：

  該代理人(Agent)以買進持有策略(Buy and Hold)為比較對象，以平均數據而言，績效超越買進持有(Buy and Hold)策略。

2.5.2. 劉的 DQN 動態資產配置

劉於論文 [8]，將 DQN 應用於動態資產配置， 以 ETF 金融商品為例，取 2 負相關股票型及債券型 ETF 做為投資組合，依據過去 21 日的歷史，調整資產比例。以下將簡單分為兩

部分介紹：如何使用資料集，強化學習(Reinforcement Learning, RL)三要素又如何設計。最後，交代實驗的比較結果。

- 資料集配置：

  首先，以股票型 ETF 19 檔，及債券型 ETF 15 檔，作所有組合(All Combination)，將產生 285 種投資組合樣本。接著，依 14：3：3 的比例分配此 285 種投資組合樣本，依序為訓練集(Train) 199 種、驗證集(Validate) 43 種、集測試集(Test) 43 種。

- 狀態(State)：

  劉的代理人(Agent)每次做決策時，以兩項資產前 21 天至當日為依據，每檔 ETF 各取至當天的日報酬：$\{r(\mathrm{t}-\mathrm{i})|0 \leq \mathrm{i} \leq 21\}$，當天及前一天的調整後收盤價(Adj Close)：{price(t-1), price(t)}，前一日的持有股數：{position(t-1)}，共計 50 項數據。

- 動作(Action)：

  該代理人(Agent)具有 25 種動作，以下僅列出部分作參考，如表 4 所示，正值代表賣出股票型 ETF 並將相同金額買進債券型 ETF；負值代表賣出債券型 ETF 並將相同金額買進股票型 ETF。

表 4 劉的代理人動作集合

Action	Description
9	-7.5%
10	-5.0%
11	-2.5%
12	0.0%
13	2.5%
14	5.0%
15	7.5%

- 獎勵(Reward)：

  劉設計環境(Environment)反饋給代理人(Agent)的獎勵(Reward)為夏普指數(Sharpe Ratio)，如公式(7)，其中σ_p為投資組合標準差、$E(R_p)$投資組合預期報酬率、R_f為無風險報酬率，意涵承受單位風險將產生的報酬。

$$\text{Sharpe Ratio} = \frac{[E(R_p) - R_f]}{\sigma_p} \tag{7}$$

- 實驗結果：

  以買進持有策略(Buy and Hold)為比較對象，依據夏普指數(Sharpe Ratio)為衡量標準，劉的代理人(Agent)績效有機率超越買進持有(Buy and Hold)策略。

2.5.3. Gao 的 DQN 交易遊戲

Gao 於論文 [9]，將有效動作(Valid Action)的概念套用於 DQN，藉此排除無效動作(Invalid Action)或非法動作(Illegial Action)，以應用於時間序列的交易遊戲中。以下將針對重點三處，簡述 Gao 如何套用有效動作(Valid Action)的概念，並將其應用於本論文中。首先，選擇隨機動作的程序。接著，選擇最優動作的程序。最後，計算目標值y_i的程序。

- 選擇隨機動作：

  如圖 13 所示中第 3 行，由現態(State)有效的動作(Valid Action)中，隨機挑選動作。

- 選擇最優動作：

  如圖 13 所示中第 4 行，由現態(State)有效的動作(Valid Action)中，挑選 Q 值最優的動作。

- 計算目標y_i：

  如圖 13 所示中第 11 行，依據下一狀態(Next State)的有效動作(Valid Action)中，挑選最優 Q 值計算目標y_i。

Algorithm 2 DQN with valid action

for *episode* = 1 to E **do**
  for $t = 1$ to T **do**
    With probability ϵ select a random valid action a_t
    otherwise select the valid action a_t that maximize predicted Q
    Given a_t,emulator returns reward r_t and new state s_{t+1}
    Store transition (s_t, a_t, r_t, s_{t+1}) in M
    Sample random minibatch of transitions (s_i, a_i, r_i, s_{i+1}) from M
    if episode terminates at step i+1 **then**
      Set $y_i = r_i$
    else
      Set $y_i = r_i + \gamma \underset{a_{valid}}{max} Q(s_{i+1}, a)$
    end if
    Perform a gradient descent step on $(y_i - Q(s_i, a_i))^2$
  end for
end for

圖 13 Gao 的 DQN 演算法 [9]

第三章 系統設計

本論文採用 DDQN 來進行強化學習，此章將介紹資料集的配置、RL 三要素、及神經網路等設計。

3.1. 環境(Environment)

本論文依據強化學習(Reinforcement Learning, RL)設計的環境(Environment)如圖 14 所示，後續將依序介紹，ETF 資料集的配置情形，以及狀態(State)、動作(Action)、獎勵(Reward)的設計。

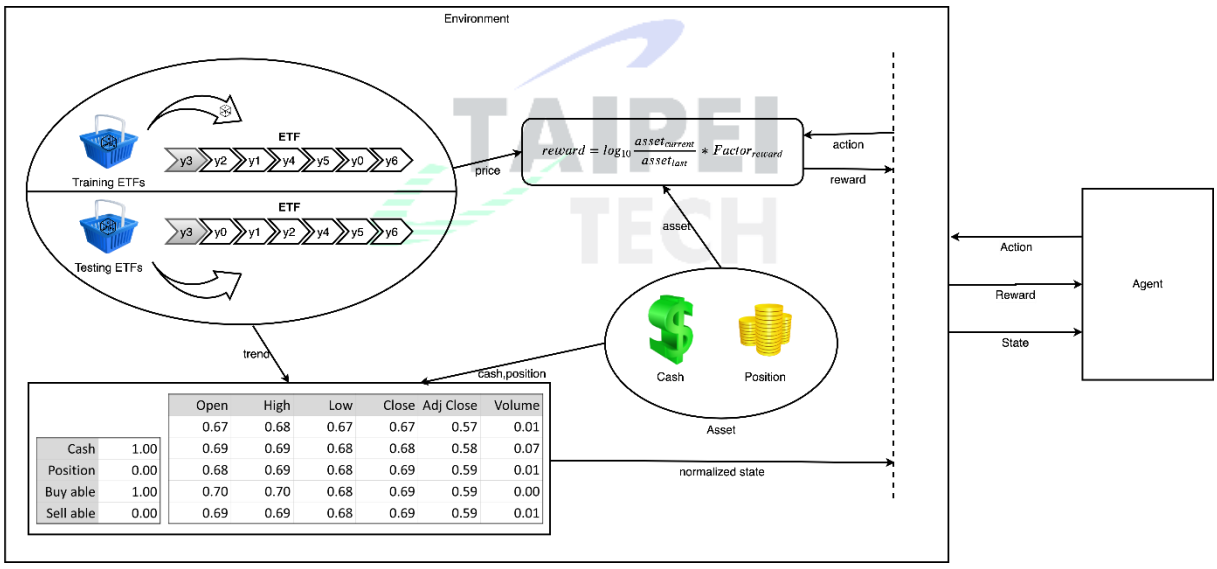


圖 14 ETF 環境示意圖

- ETF 資料配置：

有關資料來源，請參閱本論文第二章中 2.1 節，共計有 38 檔 ETF，為期 1750 個交易日，也就是 7 年份的資料。一般來說，作深度學習(Deep Learning, DL)時會將資料集分為三部分：訓練集、驗證集、測試集，並且以訓練集搭配驗證集調整超參數(Hyperparameter)及模型，

使用測試集評估其泛化性(Generalizability)。本論文效法此傳統，將上述提及的 38 檔 ETF 隨機分為兩類各 19 檔，19 檔為訓練集，剩餘 19 檔為測試集。

此外，由於 Chen 的論文 [1]以 4 年份的資料作訓練(Train)，以 1 年份的資料作測試。為後續的實驗結果能與之比較，本論文將此 7 年份的資料以年為單位切為 7 等份，每份皆有 250 個交易日的資料，並如圖 15 所示，由訓練集中每檔 ETF 皆從 7 年中挑選同 1 年作驗證(Validate)，剩餘的 6 年每檔 ETF 皆隨機挑選 4 年作訓練(Train)，最後由測試集中每檔 ETF 皆挑選與驗證(Validate)的同 1 年作測試 (Test)，而不是隨機挑選。

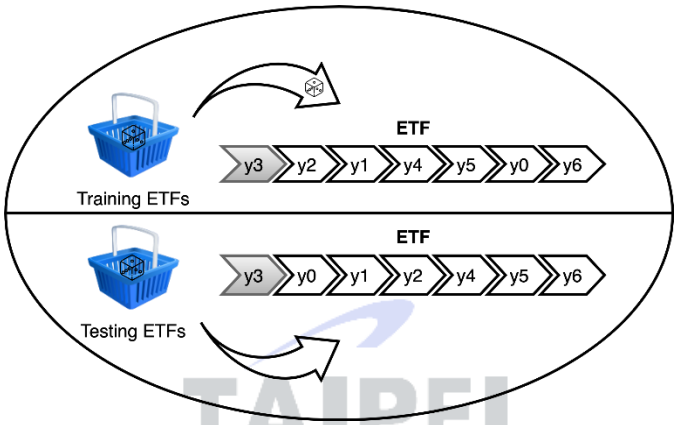


圖 15 ETF 資料配置示意圖

- 狀態(State)：

依據 DQN 要素之一：狀態(State)，首先參考自 Chen 的論文 [1]，以單純的歷史價位作為資訊，並無帳戶的現金(Cash)及持股(Position)等訊息。考量到該交易的世界中，現金(Cash)及持股(Position)應該是屬於滿重要的情報，地位不亞於價格的份量。再者，隨著不斷的交易，現金(Cash)及持股(Position)變化萬千，單憑價格資訊不足以描述唯一的狀態。為此本論文除了原本舊有的價格資訊基礎上，額外多賦予現金(Cash)及持股(Position)的訊息。

另外，由於本論文僅以單向作多交易為基礎，放空交易牽涉融券保證金等相關複雜的操作，暫且不考慮作空的選項，以達簡化問題的複雜度。且在交易的過程中有機率作無效的動作(Invalid Action)：若現金(Cash)不足的條件下，仍要做買(Buy)的動作會造成違約交割；若持股(Position)不足的條件下，仍要做賣(Sell)的動作也會造成違約交割。而因為訓練過程中，有違約交割的困擾，為處理此議題，本論文加入兩條交易訊號線做為情報，告知代理人究竟能不能買賣，期盼代理人(Agent)能學會避開 Invalid Action。不料實驗結果顯示：無

論是否加入訊號線，皆不能 100%防止代理人執行 Invalid Action。在此之後本論文從 Action 方面著手，排除 Invalid Action，直接限制代理人(Agent)僅能做有效動作(Valid Action)，即不會造成違約交割的動作，使得代理人(Agent)不必分神學習避開 Invalid Action，而是專注學習如何獲利。

如同上述所描述，除了價格資訊之外，仍須有現金(Cash)、持股(Position)、交易訊號等情報，本論文將狀態分為兩部分，一為與市場行情有關的特徵，也就是舊有的資訊，如價位、成交量等；二為與帳戶有關的特徵，也就是額外賦予的情報，如現金、持股、交易訊號等。

首先是第一部分，處理與市場有關的特徵：如第二章的2.1節所描述，該歷史資料中所記錄的每個交易日皆包含 6 個特徵，開盤價(Open)、最高價(High)、最低價(Low)、收盤價(Close)、調整後的收盤價(Adj Close)、以及成交量(Volume)。順代一提，在作訓練之前，將資料作適當的前處理以達有效的訓練，也是重要的鐵則。本論文將開盤價(Open)、最高價(High)、最低價(Low)、收盤價(Close)、調整後的收盤價(Adj Close)為一組，另一組為成交量(Volume)，分別帶入正規化(Normalization)的程序，如公式(8)，其中X_{max}為該檔 ETF 中最大值。

$$f(x) = \frac{x_i}{X_{max}} \tag{8}$$

接著，如表 5 所示，取用 ETF 過去 5 個交易日的 6 個特徵，作為狀態(State)的一部份，形狀(Shape)為$5 \times 6 \times 1$，包含開盤價(Open)、最高價(High)、最低價(Low)、收盤價(Close)、調整後的收盤價(Adj Close)、成交量(Volume)。

表 5 行情狀態示意表

Index	Open	High	Low	Close	Adj Close	Volume
0	0.67	0.68	0.67	0.67	0.57	0.01
1	0.69	0.69	0.68	0.68	0.58	0.07
2	0.68	0.69	0.68	0.69	0.59	0.01
3	0.70	0.70	0.68	0.69	0.59	0.00
4	0.69	0.69	0.68	0.69	0.59	0.01

至於第二部分，如表 6 所示：提供帳戶的訊息，作為狀態(State)的一部份，形狀(Shape)為4，包含現金(Cach)、持股數(Position)、可買的訊號(Buy able)、可賣的訊號(Sell able)。當帳戶有現金(Cash)足以買股時，可買的訊號為1否則為 0；當還有持股數(Position)未兌現時，可賣的訊號為 1 否則為 0。正規化(Normalization)的程序則是根據初始資產，將現金(Cash)、持股(Position)分別除上初始資產，如公式(9)。而交易訊號則因數值已介於 0~1 的區間，在此不作額外的正規化程序。

$$f(x) = \frac{x_i}{\text{Cash}_{initial}} \tag{9}$$

表 6 帳戶狀態示意表

	Value
Cash	1.00
Position	0.00
Buy able	1.00
Sell able	0.00

- 動作(Action)：

依據 DQN 框架的要素之二：動作(Action)，首先回顧第二章中 2.5 節找到表 3，Chen 的方法中 [1]為代理人(Agent)定義 5 種動作(Action)：分別為買進 20 個單位的 ETF、買進 10 個單位的 ETF、持有不動作、賣出 10 個單位的 ETF、賣出 20 個單位的 ETF。可惜的是僅有交易量的大小，並未提及究竟以那個價位作交易：開盤價(Open)、最高價(High)、最低價(Low)、收盤價(Close)、調整後的收盤價(Adj Close)。

由於交易的單位為離散數據，該代理人除了要學習識別趨勢之外，還得學習交易量大小的拿捏，負擔稍嫌過重。此外，由於必須學每個動作的 Q 值，訓練難易度隨著動作數的擴大而增加。實驗結果發現減少動作數，可以達到比較有效的訓練。在此本論文總括此 5 個動作(Action)，歸納為 3 個動作(Action)，分別為買入、持有、賣出，暫且不討論交易量大小，以達簡化問題，並減少代理人的學習負擔，學會依趨勢作交易即可，無需考慮交易量的問題。順代一提，本論文交易時皆以收盤價(Close)為依據。

