前言

針對每個可想像得到的投資策略而言，最終會讓 *alpha* 歸零嗎？託眾多聰明人類與智慧電腦的福，使得金融市場著實成為完全競爭情況，因而可以安逸的坐享其成，認為所有資產定價正確，此成真之日將會到來嗎？

——Robert Shiller (2015)

人工智慧（AI）在 21 世紀 10 年代發展成為關鍵技術，而被認為是 21 世紀 20 年代的主要技術。在技術創新、演算法突破、巨量資料（大數據）可用程度與運算能力不斷增強的激勵下，許多行業都在 AI 驅動中發生重大變化。

雖然媒體與大眾的注意力大多集中在諸如遊戲與自駕車領域的突破，不過 AI 也已成為金融業的主要科技力。然而，金融 AI 仍處於初期階段——相較於網路搜尋或社交媒體行業而言，的確如此。

本書闡述與金融 AI 相關的諸多重要面向。金融 AI 已是範圍廣泛的議題，而單一著作得聚焦特定內容論述。因此，本書將探討相關基礎內容（參閱第一部分與第二部分）；以 AI（較具體而言是採用類神經網路）察覺金融市場的**統計無效率**情況（參閱第三部分）；這樣的無效率情況（由成功預測未來市場變化的 AI 演算法具體呈現）是演算法交易之中**經濟無效率**的利用前提（參閱第四部分）；能夠有系統的利用統計無效率與經濟無效率，證明與下列金融既定理論基石互相矛盾：**效率市場假說**（EMH），成功**交易機器人**的設計被認為是 AI 可能引領的金融聖杯之路，本書末尾探討 AI 對於金融業的結果與金融奇點的可能性（參閱第五部分）；另外還有技術附錄，以直接的 Python 程式碼從無到有建置類神經網路，並就此列舉應用範例（參閱第六部分）。

AI 應用於金融行業與應用於其他領域並沒有太大差異。2010 年代 AI 的重大突破，是將增強式學習（RL）應用於遊戲機台（諸如 1980 年代發行的 Atari，參閱 Mnih et al. 2013）以及棋盤遊戲（譬如西洋棋或圍棋，參閱 Silver et al. 2016）所造就的成果。別的領域不說，目前已將用在遊戲環境的 RL 所得經驗，用於設計與建置自駕車或改善醫療診斷這類具有挑戰的問題上。表 P-1 為 AI 與 RL 在不同領域的應用比較。

表 P-1　不同領域的 AI 比較

領域	代理人	目標	作法	獎勵	阻礙	風險
遊戲機台	AI 代理人（軟體）	分數最大化	虛擬遊戲環境 RL	點數與分數	計畫與延遲獎勵	無
自動駕駛	自駕車（軟體 + 車）	位置 A 到 B 的安全駕駛	虛擬（遊戲）環境 RL、實境駕駛測試	懲罰錯誤	虛擬到實際的轉換	財產損失、人類傷亡
金融交易	交易機器人（軟體）	長期績效最大化	虛擬交易環境 RL	金融報酬率	效率市場與競爭	金融損失

訓練 AI 代理人玩遊戲機台的優點是，具有完美的虛擬學習環境[1]而且毫無風險。至於自駕車，從虛擬學習環境（諸如《俠盜獵車手》這類電玩）轉到實際環境時（自駕車行駛於有車子與人們存在的實際街道上），將發生重大問題。如此導致汽車事故或人類傷亡這類嚴重的風險。

就交易機器人而言，RL 也可以完全虛擬，即處於模擬的金融市場環境中。交易機器人失常引起的主要風險是金融損失，以及由於交易機器人的從眾效應所產生的潛在系統風險。然而，整個金融領域似乎是訓練、測試與部署 AI 演算法的理想場域。

因為此領域的快速發展，對於願意關注與具有野心的學生來說，配備筆記型電腦與網際網路存取，就應該可以在金融交易環境中成功運用 AI。除了近年來硬體與軟體的改進之外，如此主要歸功於線上經紀商的興盛（提供歷史與即時金融資料以及能夠透過程式 API 進行金融交易）。

1　參閱 Arcade Learning Environment（*https://oreil.ly/bGgZs*）。

本書有下列六個部分：

第一部分

第一部分說明一般 AI 的主要概念與演算法，諸如監督式學習與類神經網路（參閱第一章）。也將提及超級智慧的概念，其中描述擁有人類智慧並在某些領域具備超人智慧的 AI 代理人（參閱第二章）。並不是所有 AI 研究人員都認為在可預見的將來能有超級智慧。然而，此一概念論述為 AI（尤其是金融 AI）的探討提供有用的框架。

第二部分

第二部分有四章，關於傳統的規範金融理論（參閱第三章），以及資料驅動金融（參閱第四章）與機器學習（ML，參閱第五章）讓金融領域有所變革。第六章則將資料驅動金融與 ML 結合產生 AI 第一的金融（無模型）作法。

第三部分

第三部分是以深度學習、類神經網路與增強式學習的應用，加以察覺金融市場的統計無效率情況。此部分包含密集神經網路（DNN，參閱第七章）、循環神經網路（RNN，參閱第八章）以及增強式學習演算法（RL，參閱第九章），而往往用 DNN 顯現與近似 AI 代理人的最佳政策。

第四部分

第四部分說明演算法交易利用統計無效率的情況。主題有向量化回測（參閱第十章）、事件式回測與風險管理（參閱第十一章），以及 AI 能力的演算法交易策略執行與部署（參閱第十二章）。

第五部分

第五部分是金融業 AI 式競爭所產生的相關結果（參閱第十三章）。其中還探討金融奇點的可能性，即 AI 代理人將主宰人類知曉金融所有面向的時間點。就此的論述聚焦於金融人工智慧，即不斷產生高於任何人類或機構基準交易利潤的交易機器人（參閱第十四章）。

第六部分

本書附錄包含互動的類神經網路訓練 Python 程式碼（參閱附錄 A），以 Python 程式碼從無到有實作簡單神經網路與淺層神經網路的類別（參閱附錄 B），以及使用卷積神經網路（CNN）進行金融時間序列預測的示例（參閱附錄 C）。

作者註解

金融交易中的 AI 應用仍然處於初期階段（儘管在撰寫本書時，還有許多書籍於某種程度上探討此一主題）。然而，這類著作未能說明**經濟上**利用統計無效率情況的意涵為何。

某些避險基金已宣稱完全仰賴「機器學習」管理投資者的資本。顯著的例子是 Voleon Group，此避險基金於 2019 年底有超過 60 億美元的資產管理規模（參閱 Lee and Karsh 2020）。以機器學習超越金融市場的困難度反映在此基金 2019 年達 7% 的業績上，這一年 S&P 500 股票指數上漲將近 30%。

