

# 中華民國第 63 屆中小學科學展覽會

## 作品說明書

---

高中組 電腦與資訊學科

052509

股票年週期循環技術分析應用程式

學校名稱：國立鳳山高級中學

作者：  高二 鍾秉翰  高二 黃聖翔  高二 余悅	指導老師：  楊朝祺  許純瑋
--	-----------------------------

關鍵詞：多項式擬合、技術分析、應用程式

# 摘要

本研究係運用多項式擬合技術，表示出股價的年週期循環特性及趨勢，製作出可顯示個股股價年週期循環趨勢的應用程式。先觀察市面上常見的股票技術分析，接著用 Octave 進行研究，最後運用 Python 製作出包含使用者介面的應用程式，並將分析結果量化及輸出，本研究將此應用程式稱為「股價年週期循環技術分析應用程式」。

## 壹、前言

### 一、研究動機

關於股價的年週期循環，可能導致的原因有：淡旺季、資金、環境影響等等。所以最直接的方式就是針對股價進行分析，然而目前市場上相關的分析工具卻仍不普及，於是出於好奇與熱情，我們決定直接開發一個分析股價年週期循環的應用程式。

技術分析(Technical Analysis)是運用歷史價量與延伸數據的分析方式，然而綜觀目前幾十甚至幾百種技術分析，卻幾乎找不到可以有效率看出股價是否具年週期循環現象的方式，這使我們非常好奇此分析方式的可行性，所以我們便展開了這次跨領域的研究。

### 二、研究目的

建立一個應用程式。該應用程式可以運用多項式擬合技術，顯示出我國個股股價年週期循環趨勢，使投資型使用者可以藉此進行技術分析，研究型使用者可以藉此取得相關資料與數據。

### 三、文獻回顧

#### (一)軟體生命週期

根據 Nugroho et al.(2017)，軟體開發生命週期(Software Development Life Cycle，簡稱 SDLC)意指軟體從需求性開始，逐漸發展到成熟階段的過程。

SDLC 是為了讓開發軟體的時間成本降低，並且擁有高品質，SDLC 有一定的開發階段與流程，所有開發模型皆是從這樣的概念優化或延伸。其中比較常見的有分瀑布式、敏捷式與螺旋式等開發模型。

a Leader in Agile Training (2019)指出，V 型開發模型也在瀑布式的範疇裡面；迭

代、漸進、進化開發模型等是屬於敏捷開發模型。

a Leader in Agile Training (2019)也提到，瀑布式開發模型強調每個階段與過程，所以當已確定的階段要進行修改時，就會難以進行，開發彈性很低。螺旋式開發模型擁有風險評估，但如果已確定的階段要修改，仍會造成損失，且內容若需不斷評估，則需要花費許多時間成本。而我們編寫程式時，需要彼此反覆不斷的討論及確認彼此的想法，加上每個人對於程式的理解不一，因此將每一部份合在一起測試，恰好適合我們這次的研究，而這也就是迭代模型所擁有的特性，故我們選擇使用迭代模型。

## (二)與年週期循環類似概念

### 1.景氣循環

許繼元(2022)提及，經濟週期有擴張時期與緊縮時期，此現象稱為景氣循環，而隨著景氣循環而變化、易受經濟影響的股票稱為景氣循環股。一次景氣循環的週期通常長於一年，在週期長度上與本研究以年為週期不同，故本研究不會深入分析。

### 2.月分效應

黃益松等(2007)指出，月分效應指的是，證券市場中存在某單個月或一段期間的報酬，年復一年皆會高於其他月分或期間，可能原因為資金流動或政府政策，常見的月分效应有「元月效應」和「春節效應」。上櫃市場在 5%的顯著水準下，具元月效應、但不具春節效應；而上市市場在 5%的顯著水準下，具春節效應，卻不具元月效應。黃盈智(2016)研究 1995 年至 2014 年，指出元月效應與春節效應先後逐漸轉為不顯著，時間點與黃益松等(2007)並不違背。

月分效應與本研究所說的年週期循環現象，在研究對象上是不同的，月分效應不是針對個股，而是整個市場的特性；且月分效應只能顯示，公司外部年週期循環的因素對股價造成的影響，雖然本研究只會從股價分析而不會探討背後的原因，但卻能把公司內部年週期循環的因素對股價造成的影響納入分析。

## 貳、研究設備及器材

### 一、硬體

筆記型電腦

### 二、軟體

#### (一)程式語言 Python(3.10)

使用的套件及模組：

##### 1.網路爬蟲

- (1) `yfinance`：是第三方套件，可以爬取 Yahoo Finance API 的歷史資料，本研究主要用於取得國內各上市公司的歷史股價。
- (2) `requests`：是 Python 的外部函式庫，用途為網路爬蟲，本研究主要用於取得公司的名字。

##### 2.資料處理與儲存

- (1) `os`：是 Python 管理作業系統的相關模組，本研究主要用 `os` 模組來管理系統資料夾，以及查詢數據、圖片與分析結果的檔案是否已存在系統中，避免重複執行作業。
- (2) `json`：是 Python 管理 json 檔的模組，我們會將股票相關資訊與部分分析結果儲存於 json 檔中，方便程式間互相傳遞資料。
- (3) `csv`：是 Python 讀寫 csv 格式檔案的模組，本研究主要用 `csv` 檔儲存爬蟲下來的股價、平移後的股價。