另外，值得討論的議題是，若是當前的現金(Cash)及持股(Position)不足以完成該交易的程序時，又該如何處理？比方說必須要買的時候，現金(Cash)不夠；或者是必須要賣的時

候，反而持股數(Position)不夠等無效動作(Invalid Action)，究竟是該忽略亦或是另有安排？Chen 的方法中 [1]並無特別提及。針對此議題，本論文將套用有效動作(Valid Action)的概念，該想法請參閱第二章中 2.5 節，Gao 的方法中 [9]，針對演算法作了些調整，過濾無效的動作(Invalid Action)，使代理人(Agent)僅能作有效的動作(Valid Action)。

如同上述所描述，本論文訓練的代理人(Agent)，可以做的有效動作(Valid Action)如表 7 所示。動作 0 ($Action_0$)為全數買進，即將現金(Cash)依收盤價(Close)全數換成持股(Position)。動作 1 ($Action_1$)為不動作，即持有之股數(Position)及現金(Cash)不變。動作 2 ($Action_2$)為全數賣出，即將持有之股數(Position)全數依收盤價(Close)換回現金(Cash)。此外，全數買進賣出的動作設計，隱含著資金無限大，排除資金大小的問題。並且在此機制之下，停損等同執行賣出的動作。

該代理人(Agent)每 5 天作 1 次交易決策，直至結算日來臨時，該代理人(Agent)將賣出全部持有的 ETF，以將持股數歸零，並計算該代理人(Agent)可以創造多少獲利，又是否能賺錢。順代一提，在本論文中評估績效的方法如第二章中 2.3 節所描述，以 ROI 作為衡量工具，即終資產扣除本金後對初始資產的比值。

表 7 代理人動作集合

Action	Description
0	Buy Entirely
1	No Action
2	Sell Entirely

- 獎勵(Reward)：

依據 DQN 框架的要素之三：獎勵(Reward)，首先回顧第二章中 2.3 節，熟悉一下所謂資產的計算方法，即現金(Cash)加上持股數(Position)乘上當前價位。接著回顧同樣第二章中 2.5 節，Chen 的方法中 [1]定義的獎勵函數(Reward Function)為現資產 $asset_{current}$ 除上前資產 $asset_{last}$ 再取對數，可惜的是並未說明該底數為何，為復現(Reproduce)增添一點難度。

本論文所定義之獎勵函數依據上述的方法為基礎，以當前資產對前資產的比值，再取底數為 10 的對數後放大 1 個倍率(Reward Factor)，如公式(10)。一方面取對數後具備正負獎勵(Reward)值的可能，也許針對賺錢能給予正向獎勵(Reward)，若是賠錢則給予負向獎勵

(Reward)。二方面放大倍率拉開級距，避免誤差值帶來的影響。至於該獎勵放大倍率(Reward Factor)為何，究竟是否有奇效，將於 0，也就是實驗的章節中作探討。

$$reward = log_{10}\left(\frac{asset_{current}}{asset_{last}}\right) \times Factor_{reward} \quad (10)$$

3.2. DQN 參數

該代理人使用的參數如表 8 所示，初始現金為 10 個單位，初始持有股數為 0，總訓練步數(Training Steps)為 4,000,000，每隔 100 步(Step)則將估計網路(Evalutae Network)的權重拷貝至目標網路(Target Network)，折扣率(Discount Rate)為 0.95，記憶池(Replay Memory)大小為 100,000，決定動作(Action)的策略為 Linear annealing epsilon greedy，隨機係數 ε 由 1.0 線性衰減至 0.1，每次從記憶池(Replay Memory)取出 32 筆記憶做 Experience Replay。

表 8 DQN 代理人的參數

	Value
$Cash_{initial}$	10
$Position_{initial}$	0
Training Steps	4,000,000
Copy Steps	100
Gamma	0.95
Memory Size	100,000
Policy	Linear annealing epsilon greedy
$Epsilon_{max}$	1.0
$Epsilon_{min}$	0.1
Batch Size	32

3.3. 神經網路架構及參數

本論文所使用的神經網路架構如圖 16所示，該網路的輸入為兩部分：Input1 及 Input2。首先 Input1 經過三道捲積層(Convolution Layer)，然後與 Input2 合併進入三道全連接層(Dense Layer)，最後為神經網路輸出的一組 Q 值。

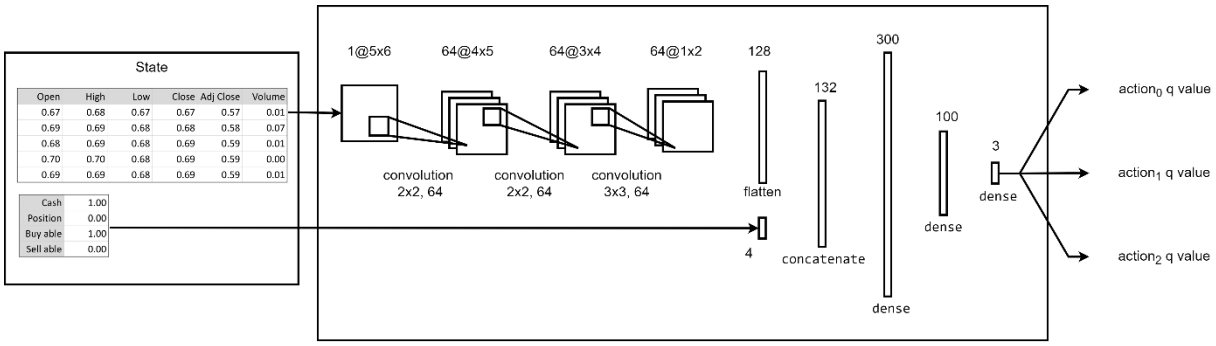


圖 16 神經網路架構

首先，兩個輸入層(Input Layer)的參數如表 9 輸入層的超參數所示，Input1 的形狀(Shape)為5 × 6 × 1，Input2 的形狀(Shape)為 4。

表 9 輸入層的超參數

	Shape
Input	5 x 6 x 1
Input	4

接著，三道捲積層(Convolution Layer)如表 10 所示，第一道捲積層(Convolution Layer)使用 64 個捲積核(kernel)，每個核(kernel)形狀(Shape)為2 × 2，激勵函式(Activation function)為線性整流單元(Rectified Linear Unit ,Relu)，權重初始化使用 he_normal。第二道捲積層(Convolution Layer)使用 64 個捲積核(kernel)，每個核(kernel)形狀(Shape)為2 × 2，激勵函式(Activation function)為線性整流單元(Rectified Linear Unit ,Relu)，權重初始化使用 he_normal。第三道捲積層(Convolution Layer)使用 64 個捲積核(kernel)，每個核(kernel)形狀(Shape)為3 × 3，激勵函式(Activation function)為線性整流單元(Rectified Linear Unit ,Relu)，權重初始化使用 he_normal。

表 10 捲積層的超參數

	Filters	**Kernel Size**	**Activation**	**Kernel Initializer**
Conv2D	64	2 x 2	Relu	he_normal
Conv2D	64	2 x 2	Relu	he_normal
Conv2D	64	3 x 3	Relu	he_normal

最後，三道全連接層(Dense Layer)表 11 所示，第一道全連接層(Dense Layer)使用 300 個單元(Units)，激勵函式(Activation function)為線性整流單元(Rectified Linear Unit ,Relu)，權重初始化使用 he_normal。第二道全連接層(Dense Layer)使用 100 個單元(Units)，激勵函式(Activation function)為線性整流單元(Rectified Linear Unit ,Relu)，權重初始化使用 he_normal。第三道全連接層(Dense Layer)使用 3 個單元(Units)，激勵函式(Activation function)為線性(Linear)，權重初始化使用 he_normal。

表 11 全連接層的參數

	Units	**Activation**	**Kernel Initializer**
Dense	300	Relu	he_normal
Dense	100	Relu	he_normal
Dense	3	Linear	he_normal

訓練該網路使用的優化器(Optimizer)為 RMSprop，損失函數(Loss function)為均方誤差(mean-square error, MSE)，如表 12 所示。

表 12 模型訓練的超參數

	Value
Optimizer	RMSprop
Loss	MSE

回顧 3.1 節，一年中約 250 個交易日，因為每 5 天交易 1 次，大概有 50 個狀態被訓練到，但狀態總變化數趨近無限大。當狀態總變化數很多而訓練資料很少的時候，一般會有過擬合(Overfitting)的效應存在。針對 Overfitting 的效應，通常可以在神經網路加入丟失層(Dropout Layer)做處理。經由實驗結果顯示捨棄 Dropout Layer 較能有效的訓練，並且考量到神經網路的職責在於取代 Q-Table，具備足夠的訓練能力即可，因此本論文決定不加入 Dropout Layer。

第四章 實驗與結果

本章將有眾多實驗，首先交代環境建置所需相關訊息，接著依序決定表現較優的參數，並與各對照組比較績效，最後交代實驗結果總結。

4.1. 環境建置

此節將介紹本論文的實作環境，如表 13 所示，處理器(Central Processing Unit, CPU)為 i7-7700、記憶體(Random Access Memory, RAM)為 16G、顯示卡(Graphics car)為 RTX 2070、作業系統(Operating System, OS)為 Windows 10 Enterprise、顯示卡驅動(Graphics Driver)版本為 419.35、CUDA 版本為 10.0、cudnn 版本為 7.5.0、Python 版本為 3.6.8、Tensorflow 版本為 1.13.1、Keras 版本為 2.2.4。

表 13 系統環境

	Value
CPU	i7-7700
Memory	16G
GPU	RTX 2070
OS	Windows 10 Enterprise
GeForce Driver	419.35
CUDA	10.0
cudnn	7.5.0
Python	3.6.8
Tensorflow	1.13.1
Keras	2.2.4

4.2. 實驗導覽

本論文整理的實驗如表 14 所示，簡單分為四項重點：

- 神經網路的調整，如丟失層(Dropout Layer)
- 探討參數的成效，如獎勵放大倍率(Reward Factor)、經驗池大小(Memory Size)

- 比較代理人(Agent)與各對照組的績效
- 以相同的參數做 Chen 的實驗 [1]

表 14 實驗總表

ID	Intent
Dropout Rate-0.5	探討參數 Dropout Rate 0.5 的影響
Dropout Rate-0.7	探討參數 Dropout Rate 0.7 的影響
Dropout Rate-0.9	探討參數 Dropout Rate 0.9 的影響
None Dropout	探討不加入 Dropout Layer 的影響
Reward Factor-10	探討參數 Reward Factor 10 倍的影響
Reward Factor-100	探討參數 Reward Factor 100 倍的影響
Reward Factor-1000	探討參數 Reward Factor 1000 倍的影響
Memory Size-10k	探討參數 Memory Size 10k 的影響
Memory Size-100k	探討參數 Memory Size 100k 的影響
Memory Size-1000k	探討參數 Memory Size 1000k 的影響
Performance-Sideways	與買進持有(Buy and Hold)在橫盤中比較績效
Performance-Downtrend	與買進持有(Buy and Hold)在跌勢中比較績效
Performance-Uptrend	與買進持有(Buy and Hold)在漲勢中比較績效
Performance-Specialized	與買進持有(Buy and Hold)比較訓練前 4 年測試第 5 年的績效

4.3. 年度趨勢統計

為了瞭解代理人在各趨勢的適用性，如跌勢、橫盤、漲勢。此節主要交代如何挑選上述的 3 個趨勢年度，以進行後續實驗。首先，計算各 ETF 的年度成長率，為該年最後 1 個交易日價格對該年第 1 個交易日價格的比值，扣掉 1 倍，再取百分比率，如公式(11)。結果請參閱附錄 A 實驗相關圖表的表 24，接著參考各 ETF 走勢圖，請參閱附錄 A 實驗相關圖表。依據該表及各 ETF 走勢圖，簡略說明如何挑選上述評估 3 個趨勢的年度：

- 評估橫盤：

  本論文選定平均數據較接近零的編號 Year1 評估橫盤，也就是第 2 個年度
- 評估跌勢：

  本論文選定平均平均數據較低的編號 Year3 評估跌勢，也就是第 4 個年度
- 評估漲勢：

  本論文選定平均數據較高的編號 Year5 評估漲勢，也就是第 6 個年度。各 ETF 趨勢圖請參閱附錄 A 實驗相關圖表。

$$growth\ rate = \left(\frac{price_{final}}{price_{initial}} - 1\right) \times 100\% \quad (11)$$

4.4. 丟失層對於代理人的影響

此實驗將測試 Dropout Layer 對於各參數所造成的影響，本實驗將測試 3 組參數，分別為 0.5、0.7、0.9，以及不加入 Dropout Layer 的結果。每一組參數皆實驗 3 次，分別代表不同的隨機種子 0、1、2，以下皆由平均數據做呈現。

實驗結果如表 15 所示，可以發現各 Dropout Rate 的參數皆難以訓練成功。反之，不加入 Dropout Layer 則能有效訓練。因此本論文以不加入 Dropout Layer 作為最終版本，以利後續之實驗。

表 15 丟失層參數比較表

	Result
Dropout Rate 0.5	Fail
Dropout Rate 0.7	Fail
Dropout Rate 0.9	Fail
None Dropout	Succeed

4.5. 獎勵放大倍率對於代理人的影響

此實驗將測試獎勵放大倍率(Reward Factor)對於各參數所造成的影響，本實驗將測試 3 組參數，分別為 10 倍、100 倍、1000 倍的結果。每一組參數皆實驗 3 次，分別代表不同的隨機種子 0、1、2。每次實驗皆有 3 個階段，分別為訓練(Train)階段、驗證(Validate) 階段、以及測試(Test) 階段。個別實驗細項請參閱請參閱附錄 A 實驗相關圖表，以下皆由平均數據做呈現。

實驗結果如表 16 所示，第 2 欄(Column)及第 3 欄，也就是驗證(Validate)及測試(Test)，分別是-4.19%及-4.12%，可以發現獎勵放大倍率(Reward Factor) 為 100 的數據較優於其他兩個參數測試得出的結果，本論文以此參數作為獎勵放大倍率(Reward Factor)最終版本，以利後續之實驗。