本書基於筆者多年來在開發、回測與部署 AI 能力演算法交易策略相關的實務經驗。呈現的作法與範例大多以筆者自己的研究為基礎，因為本質上來說，這個領域不僅新興還相當隱秘。本書的闡述與風格是持續的以實用為主，在許多情況下，具體範例缺乏適當的理論支持與全面的實證證據。書中提供的某些應用與示例，甚至可能會受到金融或機器學習專家的強烈批判。

某些機器學習與深度學習的專家，譬如 François Chollet (2017)，公然質疑金融市場預測的可能性。某些金融專家，諸如 Robert Shiller (2015)，不相信會有像金融奇點這種論述。而活躍於兩個領域的其他人士，例如 Marcos López de Prado (2018) 則認為，將機器學習用於金融交易與投資，需要有大型團隊與龐大預算付出行業規模的努力才行。

本書不會試圖為所涵蓋的各個主題提供平衡的觀點或全面的參考。內容的呈現是基於筆者個人見解與經驗，還有提供具體範例與 Python 程式碼之際的實際考量所造就的結果。多數範例經過篩選與調整，以強調某些觀點或呈現激勵的結果。因此，可以認定的是，書中呈現的許多示例結果都歷經資料窺探與過度配適的效應（關於這些主題的探討，可參閱 Hilpisch 2020 ch. 4）。

本書的主要目標是讓讀者能夠利用書中程式範例作為框架，進而探索 AI 應用於金融交易的精彩之處。為了實現此一目標，本書始終依據一些簡化的假設，而主要以**金融時間序列資料**為基礎，**直接從此類資料中獲得特徵**。實務應用中，當然不必限於金融時間序列資料——也可以使用其他類型的資料與來源。本書對所得特徵的作法間接假設：金融時間序列與從其推得之特徵所顯示的樣式（型態），至少在某種程度上會隨著時間持續留存，進而可用於預測未來價格動向。

以此背景而言，本書呈現的所有範例與程式碼皆為技術與說明的性質，不具有任何推薦含意或投資建議。

針對想要部署本書作法與演算法交易策略的讀者而言，筆者的著作《Python for Algorithmic Trading: From Idea to Cloud Deployment》（O'Reilly）有提供更多程序導向與技術細節內容。這兩本書在諸多面向是相輔相成的。對於剛剛起步使用 Python 進行金融應用或尋求複習與參考書的讀者來說，筆者的著作《Python for Finance: Mastering Data-Driven Finance》（O'Reilly）完整涵蓋 Python 應用於金融領域的重要主題與基本技能。

參考文獻

下列為前言所引用的論文與書籍：

Chollet, François. 2017. *Deep Learning with Python*. Shelter Island: Manning.

Hilpisch, Yves. 2018. *Python for Finance: Mastering Data-Driven Finance.* 2nd ed. Sebastopol: O'Reilly.

———. 2020. *Python for Algorithmic Trading: From Idea to Cloud Deployment.* Sebastopol: O'Reilly.

Lee, Justina and Melissa Karsh. 2020. "Machine-Learning Hedge Fund Voleon Group Returns 7% in 2019." *Bloomberg*, January 21, 2020. *https://oreil.ly/TOQiv*.

López de Prado, Marcos. 2018. *Advances in Financial Machine Learning*. Hoboken, NJ: John Wiley & Sons.

Mnih, Volodymyr et al. 2013. "Playing Atari with Deep Reinforcement Learning." arXiv. December 19. *https://oreil.ly/-pW-1*.

Shiller, Robert. 2015. "The Mirage of the Financial Singularity." Yale Insights. July 16. *https://oreil.ly/VRkP3*.

Silver, David et al. 2016. "Mastering the Game of Go with Deep Neural Networks and Tree Search." *Nature* 529 (January): 484-489.

規範金融

> *CAPM* 依據許多不切實際的假設。例如，假設投資人只關心投資組合一期的報酬平均數與變異數，這是過於極端的假設。
>
> ——Eugene Fama and Kenneth French (2004)

> 與人類（而非基本粒子）相關的科學對典雅數學呈現頑強抗拒力。
>
> ——Alon Halevy et al. (2009)

本章論述規範金融中主要的理論與模型。就本書的宗旨並簡單來說，**規範理論**（*normative theory*）是依據某些假設（數學公理），並從相關假設中得出對應見解、結果等內容。另一方面，**實證理論**（*positive theory*）是根據觀測、實驗、資料、關係等方面，而從現有資訊與衍生結果中獲得見解，並以見解描述相關現象。Rubinstein (2006) 對於本章提及之理論與模型的淵源，有詳細論述。

第 62 頁〈不確定性與風險〉介紹金融建模的主要概念，諸如不確定性、風險、交易資產等等。第 66 頁〈預期效用理論〉探討不確定性下決策的主要經濟範式：**預期效用理論**（EUT）。EUT 的近代樣式可以追溯到 von Neumann and Morgenstern (1944)。第 72 頁〈平均數—變異數投資組合理論〉介紹 Markowitz (1952) 的平均數—變異數投資組合（MVP）理論。第 83 頁〈資本資產定價模型〉解析 Sharpe (1964) 與 Lintner (1965) 的**資本資產定價模型**（CAPM）。第 91 頁〈套利定價理論〉概述 Ross (1971, 1976) 的**套利定價理論**（APT）。

本章以主要的規範金融理論作為本書的基礎。這一點很重要，因為數個世代的經濟學家、金融分析師、資產經理人、交易員、銀行家、會計師等等皆受過這些理論薰陶。就此意義而言，金融肯定能作為一門理論與實務兼具的學科，有很大的範疇是由這些理論所造就的。

不確定性與風險

金融理論的核心是，於不確定性與風險存在的情況下投資、交易與評價。本節會以某種程度的數學形式介紹與這些主題相關的主要概念。重點是機率論的基本概念，這些概念是計量財務金融的骨幹[1]。

定義

假設只在兩個時間點觀測活動的經濟結構：今日、$t = 0$；一年後、$t = 1$。本章稍後討論的金融理論有很大程度都是以如此**靜態經濟結構**（*static economy*）為基礎[2]。

當 $t = 0$ 時，並無存在不確定性。當 $t = 1$ 時，經濟結構可以接受（有限量）S 個可能狀態 $\omega \in \Omega = \{\omega_1, \omega_2, ..., \omega_S\}$。$\Omega$ 是**狀態空間**（*state space*），其基數（cardinality）為 $|\Omega| = S$。

Ω 的（集合）**代數**（*algebra*）$\mathscr{F}$ 是具有下列內容的一組集合：

1. $\Omega \in \mathscr{F}$
2. $\mathbb{E} \in \mathscr{F} \Rightarrow \mathbb{E}^c \in \mathscr{F}$
3. $\mathbb{E}_1, \mathbb{E}_2, \ldots, \mathbb{E}_I \in \mathscr{F} \Rightarrow \cup_{i=1}^{I} \mathbb{E}_i \in \mathscr{F}$