##### 3.擬合、計算與繪圖

- (1) `numpy`：是 Python 的擴充程式庫，支援多維陣列與矩陣運算，也提供數學函數庫，本研究主要用 `numpy` 模組於多項式擬合。
- (2) `matplotlib`：是 Python 的一個繪圖庫，本研究用其中的 `pyplot` 模組繪製圖片，達到使分析結果視覺化的目的。
- (3) `math`：方便數值計算。

#### 4.使用者介面架設

- (1) `pyqt5`：是將 Qt 程式轉換成 Python 程式碼的圖形介面程式庫，能直接透過 Python 程式碼建立一個使用者視窗。
- (2) `sys`：是 Python 存放系統相關資訊的模組，程式在建立圖形介面時會用到外部的命令列參數。
- (3) `time`：適用於控制時間的模組，為了維持程式主線任務中，狀態欄不斷更新，以達到執行中感覺的目的。

#### 5.避免輸入除錯

- (1)`datetime`：用以取得現在年分，避免因使用者輸入錯誤導致程式無法執行。

#### (二)介面開發工具 Qt Designer

用於快速建立使用者介面。

#### (三)原始碼編輯器 Visual Studio Code

用於方便編輯 Python 程式碼。

#### (四)試算表軟體 Excel

用於處理股價歷史數據、儲存各種數據資料。

#### (五)數值分析軟體 Octave

Octave 是一種用於數值計算與繪圖的開源軟體，類似的軟體有：Matlab、Maplesoft、Wolfram Mathematica 等。我們在研究初期用 Octave 軟體進行多項式擬合、股價觀察、繪圖與疊圖分析。

#### (六)命令提示字元 CMD

用於安裝 Python 模組、把在 Qt Designer 設計出的 ui 檔轉換成 py 檔。

## 參、研究過程或方法

### 一、研究方式

本研究採取迭代式開發，原因是我們規劃約一個月重新討論一次大方向，且都是以口頭交流。下方不詳述所有迭代過程，而是以迭代中最重要的一次改變為界，也就是 Octave 做

法的結束，Python 做法的開始，詳細討論那次迭代改變的原因，以及該次迭代前發現應該討論的問題與變因。

## 二、需求分析

### (一)第一次名詞釋義

#### 1.年週期循環與年週期循環趨勢

在本研究中，年週期循環指的是，同一隻股票的價格，在每一年會出現類似趨勢的現象；而年週期循環趨勢，指的是個股的趨勢本身，在本研究中以一個多項式表示這個趨勢。

#### 2.預測

本研究中提到的預測，指的是股價的年週期循環趨勢，而非股價的精確值。

#### 3.擬合多項式技術分析

本研究的分析方式為一種技術分析，是利用歷史股價擬合後得出的擬合多項式，分析股價的年週期趨勢，是一種對未來股價的分析與預測方式。

#### 4.公司行為

本研究指的公司行為，侷限於公司以其名義，行使或履行增減資、股利發放與採取彈性面額，這三種會對股價造成影響的權利或義務。

#### 5.意外年

在隨機股票的隨機年分，會出現與其它年分明顯不同的走勢，造成此現象的原因可能為：公司行為、政府政策、天災人禍等等非年週期的因素，在本研究定義出現此現象的年分為意外年。意外年是有專一性的，會針對個股或受到相同因素影響的股票，不是針對所有股票。

### (二)觀察技術分析

技術分析是指將歷史的股價與成交量等資料，繪製成圖表或透過公式轉換為指標進行觀察，以達到預測未來趨勢的目的，主要的類別分為兩種：形態學、指標法。

#### 1.形態學(Chart Patterns)

單純觀看股票價量線圖，尋找是否出現通用規律。

## 2.指標法(Technical Indicators)

指標法是將歷史數據經過公式處理後，分析結果以預測未來股價的技術，而我們要做的分析方式是屬於指標法，常見的有：K 線、MA 線、MACD 指標、KD 指標、布林通道和 RSI 指標等，概念皆與本研究分析的方式不同。

## 3.觀察結果

(1)大多數技術分析都會以線圖呈現，以方便使用者觀察。

(2)雖然目前市面技術分析的運用已相當普及，在價格方面已有許多線圖與指標，但是關於從股價找出以年為週期循環的分析方式卻不常見，也缺乏相關分析的方法與方便視覺觀察的線圖。我們不禁對是否可以用年為週期的方式分析股價提出質疑，也許從產業鏈、季節供需或行業淡旺季的角度不難想像，部分股票的價格會以年為週期的方式循環，但至今以年為週期分析未來股價趨勢的方式卻尚不普及。

### (三)討論需求

#### 1.產出結果需求

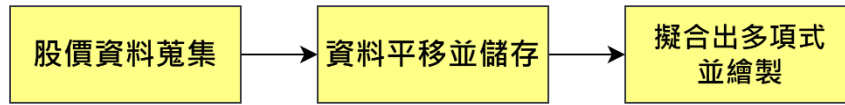
為了達到最初的研究目的，我們需要將多年的歷史股價資料擬合成一個多項式，但不是直接擬合，而是將多年的資料以年首年尾斷開，將平移後的收盤價先多年全部疊合再擬合，得出目標的擬合多項式，接著根據上文觀察技術分析的觀察結果(1)，我們需要將擬合出的多項式，繪製並產出方便使用者觀察的圖片，最後將此投資分析結果進行回溯測試(Backtesting)，用歷史資料模擬預測未來趨勢，並分析結果。