表 16 獎勵放大倍率比較表

	Train	Validate	Test
Reward Factor 10	27.62%	-7.58%	-5.14%
Reward Factor 100	44.40%	-4.19%	-4.12%
Reward Factor 1000	49.09%	-7.61%	-6.81%

4.6. 經驗池大小對於代理人的影響

此實驗將測試經驗池大小(Memory Size)對於各參數所造成的影響，本實驗將測試 3 組參數，分別為 10k、100k、1000k 的結果。每一組參數皆實驗 3 次，分別代表不同的隨機種子 0、1、2。每次實驗皆有 3 個階段，分別為訓練(Train)階段、驗證(Validate) 階段、以及測試(Test) 階段。個別實驗細項請參閱請參閱附錄 A 實驗相關圖表，以下皆由平均數據做呈現。個別實驗細項請參閱請參閱附錄 A 實驗相關圖表，以下皆由平均數據做呈現。

實驗結果如表 17 所示，第 2 欄(Column)及第 3 欄，也就是驗證(Validate)及測試(Test)，分別是-4.19%及-4.12%，可以發現經驗池大小(Memory Size)為 100k 的數據較優於其他兩個參數測試得出的結果，本論文以此參數作為經驗池大小(Memory Size)最終版本，以利後續之實驗。

表 17 經驗池大小比較表

	Train	Validate	Test
Memory Size 10k	38.07%	-6.37%	-3.68%
Memory Size 100k	44.40%	-4.19%	-4.12%
Memory Size 1000k	39.89%	-6.98%	-6.12%

4.7. 代理人於各趨勢的績效表現

此節將比較代理人(Agent)與買進持有(Buy and Hold)策略的績效，分別考慮橫盤、跌勢、漲勢等趨勢。個實驗皆執行 3 次，分別使用不同的隨機種子 0、1、2。每次實驗皆有 3 個階段，分別為訓練(Train)階段、驗證(Validate) 階段、以及測試(Test) 階段。最後與買進持有(Buy and Hold)策略及隨機(Random)策略比較績效。實驗細項請參閱請參閱附錄 A 實驗相關圖表，以下皆由平均數據做呈現。

4.7.1. 比較於橫盤趨勢中的績效

此實驗將測試代理人(Agent)於橫盤趨勢的績效表現，首先回顧至 0 中 4.3 節，根據表 24 以第 2 年作為評估橫盤趨勢的年度，即該表中平均數據較接近 0 的那一年作為驗證(Validate)及測試(Test)，由剩餘的 6 年中挑選 4 年作訓練(Train)。

實驗結果如表 18 所示，第 2 欄(Column)及第 3 欄，也就是驗證(Validate)及測試(Test)，可以發現代理人(Agent)的數據與買進持有策略(Buy and Hold)的數據差異不大，也就是說該代理人於橫盤趨勢中的績效與買進持有策略(Buy and Hold)不分勝負，後續將以此結果與其他趨勢的影響作比較。

表 18 橫盤績效比較

	Train	Validate	Test
Buy and Hold	2.45%	4.68%	5.84%
Random	1.08%	2.64%	2.76%
Agent for Sideways	37.71%	5.33%	5.72%

4.7.2. 比較於跌勢中的績效

此實驗將測試代理人(Agent)於跌勢的績效表現，首先回顧至 0 中 4.3 節，根據表 24 以第 4 年作為評估跌勢的年度，即該表中平均數據較低的那一年作為驗證(Validate)及測試(Test)，由剩餘的 6 年中挑選 4 年作訓練(Train)。

實驗結果如表 19 所示，第 2 欄(Column)及第 3 欄，也就是驗證(Validate)及測試(Test)，可以發現代理人(Agent)的數據與買進持有策略(Buy and Hold)的數據有明顯差距，避免虧損的幅度較優，也就是說該代理人於跌趨勢中的績效優於買進持有策略(Buy and Hold)，後續將以此結果與其他趨勢的影響作比較。

表 19 跌勢績效比較

	Train	Validate	Test
Buy and Hold	4.64%	-11.84%	-12.16%
Random	1.83%	-6.92%	-6.56%
Agent for Downtrend	44.40%	-4.19%	-4.12%

4.7.3. 比較於漲勢中的績效

此實驗將測試代理人(Agent)於漲勢的績效表現，首先回顧至 0 中 4.3 節，根據表 24 以第 6 年作為評估跌勢的年度，即該表中平均數據較低的那一年作為驗證(Validate)及測試(Test)，由剩餘的 6 年中挑選 4 年作訓練(Train)。

實驗結果如表 20 所示，第 2 欄(Column)及第 3 欄，也就是驗證(Validate)及測試(Test)，可以發現代理人(Agent)的數據與買進持有策略(Buy and Hold)的數據有明顯差距，獲利能力尚有發展空間，也就是說該代理人於漲趨勢中的績效不及買進持有策略(Buy and Hold)，後續將以此結果與其他趨勢的影響作比較。

表 20 漲勢績效比較

	Train	Validate	Test
Buy and Hold	2.66%	12.05%	13.32%
Random	1.52%	4.98%	5.86%
Agent for Uptrend	32.77%	3.61%	7.79%

4.8. 比較 Agent 訓練前 4 年測試第 5 年的績效

此實驗將測試代理人(Agent)於訓練前 4 年測試第 5 年績效表現，目的為嘗試重現 Chen 的績效 [1]，即以第 5 年作為驗證(Validate)及測試(Test)，由前 4 年作訓練(Train)。本實驗將執行 3 次，分別使用不同的隨機種子 0、1、2。每次實驗皆有 3 個階段，分別為訓練(Train)階段、驗證(Validate) 階段、以及測試(Test) 階段。最後與買進持有(Buy and Hold)策略及隨機(Random)策略比較績效。個別實驗細項請參閱請參閱附錄 A 實驗相關圖表，以下皆由平均數據做呈現。

實驗結果如表 21 所示，第 2 欄(Column)及第 3 欄，也就是驗證(Validate)及測試(Test)。可以發現代理人(Agent)的數據相較於與買進持有策略(Buy and Hold)的數據，儘管獲利能力不錯，仍無超越買進持有策略(Buy and Hold)，也就是說要100%復現(Reproduce) Chen 的實驗績效有一定難度。

表 21 訓練前 4 年測試第 5 年績效比較

	Train	Validate	Test
Buy and Hold	13.05%	22.11%	6.95%
Random	4.76%	10.04%	3.03%
Specialized Agent	348.25%	8.61%	5.19%

4.9. 結果與討論

本節依據上述的實驗，分為三部分作統整：參數選擇、趨勢的影響、以相同的參數做 Chen 的實驗 [1]。實驗過程中，本論文亦嘗試各種方法，結果皆不甚理想，以下以 5 個面向簡略陳述本論文嘗試的各種方法：

- 狀態：

  在狀態方面所做的嘗試：本論文試著調整每次參考過去歷史的區間{5, 10, 20,40}日，或者更換正規化函數，以及測試加入交易訊號線、現金、持股與否等影響。結果表示加入現金及持股對於描述唯一狀態是有幫助的，而參考過長的歷史效益不大，反而有過多雜訊不利於判斷。代理人對於各種趨勢的泛化能力(generalizability)具有較多發展空間，可以朝此方面加以優化。

- 動作：

  在動作方面所做的嘗試：本論文試著調整每次交易時間的間隔{5, 10, 20}日，或是各可執行的動作數{3, 5, 7}個，各動作交易單位大小{1, 2, 5, 10, 20}個單位，交易價位以 ETF 原價位或正規化後的價位，以及測試忽略無效動作(Invalid Action)與否等影響。結果表示套用 Valid Action 的概念幫助最顯著，一方面不需針對懲罰性獎勵調參數，二方面不用處理違法動作的後續。由於市場有漲有跌，多空雙做是值得嘗試的。

- 獎勵：

  在獎勵函數方面所做的嘗試：本論文試著調整各獎勵函數，如以常數為主的獎勵，或是對數為主的獎勵，獎勵放大或平移，以及測試針對無效動作(Invalid Action)該給的懲罰大小。結果表示常數為主的獎勵不適合，較無法區別獲利或虧損的幅度。適當的放大獎勵則有助於學習的過程，過大則不合適，也許 Clip Rewards 是值得嘗試的。

- 神經網路架構：

  在神經網路方面所做的嘗試，本論文試著調整構神經網路的架構，如調整捲積層個數，加入池化層(Pooling)等，或是全連接層中單元個數(Units)，優化器(Optimizer)調配{SGD、RMSProp、ADAM}，以及測試激勵函數(Activation Function)選擇{Relu, tanh, linear}的效果等。結果表示優化器 SGD 相較於其他兩個，訓練過程較坎坷，容易產生 NAN 的現象。對於此回歸問題而言，激勵函數 Relu 不適合套用於最後一層。精進神經網路效益不顯著，具備穩健的訓練能力即可，至於有沒有獲利能力並非它的職責。

- 強化學習方法：

  在強化學習演算法方面所做的嘗試：本論文試著更換演算法，如 DQN 及 DDQN、測試加入經驗回放池與否等影響。結果表示使用經驗回放池確實對於學習的過程有顯著的效益，DDQN 則較 DQN 有助於平均數據的穩定性。此外，在 RL 家族中仍有其他演算法值得做嘗試。

由 4.4 的各項實驗中，觀察其結果可以得知，不加入 Dropout Layer，相對於{0.5, 0.7, 0.9}等 Dropout Rate，能較有效的訓練。由 4.5 的各項實驗中，觀察其結果可以得知，獎勵放大倍率(Reward Factor)取 100 這個數值，為{10, 100, 1000}中對於 ROI 的表現較有利的選擇。由 4.6 的各項實驗中，觀察其結果可以得知，經驗池大小(Memory Size)取100k這個數值，為{10k, 100k, 1000k}中對於 ROI 的表現較有利的選擇。

由 4.7 的各實驗中，觀察其結果可以得知，該代理人於橫盤趨勢中與買進持有(Buy and Hold)策略不分勝負。於跌勢中，相較於買進持有(Buy and Hold)策略，有能力避免大幅虧損。於漲勢中，雖不及買進持有(Buy and Hold)策略，仍有獲利能力。可能的因素為：RL 演算法是往最大化報酬的方向前進，也就是說會盡量避免虧損的方向，若處於很強的漲勢則未必會賺的比 Buy and Hold 多。最後，參考 4.8 的實驗結果，儘管該代理人(Agent)有不錯的績效表現，做了各種參數及方法上的嘗試後，要復現(Reproduce) Chen 於 [1]的實驗績效仍有難度。

第五章 結論與未來研究方向

5.1. 結論

本論文以 DDQN 的方法訓練 ETF 之交易代理人，嘗試各種參數優化後，依據三個面向分別探討該代理人的績效：橫盤、跌勢、漲勢。根據實驗結果顯示，該代理人於各趨勢中的獲利表現不盡相同，表明即使有獲利能力，對於泛化(Generalization)的能力仍有發展空間。

因應本論文以強化學習家族的演算法訓練交易代理人，並參閱各文獻之方法，以下將從 6 個面向，簡略敘述本論文之貢獻：

- 狀態方面：

  針對狀態方面，本論文除了價位資訊，仍給予交易訊號及現金及持股等訊息，以更精確描述唯一狀態。
- 動作方面：

  針對動作方面，本論文除了有效動作的程序之外，亦考量無效動作的情境，並加以處裡。
- 獎勵方面：

  針對獎勵方面，本論文在對數函數為主的獎勵上，加入放大倍率等因素，並加以實驗，得到較優之參數。
- 復現方面：

  針對復現方面，本論文以實驗數據佐證，要復現實驗的績效數據，仍有一定難度，發展空間前途無量。
- 因素方面：

  針對因素方面，本論文以各種可能的方向，嘗試多種組合並加以實驗，結果不甚理想。
- 獲利方面：

  針對獲利方面，本論文以實驗數據佐證，代理人於各趨勢的獲利表現不盡相同，顯示對於泛化能力仍有進步空間。

5.2. 未來研究方向

本節將討論未來可能發展方向，以下簡單依據交易策略、演算法、過擬合、衡量績效等 4 個方向，於表 22 中敘述之。

表 22 未來發展方向

	Description
交易策略	在交易市場中，交易動作其實不只能做多(Long)，做空(Short)也是一個可以考慮的選項。於本論文中僅以做多為基礎探討，未來發展若採多空方向雙作的方向，將有更廣的獲利空間。
演算法	強化學習家族中，還有多種演算法，在本論文中以 value based 為主，選用 Q-Learning 家族的 DDQN 實作並探討。未來發展若嘗試 policy based 的 Policy Gradient 家族，也是一個不錯的選擇。
過擬合	在 training 過程中時常有 overfitting 的現象，未來發展值得嘗試套用技術分析，定義幾個 flag：如 W 底、M 頭等，簡化成幾種型態，以縮小總狀態數。而套用諸如此類的技術分析，需要一點 effort 定義一個 window 作為往前看的天數。
衡量績效	績效衡量方面有多種工具，於本論文中以 ROI 為基礎做最佳化。考量 CP 值及獲利穩定度，未來發展值得嘗試 Sharpe Ratio 及 Profit Factor。

參考文獻

[1] F.-T. Chen, "Convolutional Deep Q-learning for ETF Automated Trading System," *國立政治大學應用數學系碩士學位論文*, 2017.

[2] "MoneyDJ 理財網," [Online]. Available: https://www.moneydj.com/. [Accessed 16 6 2019].

[3] "Yahoo Finance," [Online]. Available: https://finance.yahoo.com/. [Accessed 16 6 2019].

[4] Richard S. Sutton and Andrew G. Barto, Reinforcement Learning : An Introduction, 1998.

[5] Chris Jch Watkins and Peter Dayan, "Q-Learning," *Machine Learning*, pp. 279-292.

[6] Mnih, Volodymyr; Kavukcuoglu, Koray; Silver, David; Rusu, Andrei A.; Veness, Joel; Bellemare, Marc G.; Graves, Alex; Riedmiller, Martin; Fidjeland, Andreas K.; Ostrovski, Georg; Petersen, Stig; Beattie, Charles; Sadik, Amir; Antonoglou, Ioannis; King, Helen; Kumaran, Dharshan; Wierstra, Daan; Legg, Shane; Hassabis, Demis, "Human-level control through deep reinforcement learning," *Nature*, 2015.