$\mathbb{E}^C$ 為集合 $\mathbb{E}$ 的補集（complement）。冪集（power set）$\wp(\Omega)$ 是最大的（集合）代數，而集合 $\mathscr{F} = \{\varnothing, \Omega\}$ 是 Ω 的最小（集合）代數。代數是經濟結構中**可觀測事件**（*observable event*）的模型。在這種情況下，經濟結構的單一狀態 $\omega \in \Omega$ 可詮釋為**原子事件**（*atomic event*）。

狀態 $\omega \in \Omega$ 發生的機率是實數 $0 \leq p_\omega \equiv P(\{\omega\}) \leq 1$，或事件 $\mathbb{E} \in \mathscr{F}$ 發生的機率是實數 $0 \leq P(\mathbb{E}) \leq 1$。若已知所有狀態發生的機率，則 $P(\mathbb{E}) = \Sigma_{\omega \in \mathbb{E}} P_\omega$。

1 關於機率論的介紹內容可參閱 Jacod and Protter (2004)。

2 在**動態**經濟結構中，不確定性會逐漸解決（例如，今後一年間的每一日逐漸排除）。

機率測度（*probability measure*）$P:\mathscr{F} \to [0,1]$ 具有下列性質：

1. $\forall \mathbb{E} \in \mathscr{F}: P(\mathbb{E}) \geq 0$
2. 對於互斥集（disjoint set）$\mathbb{E}_i \in \mathscr{F}$ 而言，$P\left(\cup_{i=1}^{I} \mathbb{E}_i\right) = \Sigma_{i=1}^{I} \mathbb{E}_i$
3. $P(\Omega) = 1$

三元素 $\{\Omega, \mathscr{F}, P\}$ 集結形成**機率空間**。機率空間是此模型經濟結構中**不確定性**的數學形式表徵。若機率測度 P 固定，則稱經濟結構處於**風險**之中。若經濟結構中的所有代理人已知此風險，則稱此經濟結構具有**對稱資訊**（*symmetric information*）。

已知機率空間 $\{\Omega, \mathscr{F}, P\}$，而**隨機變數**為函數 $S:\Omega \to \mathbb{R}_+, \omega \mapsto S(\omega)$，其為 $\mathscr{F}$—可測的。如此意味著，對每個 $\mathbb{E} \in \{[a, b[: a, b \in \mathbb{R}, a < b\}$ 有下列結果：

$$S^{-1}(\mathbb{E}) \equiv \{\omega \in \Omega : S(\omega) \in \mathbb{E}\} \in \mathscr{F}$$

若 $\mathscr{F} \equiv \wp(\Omega)$，則下列為隨機變數的**期望值**（*expectation*）定義：

$$\mathbf{E}^P(S) = \sum_{\omega \in \Omega} P(\omega) \cdot S(\omega)$$

不然期望值的定義如下：

$$\mathbf{E}^P(S) = \sum_{\mathbb{E} \in \mathscr{F}} P(\mathbb{E}) \cdot S(\mathbb{E})$$

通常，假設某個金融經濟結構為**完全競爭**（*perfect*），其意味著，別的不說，此時沒有交易成本，可用的資產具有固定價格而且數量無限，一切活動以光速進行，代理人具有完整與對稱的資訊。

數值範例

此時假設處於**風險** $\{\Omega, \mathscr{F}, P\}$ 下的簡單**靜態**經濟結構，而存在以下情況：

1. $\Omega \equiv \{u, d\}$
2. $\mathscr{F} \equiv \wp(\Omega)$
3. $P \equiv \left\{P(\{u\}) = \frac{1}{2}, P(\{d\}) = \frac{1}{2}\right\}$

交易資產

此經濟結構中交易兩筆資產。第一項是風險資產——**股票**，今日確定價格為 $S_0 = 10$，而明日不確定的 payoff 以下列隨機變數形式表示：

$$S_1 = \begin{cases} S_1^u = 20 \text{ if } \omega = u \\ S_1^d = 5 \quad \text{if } \omega = d \end{cases}$$

第二項是無風險資產——**債券**，今日確定價格為 $B_0 = 10$，明日確定的 payoff 如下：

$$B_1 = \begin{cases} B_1^u = 11 \text{ if } \omega = u \\ B_1^d = 11 \text{ if } \omega = d \end{cases}$$

數學形式上，此模型經濟結構可以寫作 $\mathscr{M}^2 = (\{\Omega, \mathscr{F}, P\}, \mathbb{A})$，其中 $\mathbb{A}$ 為可交易資產，其形式為今日的價格向量 $M_0 = (S_0, B_0)^T$ 以及下列明日市場 payoff 矩陣：

$$M_1 = \begin{pmatrix} S_1^u & B_1^u \\ S_1^d & B_1^d \end{pmatrix}$$

套利定價

以此經濟結構，可以解決如下的問題：以股票履約價 $K = 14.5$ 導出**歐式買權**（*European call option*）的公允價值（fair value）。歐式買權無套利的價值 C_0 是依股票與債券的投資組合 ϕ 複製此選擇權的 payoff（C_1）推導而來。此複製投資組合的價格也必然是歐式買權的價格。否則將存在（無限）套利空間。以 Python 運用這樣的複製論點，輕而易舉[3]：

```
In [1]: import numpy as np

In [2]: S0 = 10  ❶
        B0 = 10  ❶

In [3]: S1 = np.array((20, 5))  ❷
        B1 = np.array((11, 11))  ❷

In [4]: M0 = np.array((S0, B0)) ❸
        M0  ❸
```

3　有關套利之風險中立評價以及評價相關的詳細資訊，可參閱 Hilpisch (2015, ch. 4)。

```
Out[4]: array([10, 10])

In [5]: M1 = np.array((S1, B1)).T  ❹
        M1  ❹
Out[5]: array([[20, 11],
               [ 5, 11]])

In [6]: K = 14.5  ❺

In [7]: C1 = np.maximum(S1 - K, 0)  ❻
        C1  ❻
Out[7]: array([5.5, 0. ])

In [8]: phi = np.linalg.solve(M1, C1)  ❼
        phi  ❼
Out[8]: array([ 0.36666667, -0.16666667])

In [9]: np.allclose(C1, np.dot(M1, phi))  ❽
Out[9]: True

In [10]: C0 = np.dot(M0, phi)  ❾
         C0  ❾
Out[10]: 2.0
```