#### 2.應用程式需求

為了達到最初的研究目的，我們建立的程式要讓使用者可以直接接觸到使用者介面，在上面輸入完欲分析股票的資訊後，程式要能顯示產出結果需求。此時我們將最初的程式架構分成：股價資料蒐集、資料平移並儲存以及擬合出多項式並繪製，三階段討論，如圖 1。

圖 1

### 設計流程 1



股價資料蒐集階段，我們為了取得使用者欲分析股票的歷史收盤價資料，需要一個可以連接網上公開資料庫 API 的爬蟲程式；在資料平移並儲存階段，平移的方式是將每一年的每天資料分別減去該年股價的算術平均數，目的是平移後數據將比較適合疊圖分析，而儲存資料是為了避免下次分析相同股票時要再次蒐集資料，因此以檔案形式將資料儲存於使用者的電腦資料夾中；在擬合出多項式並繪製階段，我們需要一個有處理數值功能，並且可以繪製線圖的程式或軟體。

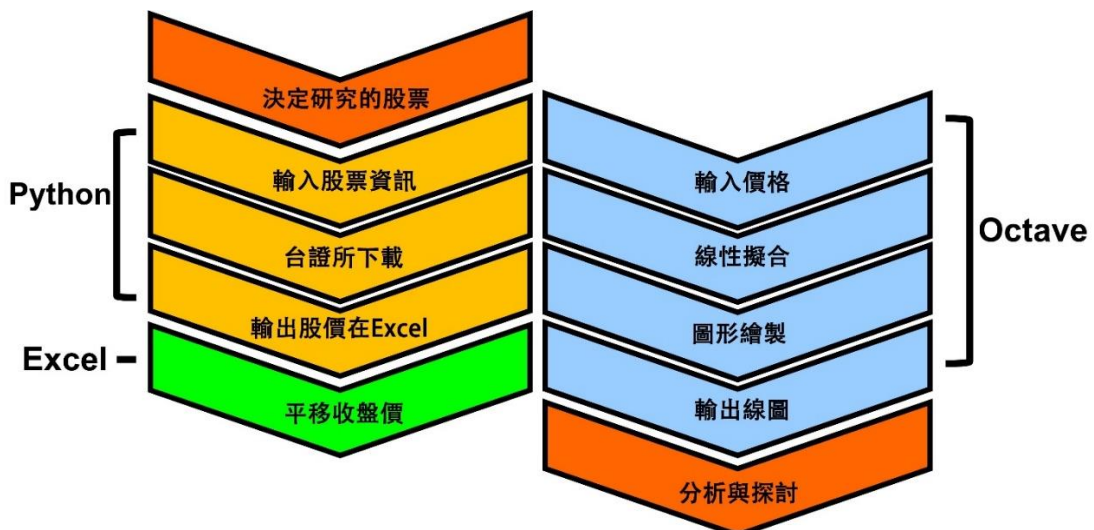
### 三、Octave 做法

#### (一)設計最初研究架構

根據上文需求分析的應用程式需求，我們設計出以下關於 Octave 做法的分析步驟。Octave 做法的研究流程如圖 2：橘色部分是運用 Python 寫成的爬蟲程式，主要用途是連接台灣證券交易所 API 下載歷史股價；接著將下載後的股價在 Excel 平移並儲存；最後手動複製平移後的價格到 Octave 進行多項式擬合，並繪製出方便視覺觀察與分析的線圖。

圖 2

#### Octave 做法的研究架構





Octave 是一個數值分析軟體，選用它的原因有：它符合擬合及繪圖的需求、免費、開源而且網路學習資源豐富。雖然 Octave 做法不是本研究最後呈現的方式，但這卻是我們開發出最後應用程式不可或缺的基石，因為我們從中觀察到許多值得討論的變因。

## (二)研究對象、方式及原因

我們統一分析在我國上市各類股，市值前十大的股票。原因是為了避免只研究少數在我國發展較為興盛的類股，導致研究不夠全面；而上市且市值大的公司，趨勢和波動更穩定，較不會因為小資金的流動而有很明顯的變化，炒作股票的門檻也相對較高。

分析方式為，先將 2011 年到 2018 年八年的股價每年分別平移，接著將八年相同交易日的平移後價格取算術平均數，然後再擬合成一個三次多項式，並與 2019 單年平移後股價擬合成的三次多項式疊圖，最後將兩個三次多項式繪製成一張線圖。選擇擬合 2011 年到 2018 年的原因是，台灣證券交易所的資料限制以及配合 2019 年；選擇 2019 年為模擬預測的年分的原因是，2019 年年底 COVID-19 疫情爆發，改變許多產業的原貌與結構，所以我們推測若加入 2019 年之後的年分可能會導致預測較不準確；而選擇擬合三次多項式的原因是，我們先假設在一年中，股價會有一次主要的漲幅與跌幅，所以三次應該是最符合大多數股價波動的擬合多項式次數。

## (三)研究分析

我們將 2011 年到 2018 年平移後擬合，模擬預測 2019 年的股價趨勢，並用肉眼觀察作為判斷是否有規律的依據。我們經過分析與整理後，將它們分成線圖貼近、相同起伏、部分起伏符合、毫不相干和異常五類。