[7] Hado van Hasselt, Arthur Guez and David Silver, "Deep Reinforcement Learning with Double Q-learning," in *AAAI*, 2016.

[8] 劉上瑋, "The Application of Deep Reinforcement Learning on Dynamic Asset Allocation : A Case Study of U.S. ETFs," *國立政治大學金融學系研究所碩士學位論文*, 2017.

[9] X. Gao, Deep reinforcement learning for time series: playing idealized trading games, 2018.

附錄 A 實驗相關圖表

1. ETF 列表及各年度成長率

表 23 ETF 列表

ETF	投資標的	追蹤指數	ETF 名稱
ADRA	股票型	The BNY Mellon Asia 50 ADR Index	Invesco BLDRS 亞洲 50 ADR 指數 ETF
BIV	中期債券	Barclays U.S. 5 – 10 Year Government/Credit Float Adjusted Index	Vanguard 美國中期債券 ETF
CORP	公司債券	BofA ML US Corporate Index	PIMCO 投資級公司債券指數 ETF
DBO	能源	DBIQ Optimum Yield Crude Oil Index Excess Return	Invesco 德銀石油 ETF
DEF	股票型	Sabrient Defensive Equity Index	Invesco 防禦型股票 ETF
DFJ	股票型	WisdomTree Japan SmallCap Dividend index	WisdomTree 日本高股利小型股 ETF
DGP	做多型	Deutsche Bank Liquid Commodity index - Optimum Yield Gold Excess Return	德銀二倍做多黃金 ETN
DIA	股票型	Dow Jones Industrial Average	SPDR Dow Jones Industrial Average ETF
DLS	股票型	WisdomTree International SmallCap Dividend index	WisdomTree 國際高股利小型股 ETF
DON	股票型	WisdomTree MidCap Dividend index	WisdomTree 美國高股利中型股 ETF
DTD	股票型	WisdomTree Dividend Index	WisdomTree 美國總體高股利 ETF
DTUL	做多型	Barclays 2Y US Treasury Futures Targeted Exposure Index	iPath 做多二年期美國公債期貨 ETN
DVY	股票型	Dow Jones U.S. Select Dividend Index	iShares 精選高股利指數 ETF
DWX	股票型	S&P International Dividend Opportunities index	SPDR 標普國際高股利 ETF
EDV	長期債券	Barclays U.S. Treasury Strips 20-30 Year Equal Par Bond Index	Vanguard 美國延期公債 ETF
EFAV	股票型	MSCI EAFE Minimum Volatility (USD) Index	iShares Edge MSCI 歐澳遠東最小波動率 ETF
EMIF	公共事業股	S&P Emerging Markets Infrastructure Index	iShares 新興市場基礎建設 ETF
EWV	反向型	MSCI Japan Index	ProShares 二倍放空 MSCI 日本 ETF
FAB	股票型	NASDAQ AlphaDEX Multi Cap Value Index	First Trust 全市場價值型 AlphaDEX 指數 ETF
FEX	股票型	NASDAQ AlphaDEX Large Cap Core Index	First Trust 大型核心股指數 ETF
FWDI	股票型	MSCI EAFE Index	Madrona 國際指數基金
FXA	匯率型	WM/Reuters Australian Dollar Closing Spot Rate	Invesco CurrencyShares 澳幣 ETF
FXC	匯率型	WM/Reuters Canadian Dollar Closing Spot Rate	Invesco CurrencyShares 加拿大幣 ETF
FXG	必需性消費股	StrataQuant Consumer Staples Index	First Trust 必需性消費 AlphaDEX 指數 ETF
GII	公共事業股	S&P Global Infrastructure Index	SPDR S&P Global Infrastructure ETF
GMF	股票型	S&P Asia Pacific Emerging BMI Index	SPDR 標普新興亞洲太平洋 ETF
VOX	電信股	MSCI US Investable Market Communication Services 25/50 Transition Index	Vanguard 通訊服務 ETF
VPU	公共事業股	MSCI US Investable Market Utilities 25/50 Index	Vanguard 公用事業類股
VQT	股票型	S&P 500 Dynamic VEQTOR Indx TR	Barclays 標普波動性目標報酬 ETN
VSS	股票型	FTSE Global Small Cap ex US Index	Vanguard 美國以外全世界小型股 ETF
VTWV	股票型	Russell 2000 Value Index	Vanguard 羅素 2000 價值指數 ETF
VV	股票型	CRSP US Large Cap Index	Vanguard 大型股 ETF
WEAT	期貨商品	CBOT Wheat Futures PR USD	Teucrium 小麥 ETF
WREI	股票型	Wilshire US Real Estate Investment Trust Index	威爾夏美國房地產投資信託 ETF
XLB	原物料	Materials Select Sector Index	SPDR 原物料類股 ETF
XLG	股票型	S&P Top 50 Index	Invesco 標普五百前 50 大 ETF
XPH	健康護理股	S&P Pharmaceuticals Select Industry Index	SPDR 標普製藥 ETF
XRT	非必需消費股	S&P Retail Select Industry Index	SPDR 標普零售業 ETF

表 24 各年度 ETF 成長率統計表

	ETF	Year0	Year1	Year2	Year3	Year4	Year5	Year6
0	ADRA	8.96%	1.98%	4.49%	-17.99%	21.86%	28.60%	-19.33%
1	BIV	-0.74%	-4.06%	3.01%	-1.09%	-2.12%	-0.60%	-0.71%
2	CORP	1.77%	-5.67%	3.53%	-4.87%	3.33%	1.92%	-3.74%
3	DBO	-7.03%	1.06%	-46.91%	-49.67%	26.91%	14.30%	-10.86%
4	DEF	12.05%	8.16%	15.32%	-10.64%	14.72%	23.03%	-4.06%
5	DFJ	0.39%	13.28%	0.80%	6.16%	24.18%	30.30%	-19.29%
6	DGP	-11.02%	-43.79%	-11.42%	-16.10%	-0.18%	17.48%	0.38%
7	DIA	8.85%	12.78%	11.47%	-7.31%	22.66%	31.22%	-6.02%
8	DLS	14.17%	12.67%	-10.01%	-3.71%	17.69%	26.56%	-19.84%
9	DON	13.43%	16.42%	16.00%	-8.67%	24.24%	12.57%	-3.43%
10	DTD	11.98%	13.34%	13.54%	-7.76%	20.40%	17.52%	-6.69%
11	DTUL	0.79%	2.22%	6.44%	5.08%	-4.38%	-7.29%	-5.66%
12	DVY	10.74%	13.64%	13.90%	-5.53%	19.55%	12.45%	-7.04%
13	DWX	-2.08%	-6.64%	-5.78%	-27.66%	18.44%	12.89%	-13.23%
14	EDV	-4.54%	-12.12%	37.40%	-4.71%	-11.94%	4.48%	-2.56%
15	EFAV	10.22%	6.09%	6.52%	-1.23%	1.07%	18.75%	-8.16%
16	EMIF	9.23%	-14.39%	1.75%	-21.91%	14.16%	17.76%	-12.87%
17	EWV	-19.24%	-34.45%	-12.09%	-8.74%	-35.08%	-38.38%	26.50%
18	FAB	13.86%	19.16%	10.57%	-19.94%	35.09%	14.75%	-9.30%
19	FEX	12.44%	20.10%	14.30%	-11.86%	25.23%	23.38%	-9.05%
20	FWDI	4.53%	10.81%	-7.94%	-14.64%	14.74%	29.53%	-20.31%
21	FXA	-3.65%	-13.53%	-13.66%	-7.80%	7.22%	5.47%	-11.25%
22	FXC	-0.32%	-9.37%	-12.05%	-9.55%	6.24%	5.40%	-7.38%
23	FXG	12.42%	26.39%	25.24%	-1.04%	6.43%	11.46%	-12.01%
24	GII	3.51%	8.55%	7.18%	-13.05%	10.66%	13.63%	-7.70%
25	GMF	3.35%	-7.67%	15.88%	-21.76%	21.38%	40.40%	-19.07%
26	VOX	13.55%	10.22%	8.97%	-3.01%	17.00%	-5.90%	-15.26%
27	VPU	7.56%	6.91%	17.06%	0.39%	5.79%	4.12%	5.36%
28	VQT	0.38%	7.49%	7.11%	-13.35%	4.56%	21.91%	-7.43%
29	VSS	6.59%	8.54%	-6.62%	-10.64%	16.65%	25.45%	-19.48%
30	VTWV	13.56%	16.11%	5.73%	-16.17%	42.05%	9.02%	-7.51%
31	VV	12.36%	18.84%	12.99%	-7.02%	21.69%	23.92%	-6.56%
32	WEAT	-5.51%	-28.05%	-19.55%	-19.96%	-18.64%	-11.81%	-4.75%
33	WREI	12.12%	-1.13%	26.28%	-8.01%	8.25%	-3.33%	5.62%
34	XLB	7.31%	13.75%	9.86%	-17.61%	28.59%	20.98%	-14.87%
35	XLG	11.15%	15.38%	11.41%	-2.93%	19.34%	25.60%	-7.28%
36	XPH	12.88%	51.31%	22.67%	-25.02%	-6.39%	14.70%	-12.10%
37	XRT	16.75%	18.99%	18.24%	-15.92%	9.38%	11.59%	-7.01%
	Average	5.60%	4.56%	5.31%	-11.19%	11.86%	13.26%	-8.00%

2. 獎勵放大倍率 10 統計及分布

表 25 獎勵放大倍率 10 統計表

	Train	Validate	Test
Seed 0	14.13%	-8.74%	-5.79%
Seed 1	35.54%	-7.58%	-2.63%
Seed 2	33.18%	-6.42%	-7.00%
Average	27.62%	-7.58%	-5.14%

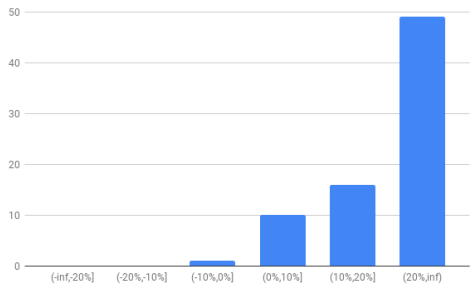


圖 17 獎勵放大倍率 10 的分布-訓練階段

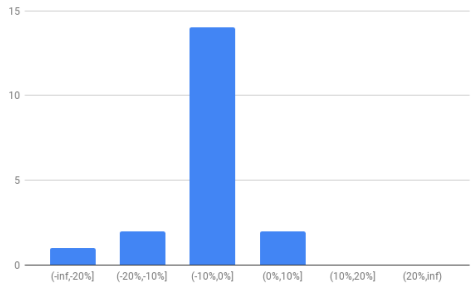


圖 18 獎勵放大倍率 10 的分布-驗證階段

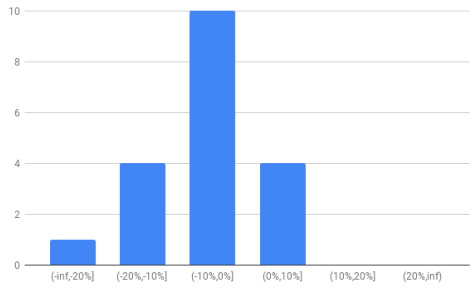


圖 19 獎勵放大倍率 10 的分布-測試階段

3. 獎勵放大倍率 100 統計及分布

表 26 獎勵放大倍率 100 統計表

	Train	Validate	Test
Seed 0	40.47%	-4.05%	-7.16%
Seed 1	47.39%	-5.95%	-1.74%
Seed 2	45.34%	-2.58%	-3.47%
Average	44.40%	-4.19%	-4.12%

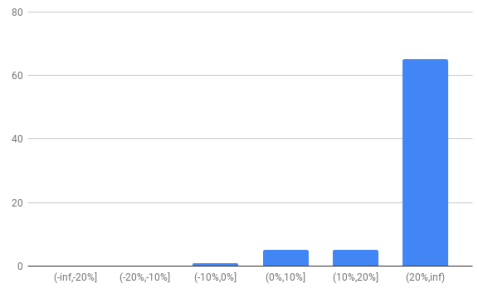


圖 20 獎勵放大倍率 100 的分布-訓練階段

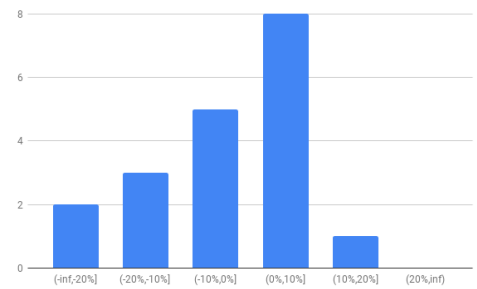


圖 21 獎勵放大倍率 100 的分布-驗證階段

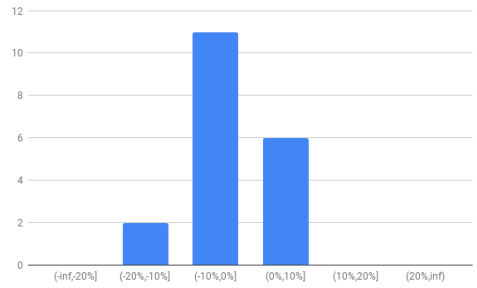


圖 22 獎勵放大倍率 100 的分布-測試階段

4. 獎勵放大倍率 1000 統計及分布

表 27 獎勵放大倍率 1000 統計表

	Train	Validate	Test
Seed 0	49.37%	-8.79%	-3.58%
Seed 1	49.28%	-6.68%	-6.47%
Seed 2	48.63%	-7.37%	-10.37%
Average	49.09%	-7.61%	-6.81%