❶ 股票與債券的今日價格。

❷ 股票（不確定）與債券（確定）明日的 payoff。

❸ 市價向量。

❹ 市場 payoff 矩陣。

❺ 選擇權的履約價。

❻ 選擇權的不確定 payoff。

❼ 複製投資組合 ϕ。

❽ 確認其 payoff 是否與選擇權的 payoff 相同。

❾ 複製投資組合的價格是選擇權的無套利價格。

套利定價

如上述範例所示，可將套利定價理論視為最強大的金融理論之一，其具備某些較為紮實的數學結果，如*資產定價基本定理*（*FTAP*）[4]。姑且不論別的原因，而可以從其他可觀測的市場參數（如：賣權交易標的股價）推出選擇權價格的事實所致。就此意義而言，套利定價先不考慮如何得出公允股價，而只是將其作為輸入內容。因此，套利定價使用少量且緩和的假設行事，凡是*少了套利*，對許多金融理論來說都無法接受。注意，其中也沒有採用機率測度得出套利價格。

預期效用理論

預期效用理論（EUT）是金融理論的基石。自 1940 年代提出以來，一直是不確定性之下決策建模的主要範式之一[5]。關於金融與投資理論的每本入門教科書通常都會有 EUT 的介紹。其中一個原因是，金融方面的其他主要結果可以從 EUT 範式中得出。

假設與結果

EUT 是個不證自明理論（axiomatic theory），可以追溯到 von Neumann and Morgenstern (1944) 的首創研究。*不證自明*在此意味著這個理論的主要結果只能從少量的公理中推導出來。有關公理效用理論各種變形與應用的概括論述，可參閱 Fishburn (1968)。

公理與規範理論

在 Wolfram MathWorld（*https://oreil.ly/pZqal*）中，可以找到針對*公理*（*axiom*）的定義：「公理是不經證明就顯而易見的真實陳述」。

當面對不確定性下的抉擇時，EUT 通常以代理人*偏好*的一小部分主要公理為基礎。雖然公理的定義另有論述，但是並非所有公理都被經濟學家們視為「不經證明就顯而易見的真實」。

4　請參閱 Hilpisch (2015, ch. 4) 以及其中的參考文獻。

5　更多背景與細節，可參閱 Eichberger and Harper (1997, ch. 1) 或 Varian (2010, ch. 12)。

Von Neumann and Morgenstern (1944, p. 25) 對於公理的抉擇評論是：

> 公理的抉擇並非全然客觀的任務。通常可以預期達到某個明確目標——某些特定的定理可以從公理中推導出來——而且就此範疇來說，問題是確切與客觀的。但除此之外，總會有本質上不那麼確切的其他重要之物：公理不宜過多，其系統盡可能簡單明瞭，每個公理應具有直覺意涵，進而直接判斷其適當性。

依此意義而言，一套公理構成一個**規範理論**，而以此理論塑造世界（或世界的一部分）。一套公理集結某個先驗（priori）應滿足的最小假設集合，而不是以某種數學形式證明或類似作為。在列出造就 EUT 的一套公理之前，於不確定性之下抉擇呈現之際，會有些關於代理人自身偏好的用詞（數學形式是 $\succeq$）。

代理人的偏好

假設具有偏好 $\succeq$ 的代理人面臨投資模型經濟結構 $\mathscr{M}^2$ 兩種交易資產的投資問題。例如，代理人可以在未來 payoff 為 $A = \phi_A \cdot M_1$ 的投資組合 ϕ_A 或未來 payoff 為 $B = \phi_B \cdot M_1$ 的投資組合 ϕ_B 之間做抉擇。假設代理人的偏好 $\succeq$ 是依據未來 payoff（而非投資組合）所定義。若代理人偏好 payoff A（強）優於 payoff B，則以 $A \succ B$ 表示，反之則以 $A \prec B$ 代表。若代理人對於兩者偏好無差別，則以 $A \sim B$ 表示。鑒於這些敘述，造就如下所述的一套 EUT 可能公理：

完整性（*completeness*）

代理人依相對的彼此，可將**所有**的 payoff 排行。即下列其中一項必定成立：$A \succ B$、$A \prec B$ 或 $A \sim B$。

遞移性（*transitivity*）

若有第三個投資組合 ϕ_C 搭配未來 payoff $C = \phi_C \cdot M_1$，則由 $A \succ B$ 且 $B \succ C$，可得 $A \succ C$。

連續性（*continuity*）

若 $A \succ B \succ C$，則存在一個數值 $\alpha \in [0, 1]$，使得 $B \sim \alpha A + (1 - \alpha)C$。

獨立性（*independence*）

若 $A \sim B$ 則 $\alpha A + (1 - \alpha)C \sim \alpha B + (1 - \alpha)C$。
同理，
若 $A \succ B$ 則 $\alpha A + (1 - \alpha)C \succ \alpha B + (1 - \alpha)C$。

優勢性（*dominance*）

若 $C_1 = \alpha_1 A + (1-\alpha_1)C$ 且 $C_2 = \alpha_2 A + (1-\alpha_2)C$，則由 $A \succ C$ 且 $C_1 \succ C_2$ 可得 $\alpha_1 > \alpha_2$。

效用函數

效用函數（*utility function*）是以數學與數值方式表示代理人偏好 $\succeq$ 的作法，即這樣的函數會對某個 payoff 賦予某一數值。在此情況下，數值的絕對內容無關緊要，而數值衍生的順序值得關注[6]。假設 $\mathbb{X}$ 代表代理人能夠表達其偏好的所有可能 payoff。則效用函數 U 定義如下：

$$U: \mathbb{X} \to \mathbb{R}_+, x \mapsto U(x)$$

若 U 表示代理人的偏好 $\succeq$，則存在下列關係：

$$\begin{aligned}
&A \succ B \Rightarrow U(A) > U(B) && \text{（強優於）}\\
&A \succeq B \Rightarrow U(A) \geq U(B) && \text{（弱優於）}\\
&A \prec B \Rightarrow U(A) < U(B) && \text{（強劣於）}\\
&A \preceq B \Rightarrow U(A) \leq U(B) && \text{（弱劣於）}\\
&A \sim B \Rightarrow U(A) = U(B) && \text{（無差別）}
\end{aligned}$$

效用函數 U 只能作正線性轉換。因此，若 U 表示偏好 $\succeq$，則 $V = a + bU$（其中 $a, b > 0$）也是如此作為。關於效用函數，von Neumann and Morgenstern (1944, p. 25) 總結如下：「因此得知：若真的存在這樣的效用數值評價，則必定是作線性轉換。即，效用就是線性轉換的數值。」

預期效用函數

Von Neumann and Morgenstern (1944) 表示，若代理人的偏好 $\succeq$ 滿足前述五個公理，則存在對應的**預期效用函數**（*expected utility function*）：

$$U: \mathbb{X} \to \mathbb{R}_+, x \mapsto \mathbf{E}^P(u(x)) = \sum_{\omega}^{\Omega} P(\omega) u(x(\omega))$$

在此 $u: \mathbb{R} \to \mathbb{R}, x \mapsto u(x)$ 是單調遞增、狀態獨立的函數，往往稱為 ***Bernoulli* 效用函數**，諸如 $u(x) = \ln(x)$、$u(x) = x$ 或 $u(x) = x^2$。