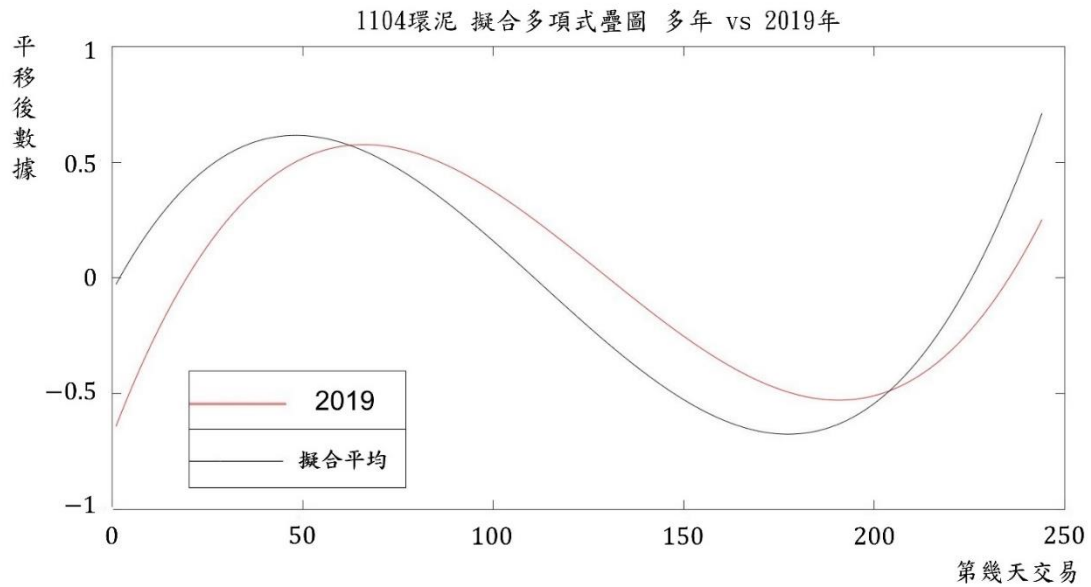
### 1.線圖分類

#### (1)有規律線圖

a.線圖貼近：如圖 3 中水泥股 1104 環泥。

圖 3

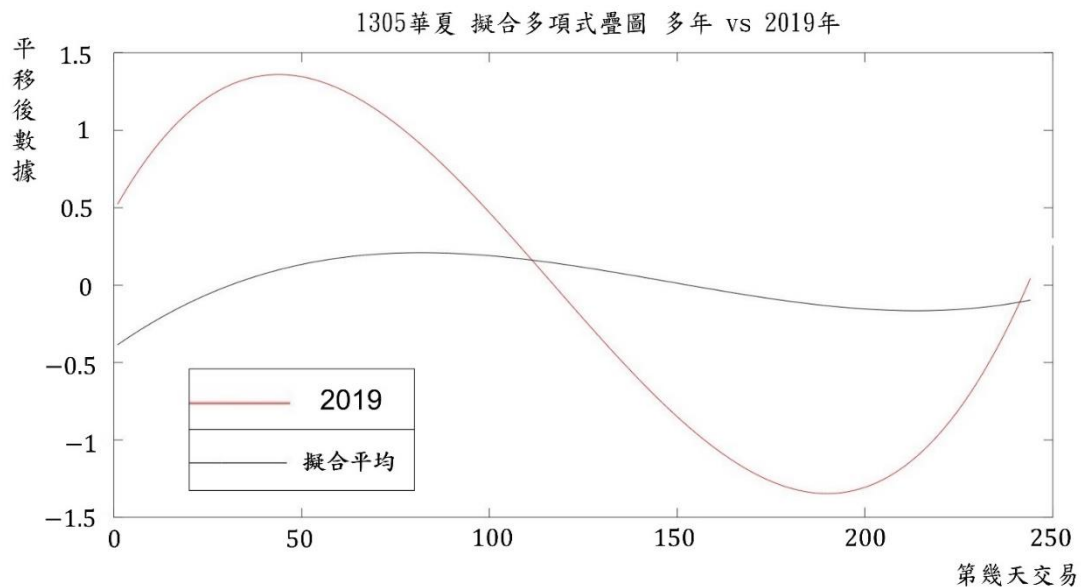
環泥的多年擬合多項式和 2019 年擬合多項式疊圖結果



b.相同起伏規律：如圖 4 中塑膠股 1305 華夏，雖起伏並未大到符合 2019 年之線圖，但起伏方向依然相同，因此也視為有規律的線圖。

圖 4

華夏的多年擬合多項式和 2019 年擬合多項式疊圖結果



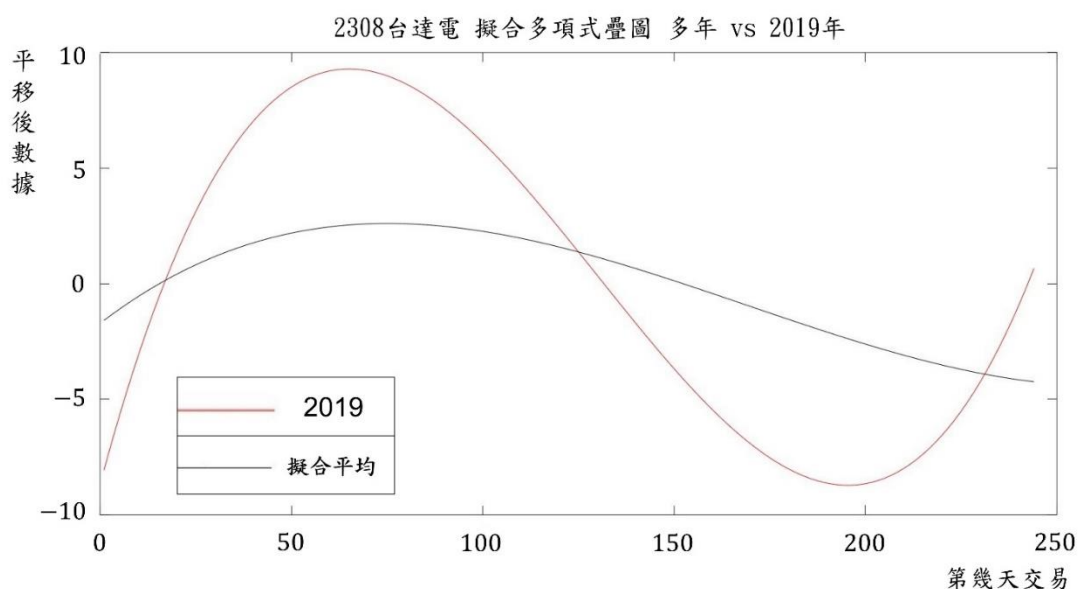
(2)不符合規律之線圖

a.部分起伏規律符合：如圖 5 中電子組件股 2308 台達電的疊圖結果可以

看到，投資者可能會根據預測結果判斷，2019 年下半年的價格緩慢下跌、年末不會有回升，而造成交易上的失誤。

圖 5

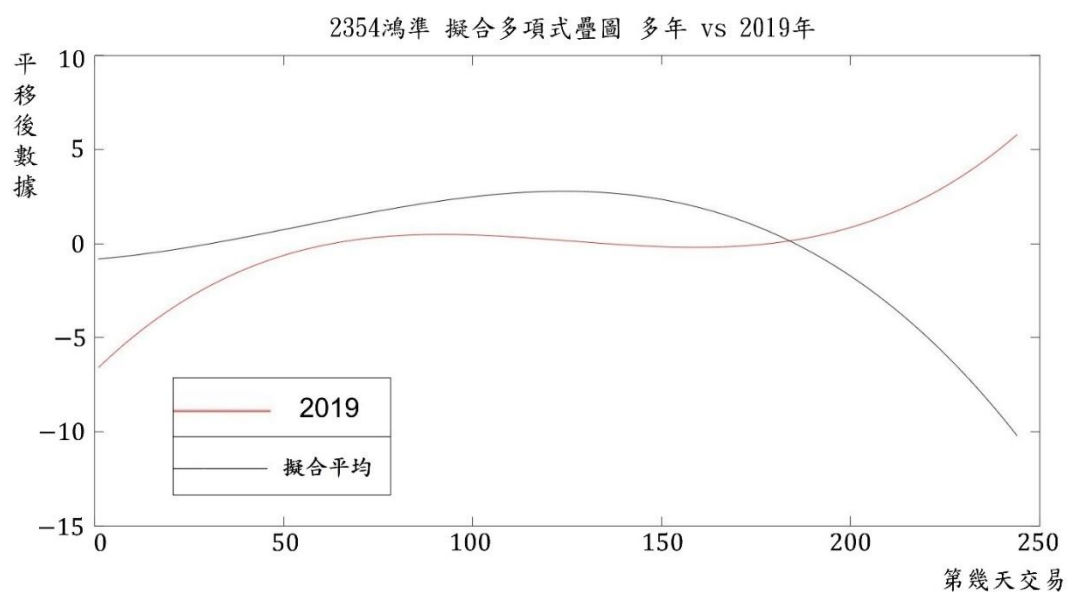
台達電多年擬合多項式和 2019 年擬合多項式疊圖



b.毫不相干：如圖 6 中電子股 2354 鴻準的疊圖結果可以看到，預測錯誤。

圖 6

鴻準多年擬合多項式和 2019 年擬合多項式疊圖