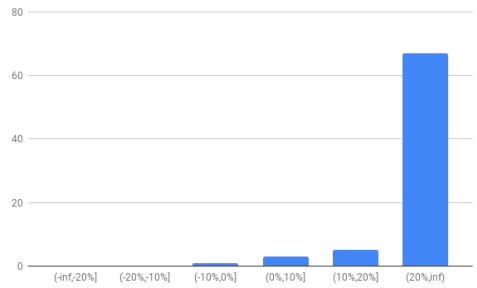


圖 23 獎勵放大倍率 1000 的分布-訓練階段

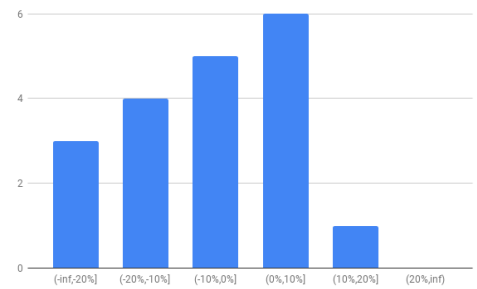


圖 24 獎勵放大倍率 1000 的分布-驗證階段

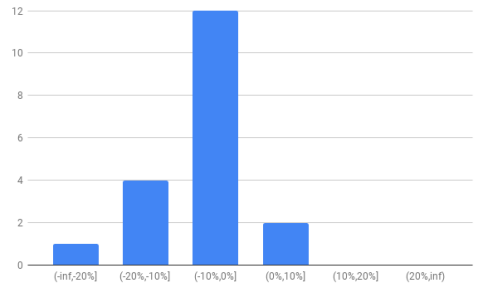


圖 25 獎勵放大倍率 1000 的分布-測試階段

5. 經驗池大小 10k 統計及分布

表 28 經驗池大小 10k 統計表

	Train	Validate	Test
Seed 0	38.97%	-6.21%	-5.42%
Seed 1	31.61%	-4.26%	0.26%
Seed 2	43.62%	-8.63%	-5.89%
Average	38.07%	-6.37%	-3.68%

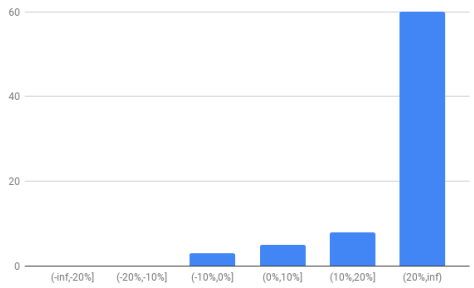


圖 26 經驗池大小 10k 的分布-訓練階段

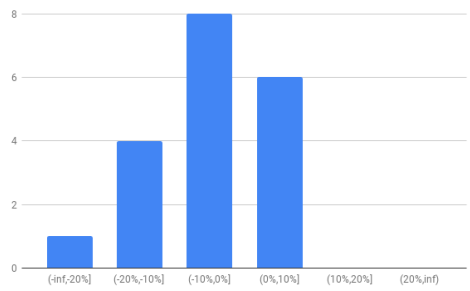


圖 27 經驗池大小 10k 的分布-驗證階段

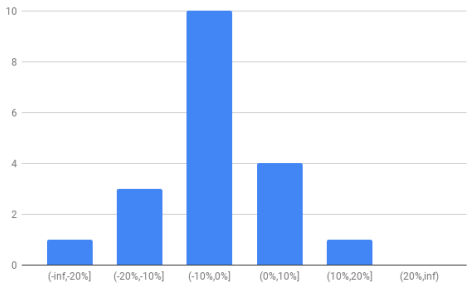


圖 28 經驗池大小 10k 的分布-測試階段

6. 經驗池大小 100k 統計及分布

表 29 經驗池大小 100k 統計表

	Train	Validate	Test
Seed 0	40.47%	-4.05%	-7.16%
Seed 1	47.39%	-5.95%	-1.74%
Seed 2	45.34%	-2.58%	-3.47%
Average	44.40%	-4.19%	-4.12%

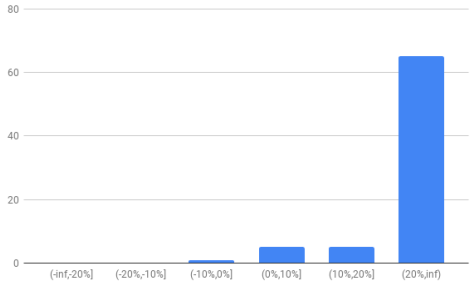


圖 29 經驗池大小 100k 的分布-訓練階段

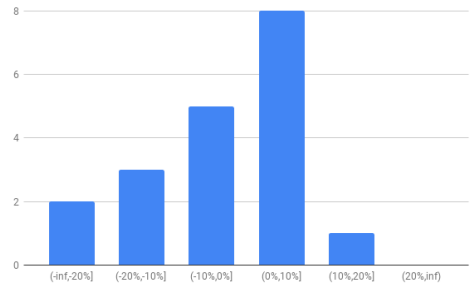


圖 30 經驗池大小 100k 的分布-驗證階段

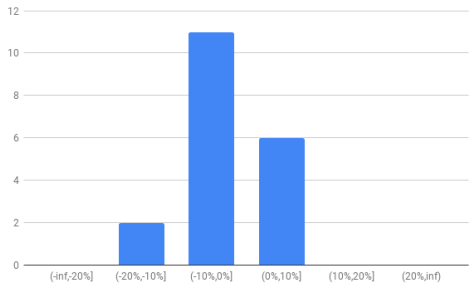


圖 31 經驗池大小 100k 的分布-測試階段

7. 經驗池大小 1000k 統計及分布

表 30 經驗池大小 1000k 統計表

	Train	Validate	Test
Seed 0	28.62%	-6.58%	-5.68%
Seed 1	47.86%	-4.89%	-6.11%
Seed 2	43.21%	-9.47%	-6.58%
Average	39.89%	-6.98%	-6.12%

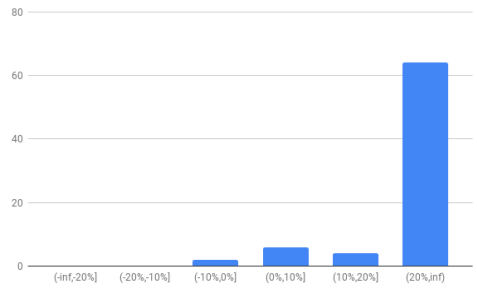


圖 32 經驗池大小 1000k 的分布-訓練階段

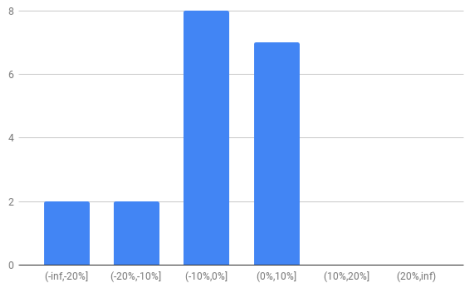


圖 33 經驗池大小 1000k 的分布-驗證階段

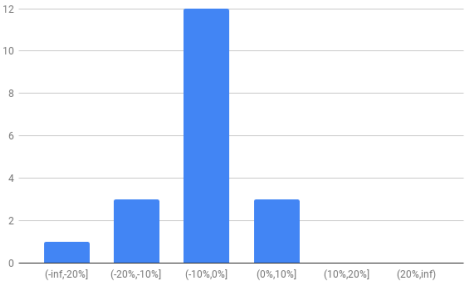


圖 34 經驗池大小 1000k 的分布-測試階段

8. 橫盤績效統計統計及分布

表 31 代理人橫盤績效統計

	Train	Validate	Test
Seed 0	41.04%	5.00%	4.00%
Seed 1	37.14%	5.84%	4.89%
Seed 2	34.93%	5.16%	8.26%
Average	37.71%	5.33%	5.72%

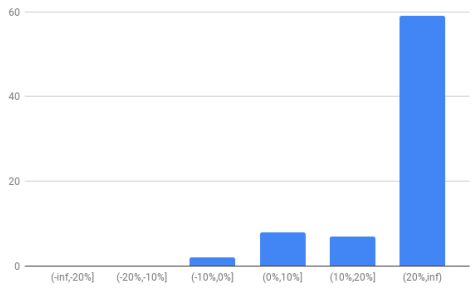


圖 35 代理人橫盤績效的分布-訓練階段

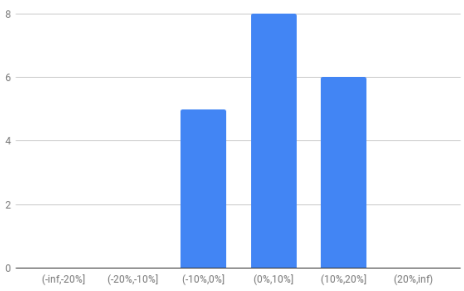


圖 36 代理人橫盤績效的分布-驗證階段

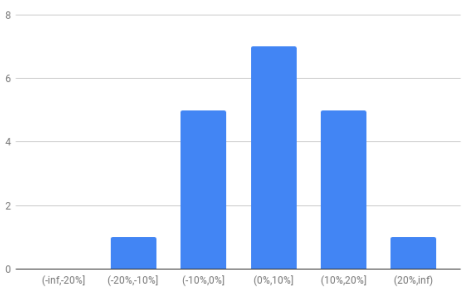


圖 37 代理人橫盤績效的分布-測試階段

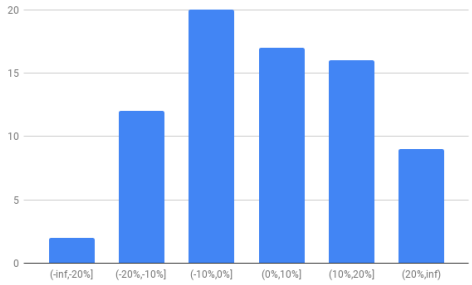


圖 38 Buy and Hold 橫盤績效的分布-訓練階段

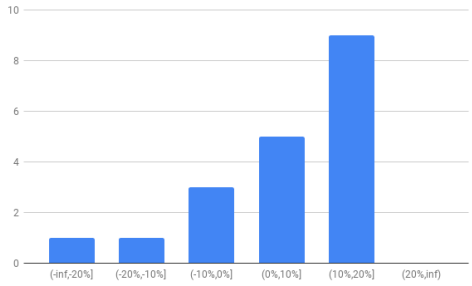


圖 39 Buy and Hold 橫盤績效的分布-驗證階段

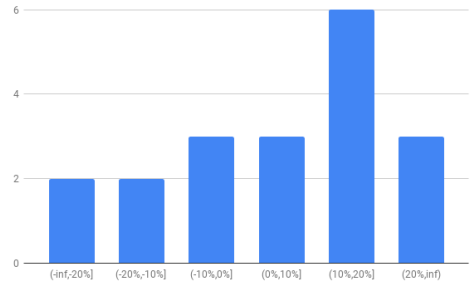


圖 40 Buy and Hold 橫盤績效的分布-測試階段

9. 跌勢績效統計統計及分布

表 32 代理人跌勢績效統計

	Train	Validate	Test
Seed 0	40.47%	-4.05%	-7.16%
Seed 1	47.39%	-5.95%	-1.74%
Seed 2	45.34%	-2.58%	-3.47%
Average	44.40%	-4.19%	-4.12%

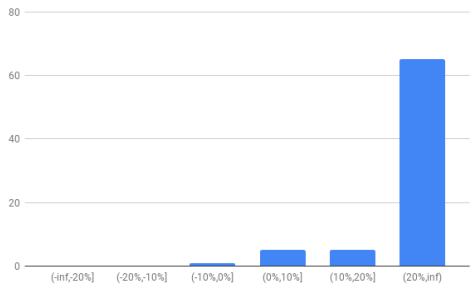


圖 41 代理人跌勢績效的分布-訓練階段

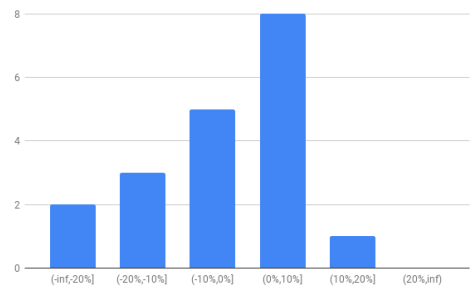


圖 42 代理人跌勢績效的分布-驗證階段

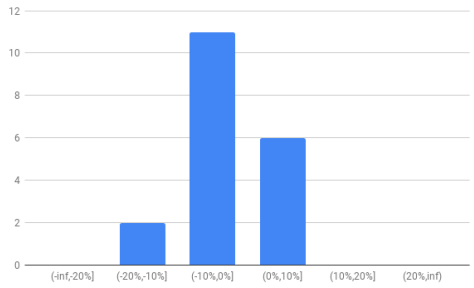


圖 43 代理人跌勢績效的分布-測試階段

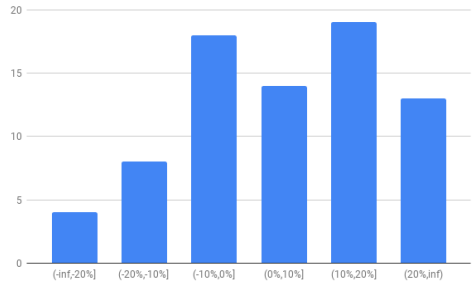


圖 44 Buy and Hold 跌勢績效的分布-訓練階段

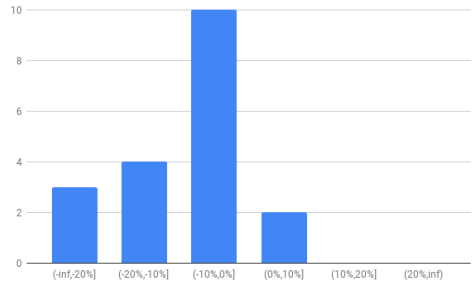


圖 45 Buy and Hold 跌勢績效的分布-驗證階段

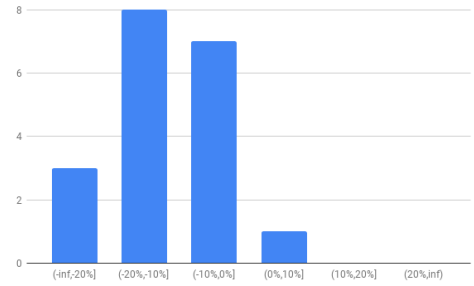


圖 46 Buy and Hold 跌勢績效的分布-測試階段

10. 漲勢績效統計統計及分布

表 33 代理人漲勢績效統計

	Train	Validate	Test
Seed 0	9.49%	6.79%	9.47%
Seed 1	42.74%	3.11%	6.47%
Seed 2	46.09%	0.95%	7.42%
Average	32.77%	3.61%	7.79%