6 通常稱之為**序號**。街道房子的門牌號碼就是貼切的序號示例。

換句話說，預期效用函數 U 將函數 u 套用於某狀態的 payoff（$x(\omega)$），並使用已知狀態發生的機率 $P(\omega)$ 對單一效用加權。對於特殊的線性 Bernoulli 效用函數 $u(x) = x$ 而言，預期效用就是狀態相關 payoff 的期望值——$U(x) = \mathbf{E}^P(x)$。

風險趨避

風險趨避（*risk aversion*）是重要的金融概念。風險趨避最常用的衡量方法是**絕對風險趨避**（ARA）之 Arrow-Pratt 衡量法，其可追溯到 Pratt(1964)。假設代理人的狀態獨立 Bernoulli 效用函數為 $u(x)$。則 ARA 之 Arrow-Pratt 衡量法的定義為：

$$ARA(x) = -\frac{u''(x)}{u'(x)}, x \geq 0$$

依據此衡量方法可以分成下列三種情況：

$$ARA(x) = -\frac{u''(x)}{u'(x)}\begin{cases} > 0 & \text{風險趨避} \\ = 0 & \text{風險中立} \\ < 0 & \text{風險愛好} \end{cases}$$

在金融理論與模型中，一般假定的適當情況為風險趨避與風險中立。而在博弈之中，可能也會找到風險愛好的代理人。

依據之前提及的三個 Bernoulli 函數：$u(x) = \ln(x)$、$u(x) = x$ 或 $u(x) = x^2$。可以輕易驗證其分別塑造的風險趨避、風險中立以及風險愛好三種代理人。例如 $u(x) = x^2$：

$$-\frac{u''(x)}{u'(x)} = -\frac{2}{2x} < 0, x > 0 \Rightarrow \text{風險愛好}$$

數值範例

此 EUT 應用可輕易以 Python 舉例說明。假設範例的模型經濟結構承襲上一節的 $\mathscr{M}^2$。偏好 $\succeq$ 之代理人依據的 EUT ，是以不同的未來 payoff 而定。此代理人的 Bernoulli 效用函數是 $u(x) = \sqrt{x}$。此範例中，投資組合 ϕ_A 造就的 payoff（A_1）優於投資組合 ϕ_D 造就的 payoff（D_1）。

下列為此應用的程式碼：

```
In [11]: def u(x):
             return np.sqrt(x)  ❶
```

```
In [12]: phi_A = np.array((0.75, 0.25))  ❷
         phi_D = np.array((0.25, 0.75))  ❷

In [13]: np.dot(M0, phi_A) == np.dot(M0, phi_D)  ❸
Out[13]: True

In [14]: A1 = np.dot(M1, phi_A)  ❹
         A1  ❹
Out[14]: array([17.75,  6.5 ])

In [15]: D1 = np.dot(M1, phi_D)  ❺
         D1  ❺
Out[15]: array([13.25,  9.5 ])

In [16]: P = np.array((0.5, 0.5))  ❻

In [17]: def EUT(x):
             return np.dot(P, u(x))  ❼

In [18]: EUT(A1)  ❽
Out[18]: 3.381292321692286

In [19]: EUT(D1)  ❽
Out[19]: 3.3611309730623735
```

❶ 風險趨避的 Bernoulli 效用函數。

❷ 權重不同的兩個投資組合。

❸ 顯示每個投資組合設置的成本相同。

❹ 一個投資組合（不確定）的 payoff，

❺ 以及另一個投資組合（不確定）的 payoff。

❻ 機率測度。

❼ 預期效用函數。

❽ 兩個（不確定） payoff 的效用值。

在此情況下，典型的問題是，已知代理人的固定預算 $w > 0$，要得出最佳投資組合（即預期效用最大化）。下列 Python 程式碼會為此問題建模，並確切的解決。在可用預算中，代理人將 60％左右的額度投入風險資產中，而約 40％的額度置於無風險資產中。結果主要由特定類型的 Bernoulli 效用函數而定：

```
In [20]: from scipy.optimize import minimize

In [21]: w = 10  ❶

In [22]: cons = {'type': 'eq', 'fun': lambda phi: np.dot(M0, phi) - w}  ❷

In [23]: def EUT_(phi):
             x = np.dot(M1, phi)  ❸
             return EUT(x)  ❸

In [24]: opt = minimize(lambda phi: -EUT_(phi),  ❹
                        x0=phi_A,  ❺
                        constraints=cons)  ❻

In [25]: opt  ❼
Out[25]:      fun: -3.385015999493397
              jac: array([-1.69249132, -1.69253424])
          message: 'Optimization terminated successfully.'
             nfev: 16
              nit: 4
             njev: 4
           status: 0
          success: True
                x: array([0.61122474, 0.38877526])

In [26]: EUT_(opt['x'])  ❽
Out[26]: 3.385015999493397
```

❶ 代理人的固定預算。

❷ 以 minimize[7] 限制預算的使用。

❸ 針對投資組合定義預期效用函數。

❹ -EUT_(phi) 最小化（即 EUT_(phi) 最大化）。

❺ 優化的初始猜值。

❻ 套用限制的預算。

❼ 最佳結果，包含 x 的最佳投資組合。

❽ 在 $w = 10$ 預算下最佳（最高）預期效用。

7 細節可參閱 *http://bit.ly/aiif_minimize*。

平均數—變異數投資組合理論

Markowitz (1952) 的平均數—變異數投資組合（MVP）理論是金融理論的另一個基石。其為不確定性方面最早的投資理論之一，只聚焦於股票投資組合建構的統計衡量。MVP 將可能影響公司股票表現的基本面，或者對公司前景成長至關重要的未來競爭力假設完全抽離。基本上，唯一計數的輸入資料是股價的時間序列，以及從中得出的統計內容，諸如（歷史）年化報酬率平均數以及（歷史）年化報酬率變異數。

假設與結果

依據 Markowitz (1952) 的論述，MVP 的主要假設是，投資人只關心預期報酬率與這些報酬率的變異數：

> 接著考量以下規則：投資人確實（或應該）認為預期報酬率是討喜之物，而報酬率的變異數是不討喜之物。此規則有許多合理之處，既可以作為投資行為的準則，也可以作為投資行為的前提。
>
> 預期報酬率為最大的投資組合，不一定是變異數為最小的投資組合。投資人可以接受變異數而獲得預期報酬率，或者放棄預期報酬率以降低變異數。

此種投資人偏好的作法，與直接定義代理人偏好以及 payoff 效用函數的作法大相逕庭。MVP 則是認為可以在投資組合預期產生報酬率的一次及二次動差（first and second moment），定義代理人的偏好與效用函數。