### (3)異常

違反或無法滿足我們對研究對象設定的條件，主要為兩種原因：2011年尚未上市、公司行為頻繁或影響巨大，這兩個導致異常的原因在後方討論變因時都會提到。

#### 2.各類股符合情形

表 1

Octave 做法中各類股市值前十大公司之股價分析結果

各類股分布	線圖貼近	相同起伏	部分起伏	毫不相干	異常	符合規律
營建股	3	2	1	3	2	5
塑膠股	1	4	1	4	0	5
電機股	3	1	1	4	1	4
貿易股	2	1	0	5	2	3
汽車股	1	2	0	5	2	3
鋼鐵股	1	2	0	5	2	3
半導體股	0	3	0	4	3	3
電腦股	2	0	4	3	1	2
食品股	2	0	3	5	0	2
光電股	2	0	2	4	2	2
油電股	2	0	1	5	0	2
網通股	1	1	2	5	1	2
水泥股	1	0	3	3	0	1
金融股	0	1	4	4	1	1
電子組件股	0	1	1	4	4	1
航運股	0	0	3	5	2	0
電子股	0	0	2	6	2	0

表 1 中可以發現，大部分類股都是部分有規律、部分無規律。基於我們研究的是股票的技術分析方式，使用者不可能也不應該單靠此分析結果決定交易行為，並且任一種技術分析方式也不會適用於所有股票，而是要根據該股票本身的各種性質而定，所以我們判斷此種分析方式是可行的，投資者在交易部分股票的時候，可以透過先觀察該股票的擬合多項式，了解未來價格可能的股價趨勢。