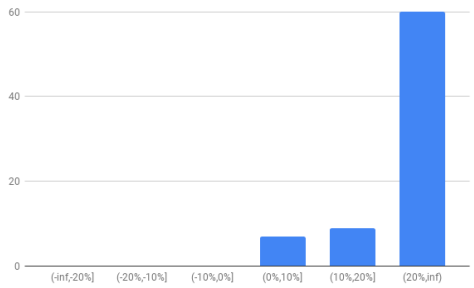


圖 47 代理人漲勢績效的分布-訓練階段

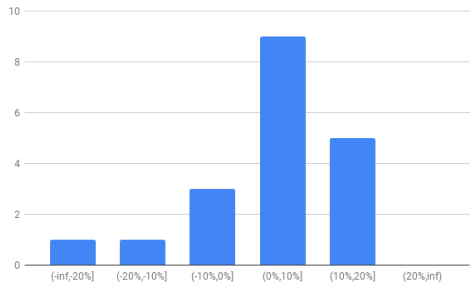


圖 48 代理人漲勢績效的分布-驗證階段

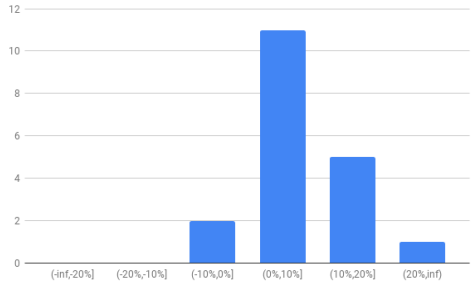


圖 49 代理人漲勢績效的分布-測試階段

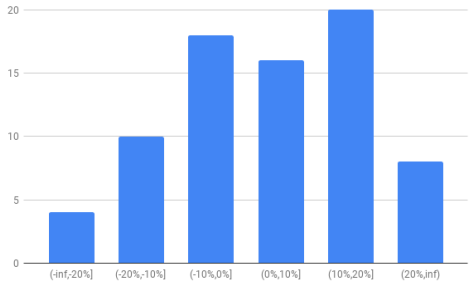


圖 50 Buy and Hold 漲勢績效的分布-訓練階段

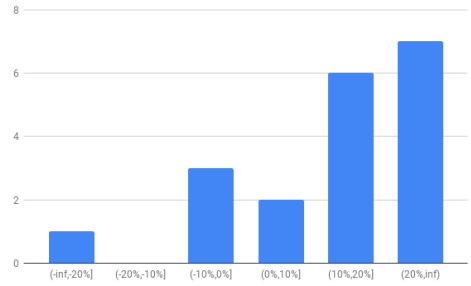


圖 51 Buy and Hold 漲勢績效的分布-驗證階段

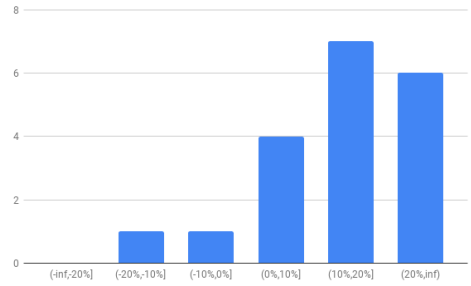


圖 52 Buy and Hold 漲勢績效的分布-測試階段

11. 訓練前 4 年測試第 5 年績效統計及分布

表 34 代理人訓練前 4 年測試第 5 年績效統計

	Train	Validate	Test
Seed 0	480.11%	8.37%	6.32%
Seed 1	95.79%	9.26%	3.84%
Seed 2	468.84%	8.21%	5.42%
Average	348.25%	8.61%	5.19%

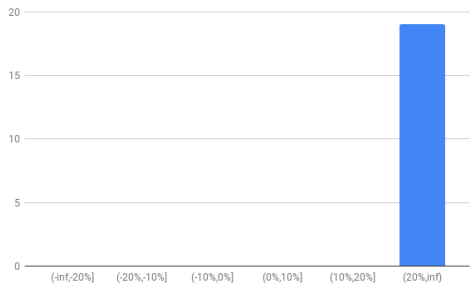


圖 53 代理人訓練前 4 年測試第 5 年績效的分布-訓練階段

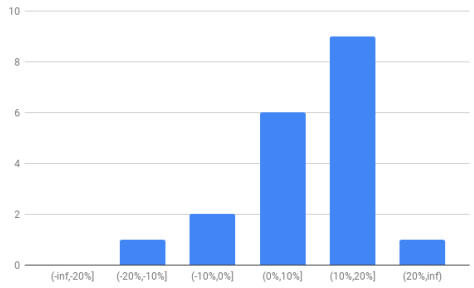


圖 54 代理人訓練前 4 年測試第 5 年績效的分布-驗證階段

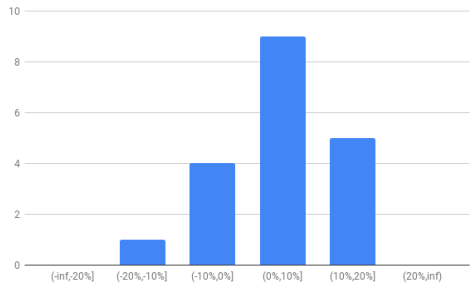


圖 55 代理人訓練前 4 年測試第 5 年績效的分布-測試階段

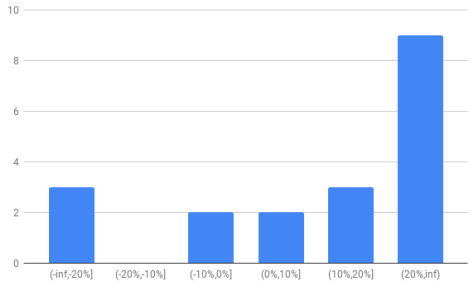


圖 56 Buy and Hold 訓練前 4 年測試第 5 年績效的分布-訓練階段

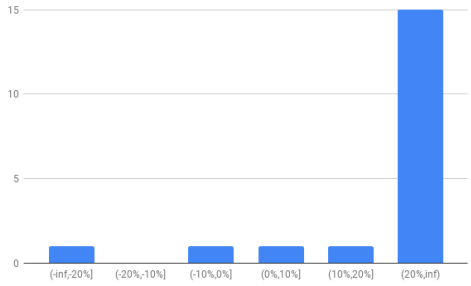


圖 57 Buy and Hold 訓練前 4 年測試第 5 年績效的分布-驗證階段

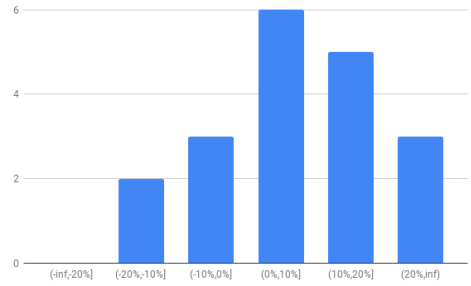


圖 58 Buy and Hold 訓練前 4 年測試第 5 年績效的分布-測試階段

12. ETF 走勢圖

圖 59 ADRA 走勢圖(收盤價)

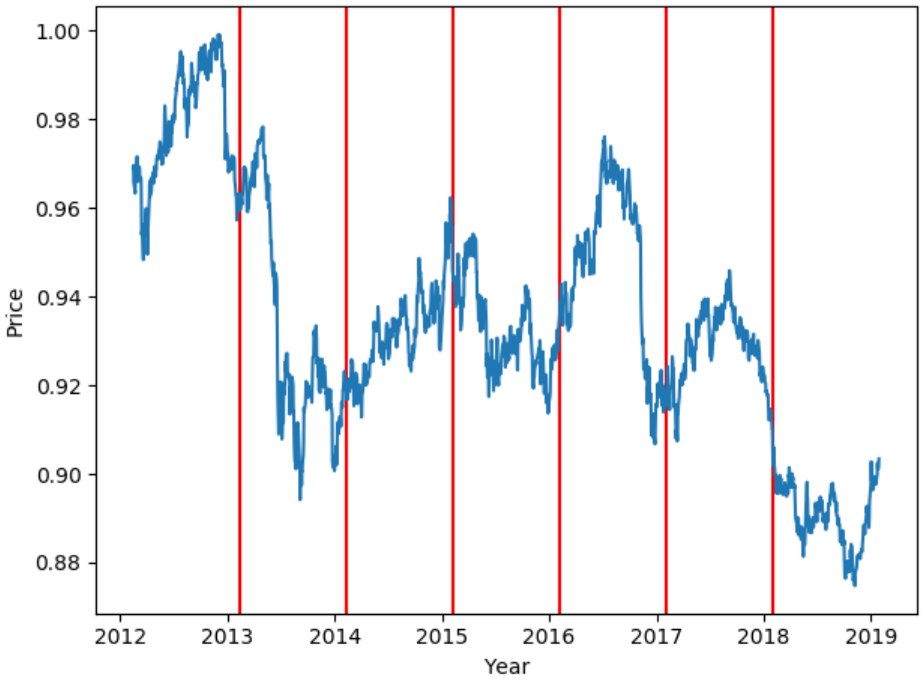


圖 60 BIV 走勢圖(收盤價)

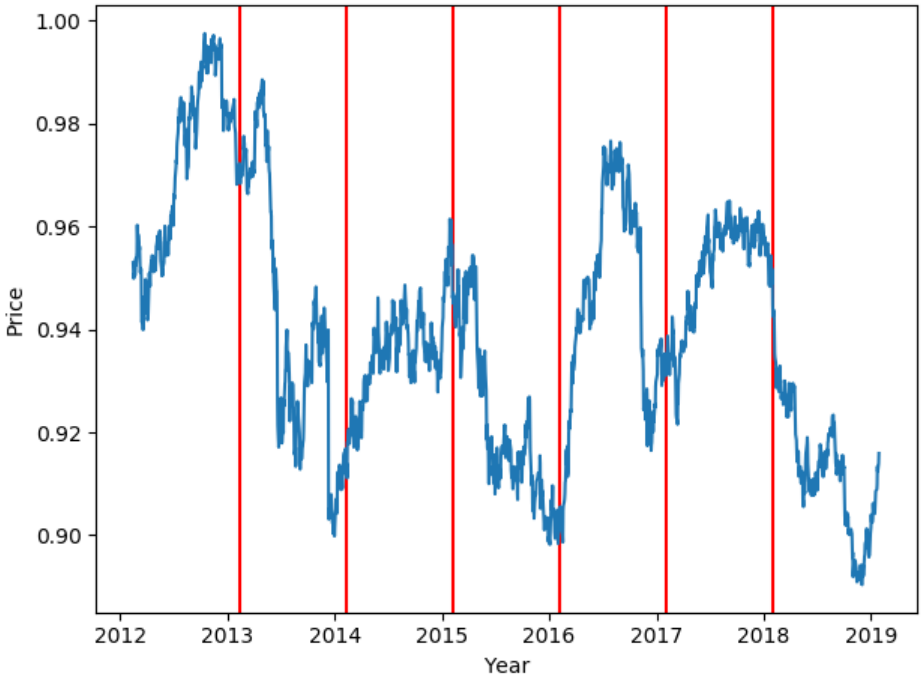


圖 61 CORP 走勢圖(收盤價)

圖 62 DBO 走勢圖(收盤價)

圖 63 DEF 走勢圖(收盤價)

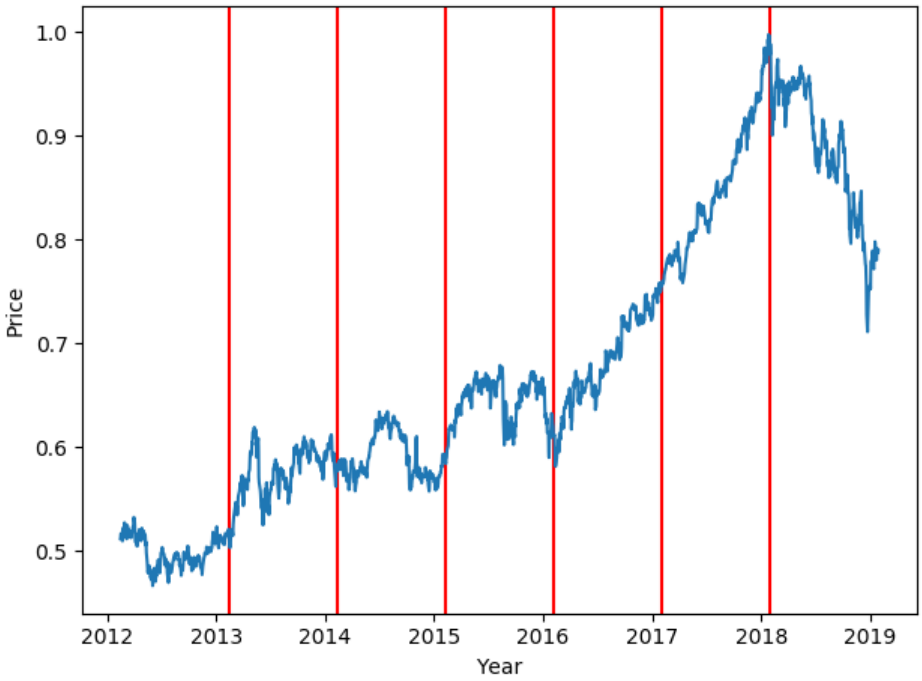


圖 64 DFJ 走勢圖(收盤價)

圖 65 DGP 走勢圖(收盤價)

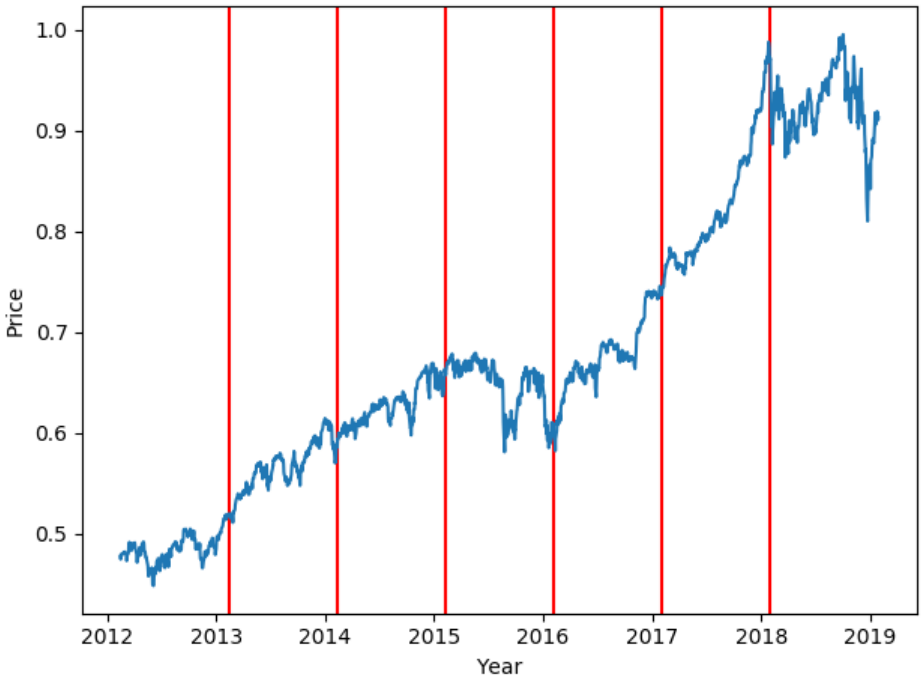


圖 66 DIA 走勢圖(收盤價)