> **隱含假設的常態分布**
>
> 一般來說，MVP 理論只關注投資組合於一個期間的風險與報酬，此與標準 EUT 並不相容。此議題的解法是，假設風險資產的報酬率是常態分布，使得一次及二次動差可充分描述資產報酬率的完整分布。下一章將說明，這在實際的金融資料中幾乎未曾出現。另一種方式是假設某個特定的二次式 Bernoulli 效用函數，如下一節所示。

投資組合統計

假定靜態經濟結構 $\mathscr{M}^N = (\{\Omega, \mathscr{F}, P\}, \mathbb{A})$，其可交易資產集合 $\mathbb{A}$，是由 N 個風險資產 $A^1, A^2, ..., A^N$ 所組成。以 A_0^n 為資產 n 今日的固定價格，而 A_1^n 為其一年的 payoff，則資產 n 的（簡單）淨報酬率向量 r^n 定義如下：

$$r^n = \frac{A_1^n}{A_0^n} - 1$$

對於具有相同發生機率的所有未來狀態而言，資產 n 的**預期報酬率**（*expected return*）為：

$$\mu^n = \frac{1}{|\Omega|}\sum_{\omega}^{\Omega} r^n(\omega)$$

因此，**預期報酬率的向量**如下：

$$\mu = \begin{bmatrix} \mu^1 \\ \mu^2 \\ \vdots \\ \mu^N \end{bmatrix}$$

投資組合（向量）為 $\phi = (\phi^1, \phi^2, ..., \phi^N)^T$，其中 $\phi_n \geq 0$ 以及 $\Sigma_n^N \phi^n = 1$，而投資組合裡的每個資產附有權重[8]。

然後，**投資組合的預期報酬率**則是投資組合加權向量與預期報酬率向量的點積（dot product）：

$$\mu^{phi} = \phi \cdot \mu$$

此時藉由下列內容定義資產 n 與資產 m 之間的**共變異數**（*covariance*）：

$$\sigma_{mn} = \sum_{\omega}^{\Omega}\left(r^m(\omega) - \mu^m\right)\left(r^n(\omega) - \mu^n\right)$$

而**共變異數矩陣**（*covariance matrix*）如下：

$$\Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} \end{bmatrix}$$

8　這些假設並非真的必要。例如，在不會顯著變更分析的情況下，可以允許賣空（short sale）。

投資組合的預期變異數則是下列雙點積：

$$\varphi^{phi} = \phi^T \cdot \Sigma \cdot \phi$$

因此投資組合的預期**波動率**（*volatility*）如下：

$$\sigma^{phi} = \sqrt{\varphi^{phi}}$$

Sharpe ratio

Sharpe (1966) 提出一種衡量指標，可判斷共同基金以及其他投資組合、甚至是單一風險資產於風險調整後的績效。其最簡單的形式是將投資組合的（預期、已實現）報酬率與其（預期、已實現）波動率相關聯。因此就數學形式而言，*Sharpe ratio* 的定義如下：

$$\pi^{phi} = \frac{\mu^{phi}}{\sigma^{phi}}$$

若 r 表示無風險短期利率，則投資組合 *phi* 相對無風險投資的**風險溢酬**（*risk premium*）或**超額報酬**（*excesse return*）為 $\mu^{phi} - r$。Sharpe ratio 的另一種形式是風險溢酬位於分子：

$$\pi^{phi} = \frac{\mu^{phi} - r}{\sigma^{phi}}$$

若無風險短期利率相對較低，則在套用相同的無風險短期利率時，兩種 Sharpe ratio 版本的數值結果差別不大。特別是以 Sharpe ratio 對不同的投資組合排行時，兩種版本應該會產生相同的排行（其他條件皆相等的情況下）。

數值範例

再度採用靜態模型經濟結構 $\mathscr{M}^2$，則依然可以使用 Python 輕易說明 MVP 的基本概念。

投資組合統計

第一段是投資組合**預期報酬率**（*expected return*）的推導：

```
In [27]: rS = S1 / S0 - 1  ❶
         rS  ❶
Out[27]: array([ 1. , -0.5])

In [28]: rB = B1 / B0 - 1  ❷
```

```
        rB ❷
Out[28]: array([0.1, 0.1])

In [29]: def mu(rX):
             return np.dot(P, rX) ❸

In [30]: mu(rS) ❹
Out[30]: 0.25

In [31]: mu(rB) ❹
Out[31]: 0.10000000000000009

In [32]: rM = M1 / M0 - 1 ❺
         rM ❺
Out[32]: array([[ 1. ,  0.1],
                [-0.5,  0.1]])

In [33]: mu(rM) ❻
Out[33]: array([0.25, 0.1 ])
```

❶ 風險資產的報酬率向量。

❷ 無風險資產的報酬率向量。

❸ 預期報酬率函數。

❹ 交易資產的預期報酬率。

❺ 交易資產的報酬率矩陣。

❻ 預期報酬率向量。

第二段是變異數與波動率，以及共變異數矩陣：

```
In [34]: def var(rX):
             return ((rX - mu(rX)) ** 2).mean() ❶

In [35]: var(rS)
Out[35]: 0.5625

In [36]: var(rB)
Out[36]: 0.0

In [37]: def sigma(rX):
             return np.sqrt(var(rX)) ❷
```

```
In [38]: sigma(rS)
Out[38]: 0.75

In [39]: sigma(rB)
Out[39]: 0.0

In [40]: np.cov(rM.T, aweights=P, ddof=0)  ❸
Out[40]: array([[0.5625, 0.    ],
                [0.    , 0.    ]])
```

❶ 變異數函數。

❷ 波動率函數。

❸ 共變異數矩陣。

第三段是投資組合預期報酬率、投資組合預期變異數以及投資組合預期波動率，皆以同一個加權投資組合說明：

```
In [41]: phi = np.array((0.5, 0.5))

In [42]: def mu_phi(phi):
             return np.dot(phi, mu(rM))  ❶

In [43]: mu_phi(phi)
Out[43]: 0.17500000000000004

In [44]: def var_phi(phi):
             cv = np.cov(rM.T, aweights=P, ddof=0)
             return np.dot(phi, np.dot(cv, phi))  ❷

In [45]: var_phi(phi)
Out[45]: 0.140625

In [46]: def sigma_phi(phi):
             return var_phi(phi) ** 0.5  ❸

In [47]: sigma_phi(phi)
Out[47]: 0.375
```