### (四)討論需求與變因

從這次的模擬預測結果我們可以看出，部分股票的價格具有年週期循環的特性，然

而這還不足以滿足上文需求分析的應用程式需求與產出結果需求。此外，在這次 Octave 做法的研究中，我們觀察到一些需要討論的變因。

## 1. 討論需求

### (1) 應用程式需求的股價資料蒐集階段

我們缺少一個方便使用者輸入、明確顯示最後分析結果的使用者介面。

### (2) 應用程式需求的資料平移並儲存階段

在 Octave 做法中，我們都是將下載來的股價以手動的方式平移，程式並不會自動完成。

### (3) 應用程式需求的擬合出多項式並繪製階段

在輸入股價數據到 Octave 時，我們都是以手動複製輸入，並且每次都要手動輸入 Octave 中的函數處理數據、擬合出多項式和繪製圖片，最後繪出的圖片我們也是手動截圖儲存。這些非自動化的分析步驟，增加了分析的使用門檻，因此我們認為，Octave 做法對於使用者並不是一個友善的方式。

### (4) 產出結果需求

Octave 做法只會產出一張線圖，所以我們只能以肉眼觀察分析，雖然這樣已經可以當作對投資有利的參考或建議，但我們卻無法量化擬合出多項式的程度，也無法將這種投資分析方式進行回溯測試，模擬實際運用在市場的效果。

### (5) 對照組

如果要進行回溯測試，我們就需要一個目前市面上廣泛被使用的投資方式，用同樣的資金投資，並比較兩種投資方式在報酬率上的差異。這種投資方式不是拿報酬率跟大盤或其它金融商品比較，因為那只會找出預測那幾年漲幅大的股票，並無法證明此投資方式是否適合該股票。

## 2. 討論變因

### (1) 擬合次數

本次 Octave 做法僅擬合三次的多項式，是因為我們先假設股價在一年間各會有一個漲幅與一個跌幅，但是我們並不能確定三次多項式一定是最適合的

擬合多項式次數，因此如果能改變擬合多項式的次數，譬如提高多項式次數，可能可以增加擬合程度，但根據龍格現象(Runge's phenomenon)，高次多項式卻未必擁有較好的擬合結果，因此需要找到該股票適合的擬合多項式次數。

### (2)擬合程度量化

本次 Octave 做法，由於無法將擬合程度量化，導致我們不能直接分析 2011 年到 2018 年擬合多項式與 2019 年的關係，所以我們只好先將 2019 年也擬合成多項式，與預測函數疊圖後用肉眼觀察，然而即使如此，擬合程度無法量化仍是需解決的問題。

### (3)擬合多年

本次 Octave 做法皆是用 2011 年到 2018 年預測結果，但是根據短期、中期或長期的投資者，可能會有不同的需求，譬如長期投資者也許希望擬合 2000 年到 2020 年的結果，因為他們看重的是長期平均的趨勢更勝波動；而短期投資者可能更希望擬合 2010 年到 2020 年就好，因為他們想掌握更精準的波動，即使那可能會增加預測失誤的風險。

### (4)預測多年

本次 Octave 做法是假設 2019 年為未知，而我們探討透過擬合 2011 年到 2018 年的多項式，模擬預測 2019 年的可能性，但如果要確定預測函數的預測準度，須蒐集預測更多年的獨立結果，才能更確定預測準度，更別提有個假設是，如果 2019 年本身為意外年，那本次預測結果將完全失去意義。

### (5)意外年

在分析擬合結果的時候，發現有意外年的存在，此現象改變了該年本有的走勢，導致預測結果可能會因此受到影響，所以若想找到更準確的擬合多項式，就需要將這些發生非年週期因素年分的價格排除，或做特殊的處理。

### (6)244 天

我們發現台股每年的交易日數量不一，可能導致擬合時出現錯誤，而近十年交易日總數都落在 241 到 248 天，所以本研究統一將所有年分都設為約在中

點的 244 天。若不足 244 天，則重複最後一天的資料補滿；若超過 244 天，則只取前 244 天。

#### (7)歷史資料來源

我們發現不同歷史股價來源會有不同的限制。我們主要考量的有以下幾點，依重要程度排序為：資料是否有公信力、是否處理所有公司行為造成的價格失真、歷史資料越完整越好、是否有成熟穩定的下載途徑、是否有上櫃與櫃資料、下載時間與次數限制等。如表 2，最後我們挑選出「Yahoo 財經」、「台灣證券交易所」和「FinMind」三個資料庫，而再經過表 2 的分析，最後決定選用「Yahoo 財經」資料庫中的「經調整收盤價」，作為本研究的歷史股價資料來源。

表 2

#### 股價歷史資料來源

股價歷史資料來源	增減資修正	彈性面額修正	權值還原	最早資料	上市	上櫃與櫃
台證所收盤價	無	無	無	2010	有	無
Yahoo 收盤價	有	有	無	2000	有	無
Yahoo 經調整收盤價	有	有	有	2000	有	無
FinMind 收盤價	無	無	無	1992	有	有
FinMind 權值還原收盤價	無	無	有	1992	有	有

#### 四、Python 做法

##### (一)第二次名詞釋義

##### 1.模擬分析與真實分析

在本研究中模擬分析與真實分析，都是擬合多項式技術分析的一部分，在進行擬合多項式技術分析時，應先模擬分析，再真實分析。

##### 2.模擬擬合年與真實擬合年

擬合年在本研究，係指用於擬合出擬合多項式的所有年分，可以是單年但通常是多年，與回測年係指相同年分集合，但擬合年強調的是，這些年用於擬合出擬合

多項式時的用途。在模擬分析中稱模擬擬合年；在真實分析中稱真實擬合年。擬合年的第一年稱擬合起始年，擬合年的最後一年稱擬合結束年。

### 3. 模擬當年與真實當年

當年在本文中，係指假設分析或分析當下的年分，只能是單年。在模擬分析中，會假設分析的年分為過去的某一年，則稱該年為模擬當年；在真實分析中，分析當下的年分，稱為真實當年，而當年為哪一年，也決定了使用者現在使用的是模擬分析還是真實分析。當年減一年為擬合結束年，當年同時也是預測起始年。