圖 67 DLS 走勢圖(收盤價)

圖 68 DON 走勢圖(收盤價)

圖 69 DTD 走勢圖(收盤價)

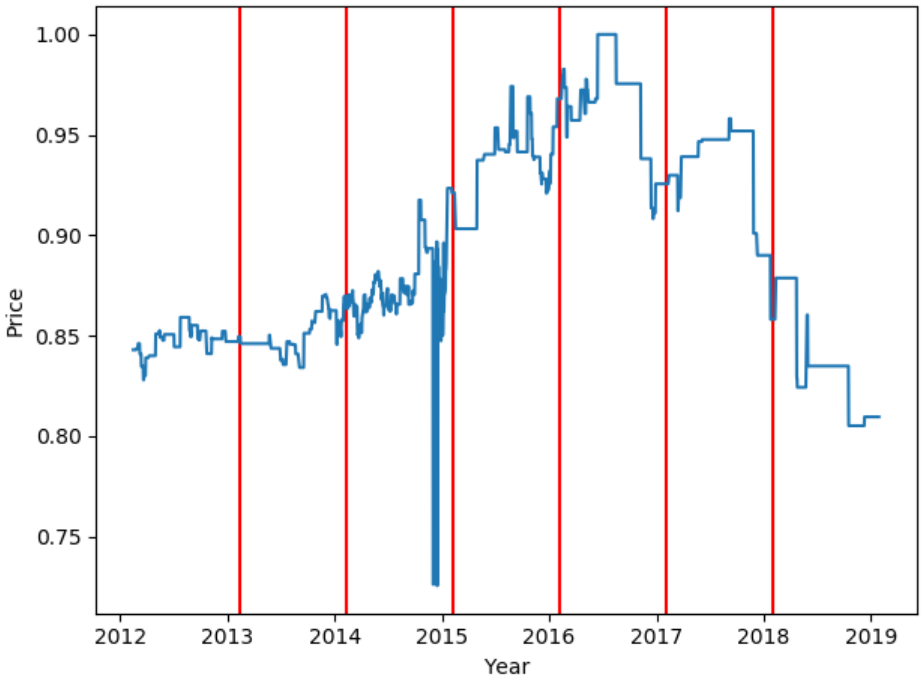


圖 70 DTUL 走勢圖(收盤價)

圖 71 DVY 走勢圖(收盤價)

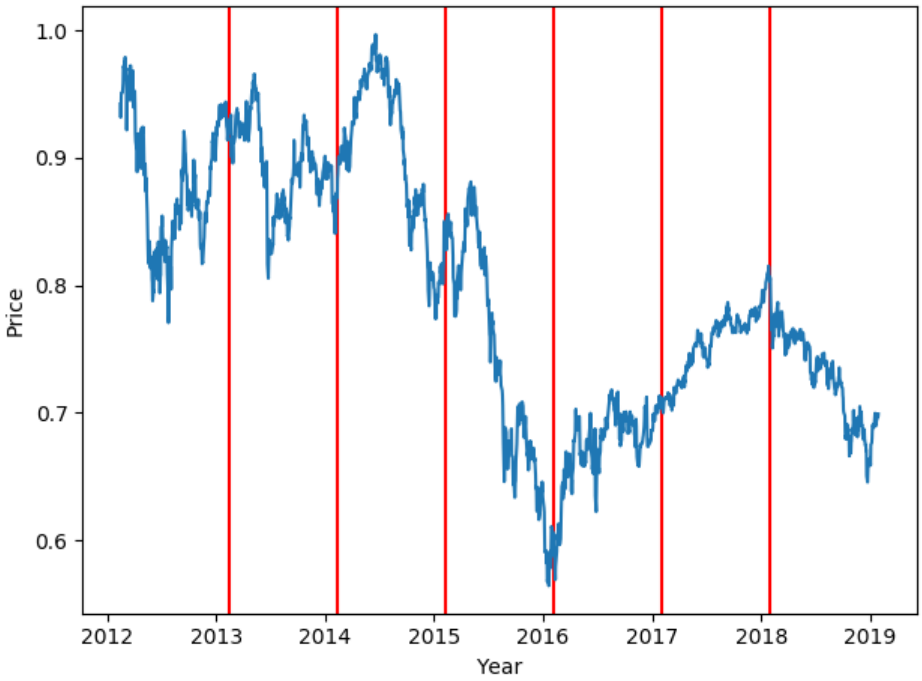


圖 72 DWX 走勢圖(收盤價)

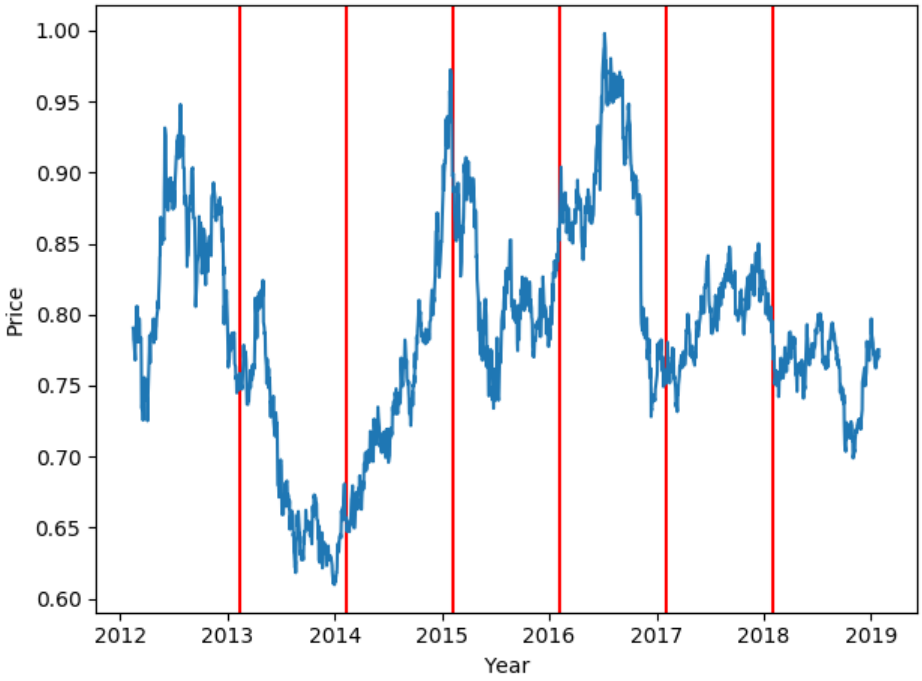


圖 73 EDV 走勢圖(收盤價)

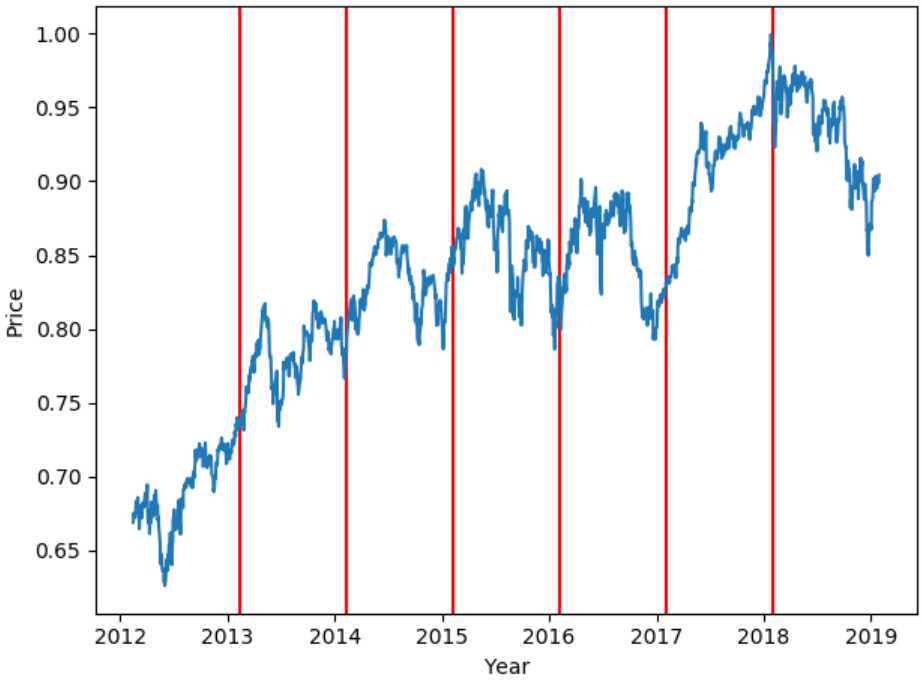


圖 74 EFAV 走勢圖(收盤價)

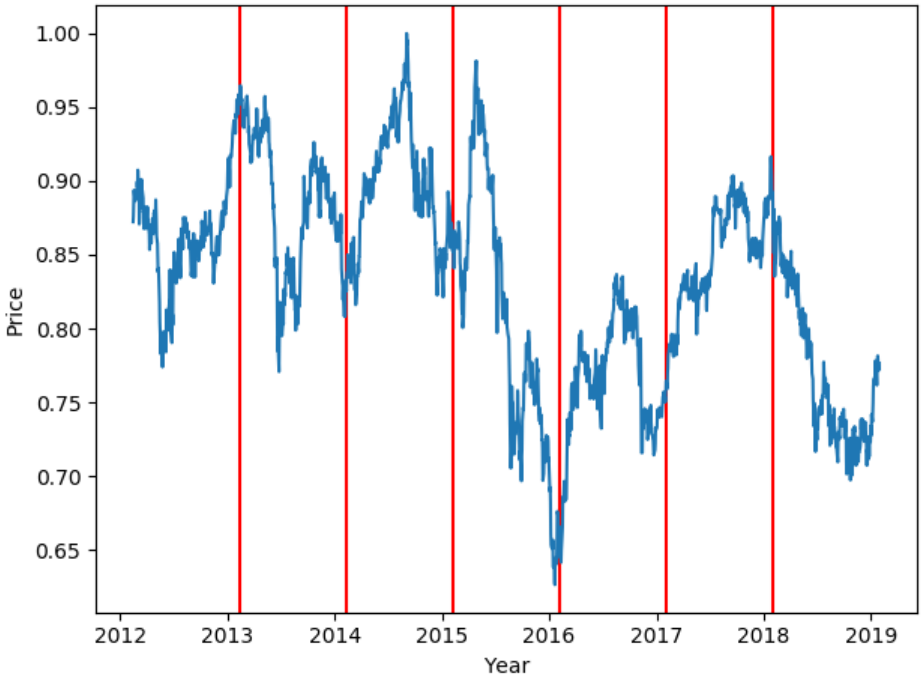


圖 75 EMIF 走勢圖(收盤價)

圖 76 EWV 走勢圖(收盤價)

圖 77 FAB 走勢圖(收盤價)

圖 78 FEX 走勢圖(收盤價)

圖 79 FEDI 走勢圖(收盤價)

圖 80 FXA 走勢圖(收盤價)

圖 81 FXC 走勢圖(收盤價)

圖 82 FXG 走勢圖(收盤價)

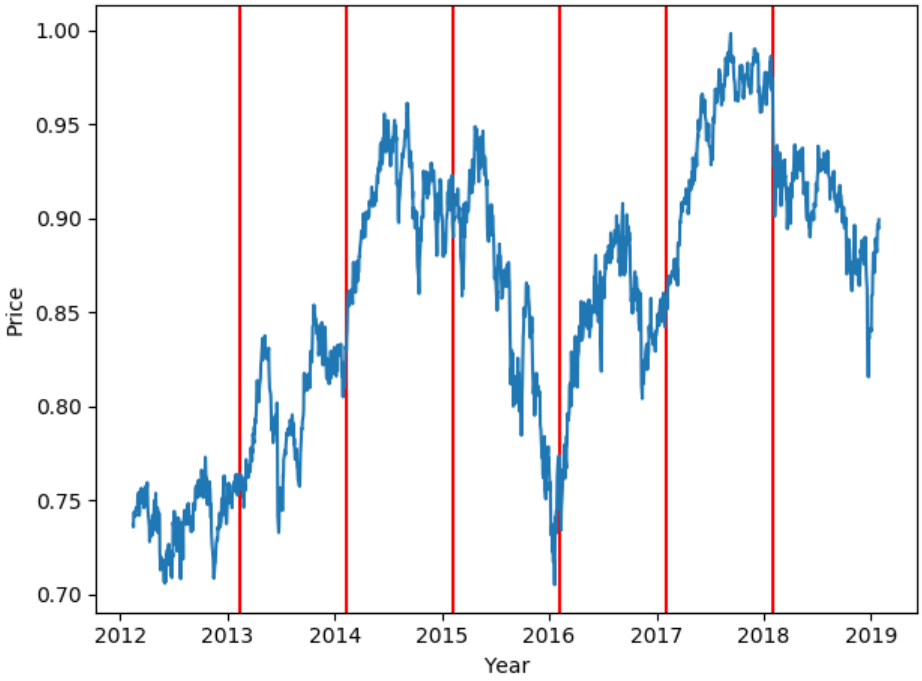


圖 83 GII 走勢圖(收盤價)