❶ 投資組合與其報酬率。

❷ 投資組合預期變異數。

❸ 同資組合預期波動率。

投資機會集合

依據投資組合權重 ϕ 的 Monte Carlo 模擬，可以視覺化呈現波動率報酬率空間中投資機會集合（圖 3-1 是由下列程式碼片段產生的結果）。

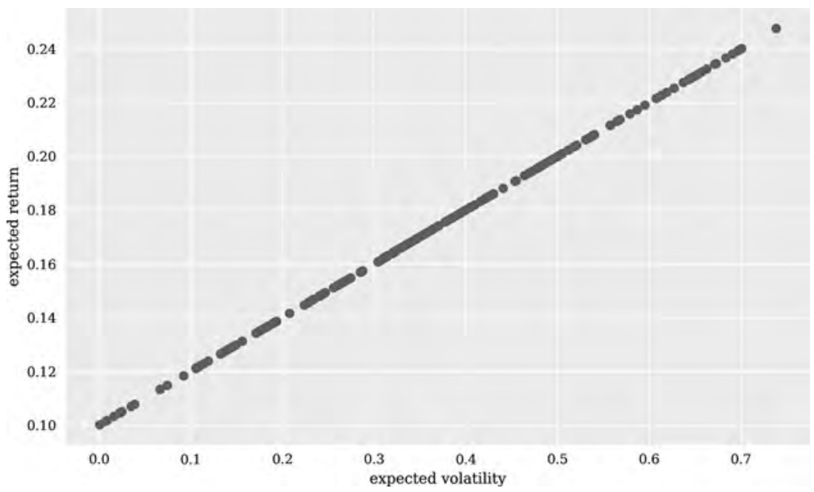

圖 3-1　預期投資組合波動率與報酬率的模擬（一種風險資產）

因為只有一種風險資產與一種無風險資產，所以機會集合為一條直線：

```
In [48]: from pylab import plt, mpl
         plt.style.use('seaborn')
         mpl.rcParams['savefig.dpi'] = 300
         mpl.rcParams['font.family'] = 'serif'

In [49]: phi_mcs = np.random.random((2, 200))  ❶

In [50]: phi_mcs = (phi_mcs / phi_mcs.sum(axis=0)).T  ❶

In [51]: mcs = np.array([(sigma_phi(phi), mu_phi(phi))
                         for phi in phi_mcs])  ❷

In [52]: plt.figure(figsize=(10, 6))
         plt.plot(mcs[:, 0], mcs[:, 1], 'ro')
         plt.xlabel('expected volatility')
         plt.ylabel('expected return');
```

❶ 隨機投資組合成分，正規化縮放至 1。

❷ 隨機成分的預期投資組合波動率與報酬率。

此時考量具有三個狀態的靜態經濟結構 $\mathscr{M}^3$，其中 $\Omega = \{u, m, d\}$。這三個狀態發生的可能性相同，即 $P = \left\{\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right\}$。可交易資產集合則由兩項風險資產 S 與 T 組成，其初始價格固定為 $S_0 = T_0 = 10$，而各自不確定的 payoff 分別如下所示：

$$S_1 = \begin{bmatrix} 20 \\ 10 \\ 5 \end{bmatrix}$$

與

$$T_1 = \begin{bmatrix} 1 \\ 12 \\ 13 \end{bmatrix}$$

依據上述的假設，下列 Python 程式碼再度以 Monte Carlo 模擬，並視覺化呈現結果（如圖 3-2）。結果是這兩項風險資產讓知名的 MVP「bullet」（彈頭）乍現。

```
In [53]: P = np.ones(3) / 3  ❶
         P  ❶
Out[53]: array([0.33333333, 0.33333333, 0.33333333])

In [54]: S1 = np.array((20, 10, 5))

In [55]: T0 = 10
         T1 = np.array((1, 12, 13))

In [56]: M0 = np.array((S0, T0))
         M0
Out[56]: array([10, 10])

In [57]: M1 = np.array((S1, T1)).T
         M1
Out[57]: array([[20,  1],
                [10, 12],
                [ 5, 13]])

In [58]: rM = M1 / M0 - 1
         rM
Out[58]: array([[ 1. , -0.9],
                [ 0. ,  0.2],
```

```
              [-0.5,  0.3]])

In [59]: mcs = np.array([(sigma_phi(phi), mu_phi(phi))
                         for phi in phi_mcs])

In [60]: plt.figure(figsize=(10, 6))
         plt.plot(mcs[:, 0], mcs[:, 1], 'ro')
         plt.xlabel('expected volatility')
         plt.ylabel('expected return');
```

❶ 針對三個狀態的新機率測度。

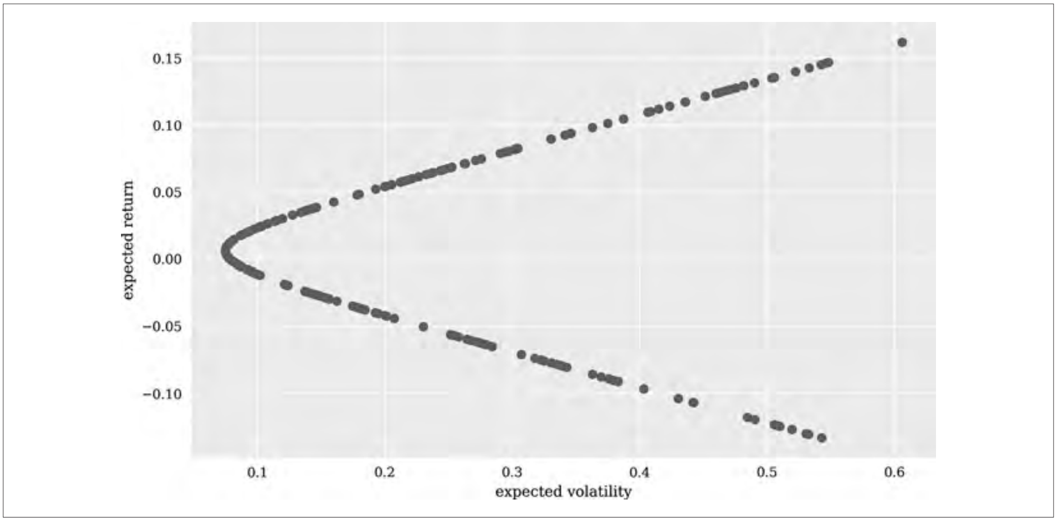

圖 3-2　預期投資組合波動率與報酬率的模擬（兩種風險資產）

最小波動率與最大 Sharpe ratio

接著要取得**最小波動率**（最小變異數）的投資組合與**最大** ***Sharpe ratio*** 的投資組合。圖 3-3 顯示此風險—報酬空間兩個投資組合對應所求的位置。