### 4. 模擬回測年與真實回測年

回測年在本文中，係指用於擬合出擬合多項式的所有年分，可以是單年但通常是多年，與擬合年係指相同年分集合，但回測年強調的是，將擬合多項式投資套用在擬合年的回測效果，而這個效果將以累積報酬率的形式進行比較，求得最適次數。在模擬分析中稱模擬回測年；在真實分析中稱真實回測年。

### 5. 模擬預測年與真實預測年

預測年在本文中，係指將擬合多項式投資，用於預測當年(包括)之後的年分，可以是單年但通常是多年。在模擬分析中，預測年稱模擬預測年，用於測試擬合多項式投資，用於這隻股票可能的報酬率結果，預測結束年最晚為當下這年減一年；在真實分析中，因為沒有當年(包括)之後年分的完整股價資料，所以在真實分析中，真實預測年無法計算報酬率。預測年的第一年為當年。

### 6. 手動踢除年

指使用者想要將意外年從擬合年中踢除，而踢除原因不是因為公司行為時，那些被踢除的年分稱手動踢除年。

### 7. 報酬率

本研究的報酬率分成三種，分別是累積報酬率(Return of Investment，簡稱 ROI)、年化報酬率(Compound Annual Growth Rate，簡稱 CAGR)和內部報酬率(Internal Rate of Return，簡稱 IRR)。



## 8. n~m 次最適次數

在本研究指的是，在其他條件都固定的前提下，比較 n 次到 m 次擬合多項式，運用擬合多項式投資，回測年報酬率最高的擬合多項式次數。我們將這個次數定義為最適次數或 n 到 m 次最適次數。

## 9. 峰值與谷值

最適次數擬合多項式中，自變數在 1 到 244 的範圍中，當斜率由正變負時，取小於該自變數的最大整數自變數，稱為峰值；當斜率由負變正時，取小於該自變數的最大整數自變數，稱之谷值。特殊情況，當 244 本身不是峰值也不是谷值時，若當年初斜率為正，且年末斜率為負時，244 為谷值；若當年初斜率為負，且年末斜率為正時，244 為峰值。

## 10. 擬合多項式投資

在本研究的定義為，程式運用擬合出的擬合多項式，進行峰值買谷值賣的模擬交易行為。在研究結果將對過程有更詳細的描述。

廣義的擬合多項式投資，是對 n~m 次都進行模擬交易，目的是透過回測年報酬率比較求出最適次數；狹義的擬合多項式投資，是只針對最適次數進行模擬交易，目的是描述最終顯示出的只有最適次數的相關分析結果，但其實狹義的相關分析結果在廣義時就已經求出了。若沒有特別說明，皆指廣義。

## 11. 定期定額投資

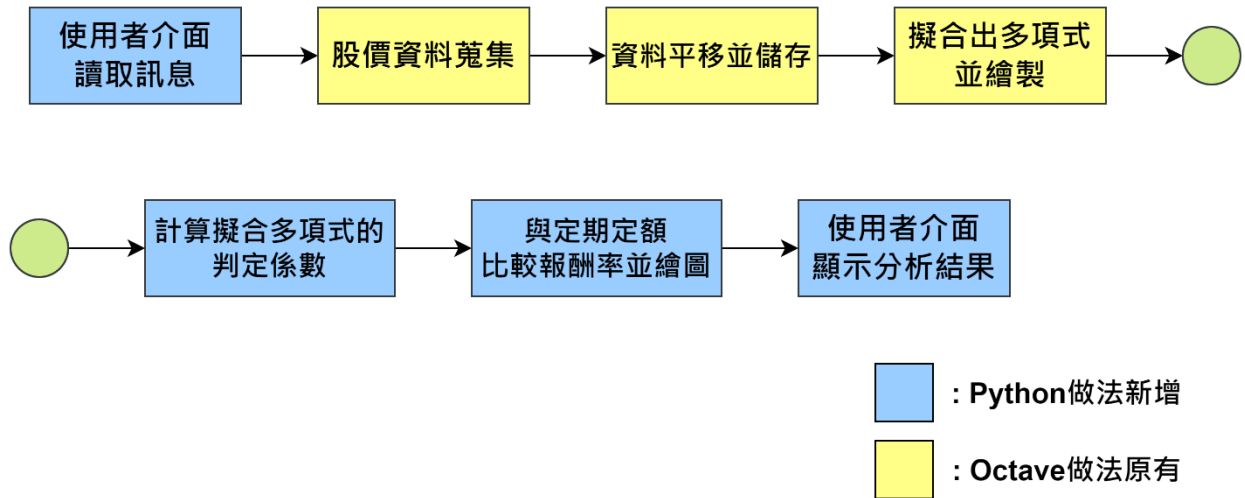
在本研究中是指，運用定期定額的方式投資，在每月月初用固定金額購買股票，不滿一股的金額則併入下個月月初金額，為最單純的定期定額方式，每月投入金額不因股價而變動，且不會因觸擊停損點或其他任何原因而賣出股票。

### (二) 設計新版研究架構

根據上方 Octave 做法的討論需求，我們把原先的三個階段擴充成七個階段，如圖 7：使用者介面讀取訊息、股價資料蒐集、資料平移並儲存、擬合出多項式並繪圖、計算擬合多項式的判定係數、與定期定額比較報酬率並繪圖、將產出結果需求顯示在使用者介面。