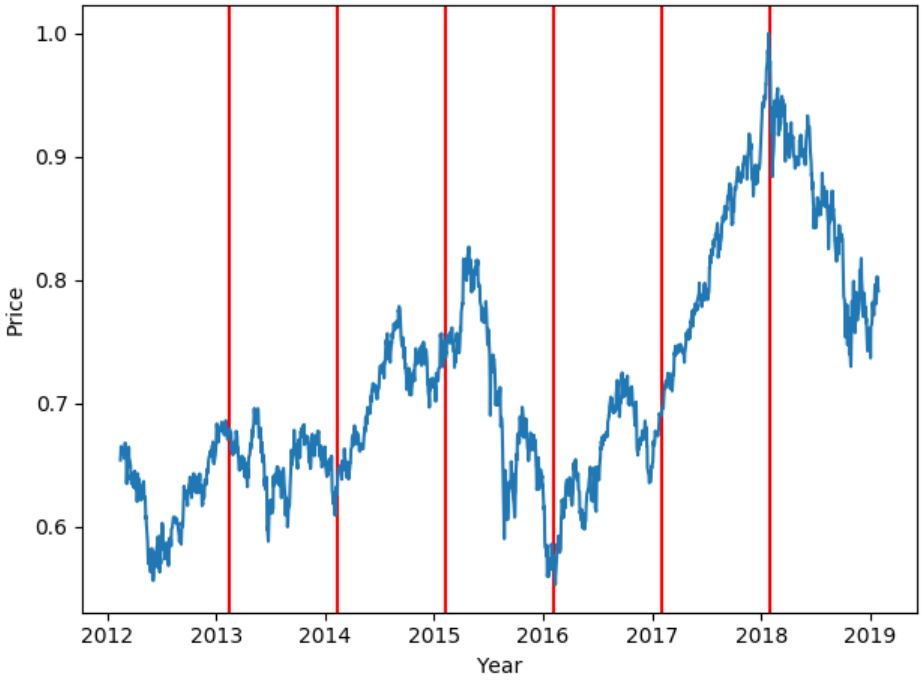


圖 84 GMF 走勢圖(收盤價)

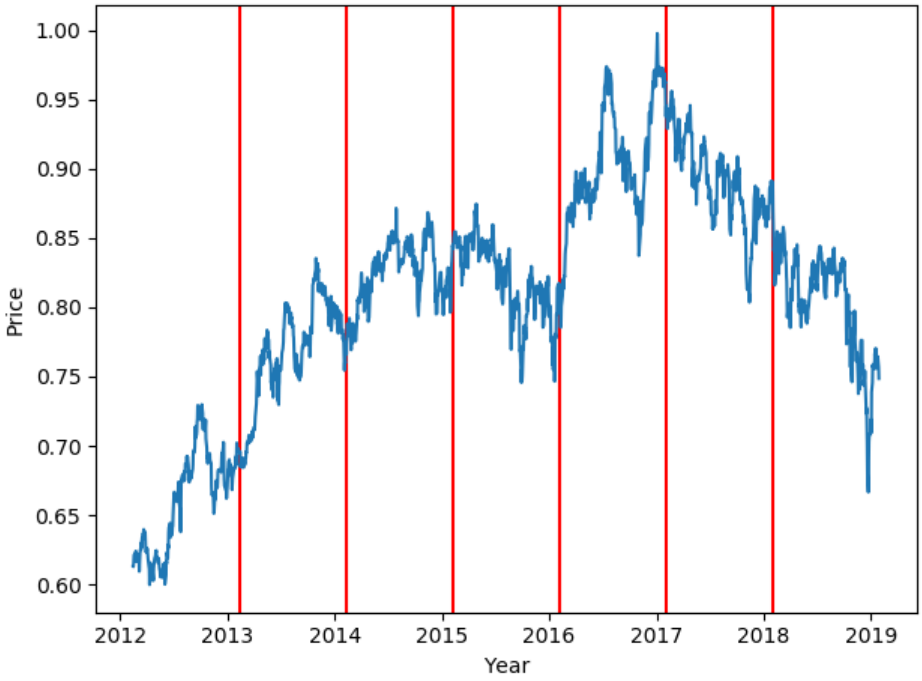


圖 85 VOX 走勢圖(收盤價)

圖 86 VPU 走勢圖(收盤價)

圖 87 VOT 走勢圖(收盤價)

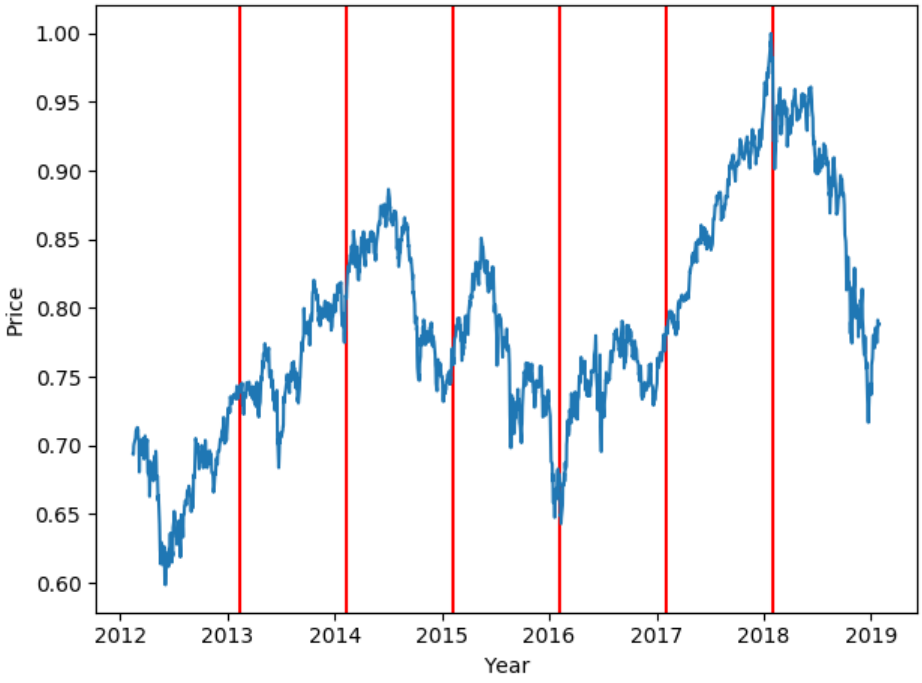


圖 88 VSS 走勢圖(收盤價)

圖 89 VTWV 走勢圖(收盤價)

圖 90 VV 走勢圖(收盤價)

圖 91 WEAT 走勢圖(收盤價)

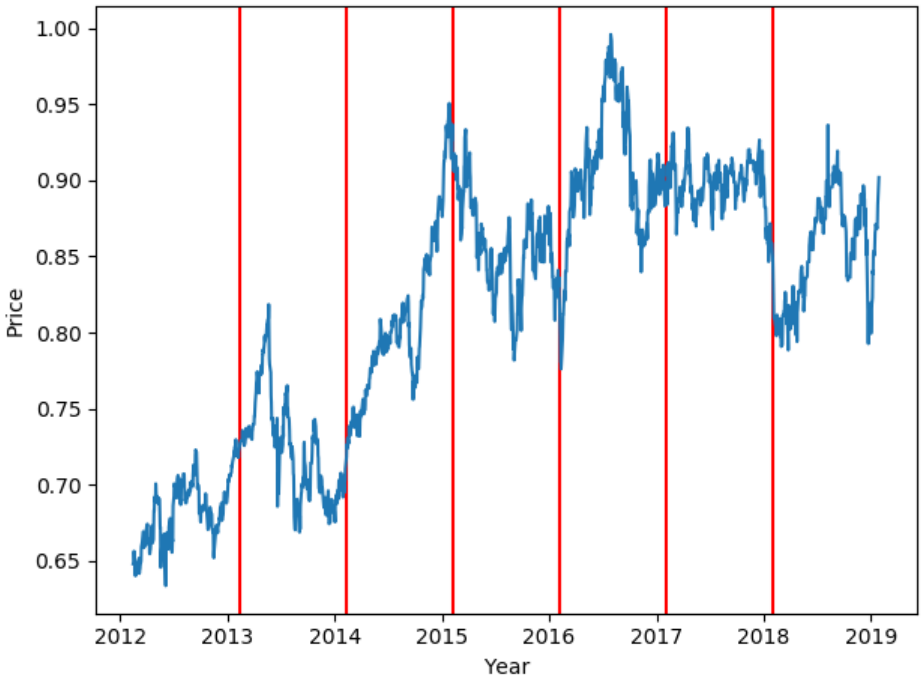


圖 92 WREI 走勢圖(收盤價)

圖 93 XLB 走勢圖(收盤價)

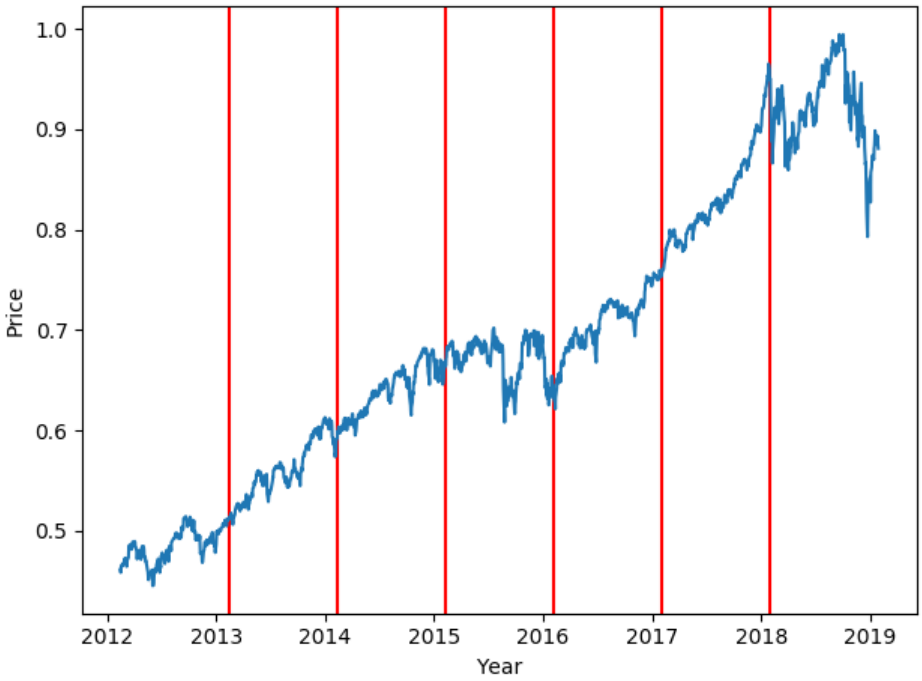


圖 94 XLG 走勢圖(收盤價)

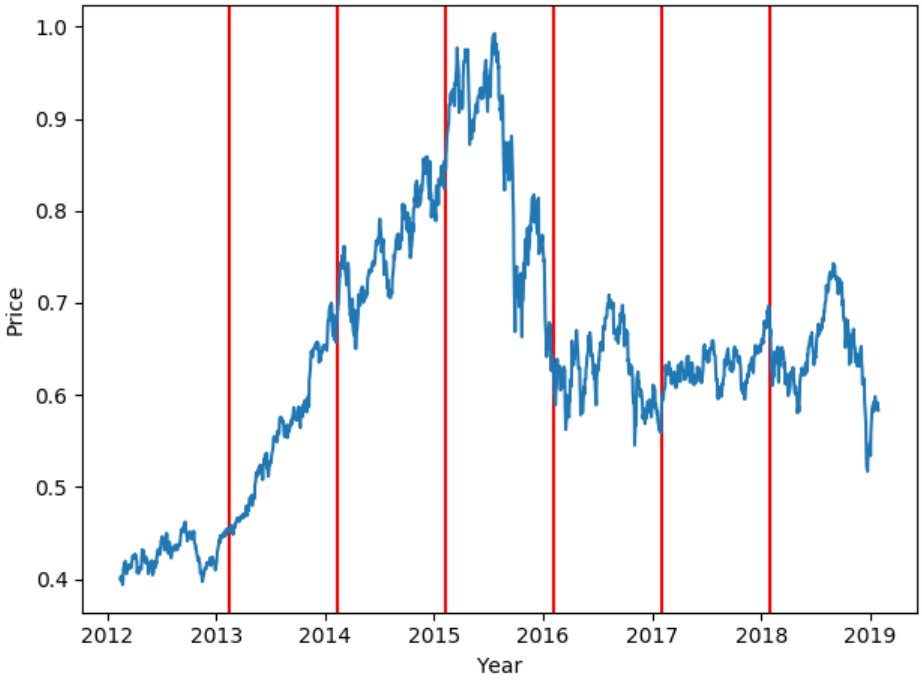


圖 95 XPH 走勢圖(收盤價)

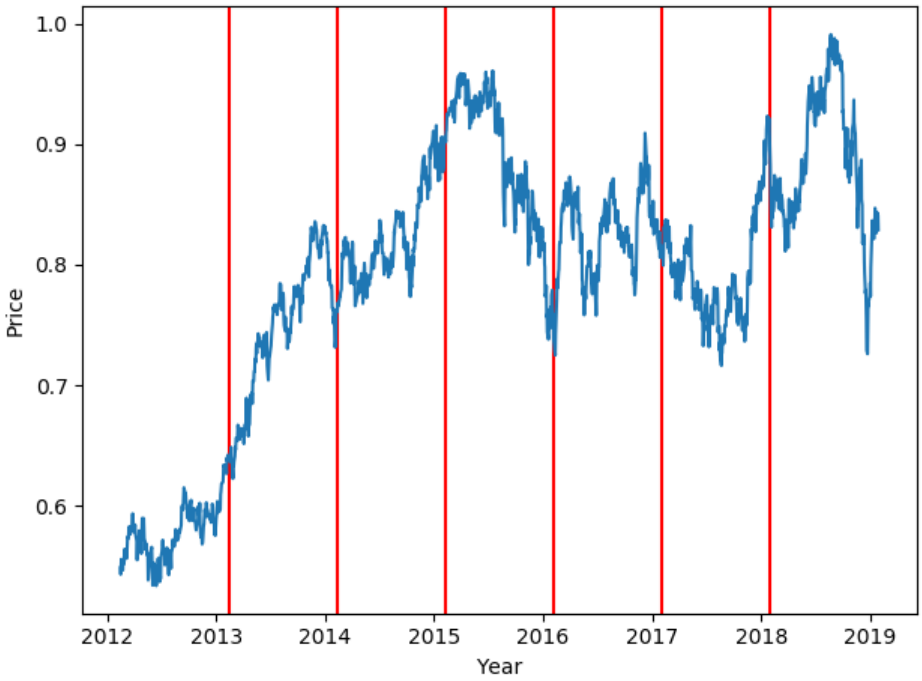


圖 96 XRT 走勢圖(收盤價)