儘管風險資產 T 的預期報酬率為負值，但是在最大 Sharpe ratio 的投資組合中，有顯著的權重。這是因為（風險）分散效應，讓投資組合風險降得比投資組合的預期報酬率低：

```
In [61]: cons = {'type': 'eq', 'fun': lambda phi: np.sum(phi) - 1}

In [62]: bnds = ((0, 1), (0, 1))

In [63]: min_var = minimize(sigma_phi, (0.5, 0.5),
                            constraints=cons, bounds=bnds)  ❶

In [64]: min_var
Out[64]:      fun: 0.07481322946910632
              jac: array([0.07426564, 0.07528945])
          message: 'Optimization terminated successfully.'
             nfev: 17
              nit: 4
             njev: 4
           status: 0
          success: True
                x: array([0.46511697, 0.53488303])

In [65]: def sharpe(phi):
             return mu_phi(phi) / sigma_phi(phi)  ❷

In [66]: max_sharpe = minimize(lambda phi: -sharpe(phi), (0.5, 0.5),
                        constraints=cons, bounds=bnds)  ❸

In [67]: max_sharpe
Out[67]:      fun: -0.2721654098971811
              jac: array([ 0.00012054, -0.00024174])
          message: 'Optimization terminated successfully.'
             nfev: 38
              nit: 9
             njev: 9
           status: 0
          success: True
                x: array([0.66731116, 0.33268884])

In [68]: plt.figure(figsize=(10, 6))
         plt.plot(mcs[:, 0], mcs[:, 1], 'ro', ms=5)
         plt.plot(sigma_phi(min_var['x']), mu_phi(min_var['x']),
                  '^', ms=12.5, label='minimum volatility')
         plt.plot(sigma_phi(max_sharpe['x']), mu_phi(max_sharpe['x']),
                  'v', ms=12.5, label='maximum Sharpe ratio')
         plt.xlabel('expected volatility')
         plt.ylabel('expected return')
         plt.legend();
```

❶ 預期投資組合波動率最小化。

❷ 定義 Sharpe ratio 函數，假設短期利率為 `0`。

❸ Sharpe ratio 最大化（實際是其負值最小化）。

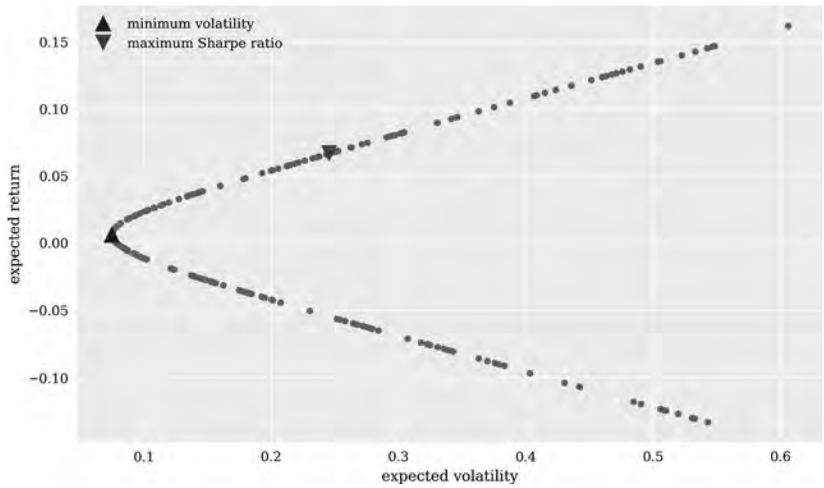

圖 3-3　最小波動率的投資組合與最大 Sharpe ratio 的投資組合

效率前緣

效率投資組合（*efficient portfolio*）是在已知預期風險（報酬率）之下具有最大（最小）預期報酬率（風險）。在圖 3-3 中，投資組合的預期報酬率低於最小風險投資組合者，這些投資組合都是**無效率**投資組合。下列程式碼可得出風險—報酬空間中效率投資組合，對應結果如圖 3-4 所示。效率投資組合整個集合稱為**效率前緣**（*efficient frontier*），而代理人只依據效率前緣選取某個投資組合：

```
In [69]: cons = [{'type': 'eq', 'fun': lambda phi: np.sum(phi) - 1},
                 {'type': 'eq', 'fun': lambda phi: mu_phi(phi) - target}]  ❶

In [70]: bnds = ((0, 1), (0, 1))

In [71]: targets = np.linspace(mu_phi(min_var['x']), 0.16)  ❷

In [72]: frontier = []
         for target in targets:
             phi_eff = minimize(sigma_phi, (0.5, 0.5),
```

```
                           constraints=cons, bounds=bnds)['x']  ❸
            frontier.append((sigma_phi(phi_eff), mu_phi(phi_eff)))
        frontier = np.array(frontier)

In [73]: plt.figure(figsize=(10, 6))
         plt.plot(frontier[:, 0], frontier[:, 1], 'mo', ms=5,
                  label='efficient frontier')
         plt.plot(sigma_phi(min_var['x']), mu_phi(min_var['x']),
                  '^', ms=12.5, label='minimum volatility')
         plt.plot(sigma_phi(max_sharpe['x']), mu_phi(max_sharpe['x']),
                  'v', ms=12.5, label='maximum Sharpe ratio')
         plt.xlabel('expected volatility')
         plt.ylabel('expected return')
         plt.legend();
```

❶ 針對預期報酬率以新限制來固定某個目標水準。

❷ 產生目標預期報酬率集合。

❸ 已知目標預期報酬率之下取得最小波動率的投資組合。

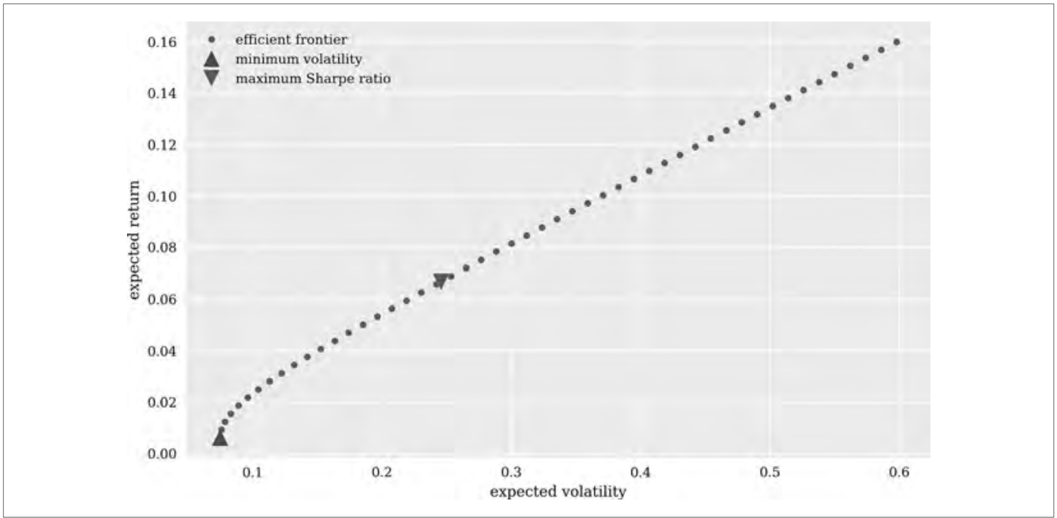

圖 3-4　效率前緣