圖 7

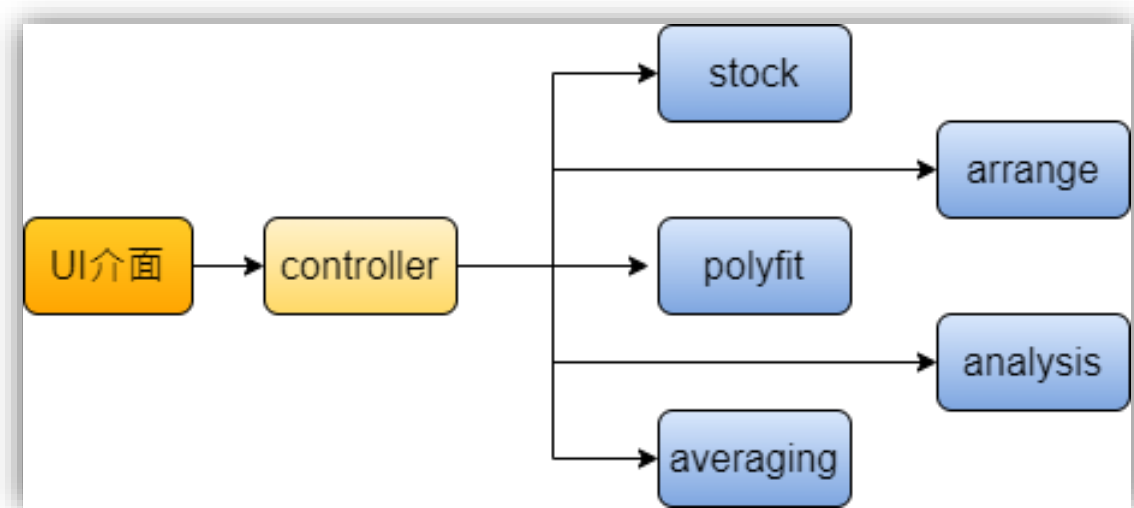
設計流程 2



並且根據上方 Octave 做法的討論需求與變因，整個程式已變得相當複雜，而為了使此程式能自動全程執行，我們決定將全部階段都用程式語言 Python 執行，接著再根據上方提到的七個階段，我們設計出下方圖 8 的程式架構。

圖 8

設計的程式架構



(三)開發

1.使用者介面相關

start1、controller1 和 ui1 管理使用者介面 1；start2、controller2 和 ui2 管理使用

者介面 2。

## 2.center

目的與用途為，統整執行相關程式，並統整回傳分析結果，導至 controller2。

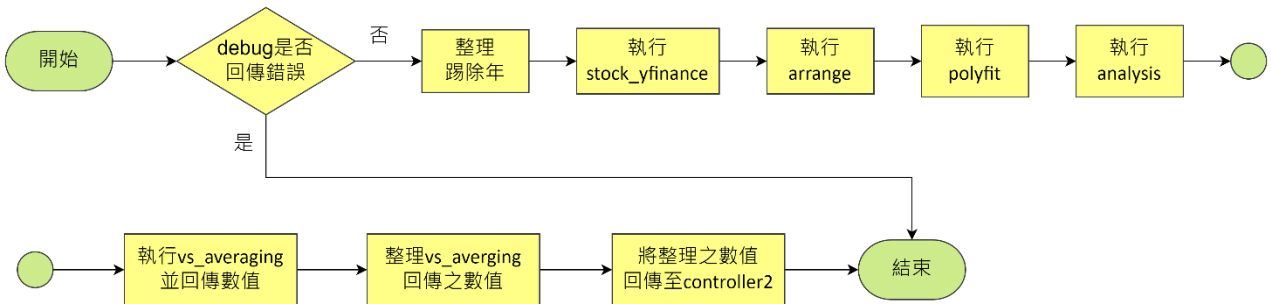
所需資料：

credibility/分析資料 json 檔(from analysis)，取判定係數傳至 controller2。

流程圖如圖 9。

圖 9

### center 流程圖



## 3.執行相關

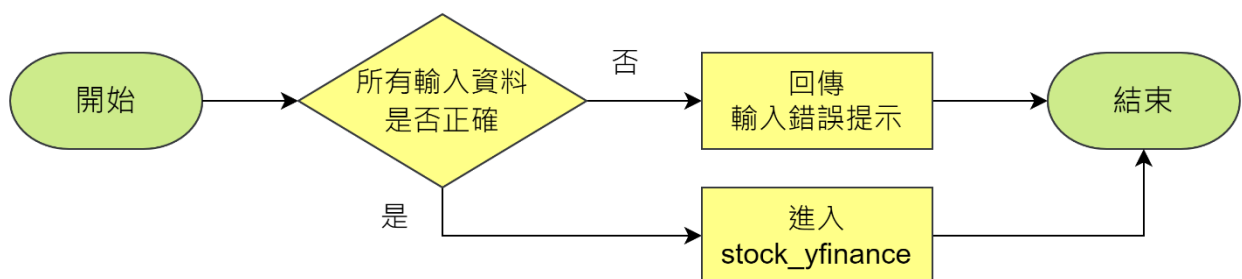
### (1)debug

目的與用途為，判定介面中所輸入之資料是否正確。

流程圖如圖 10。

圖 10

### debug 流程圖



## (2)stock\_yfinance

目的與用途為，下載調整後歷史股票資料。

創建資料：

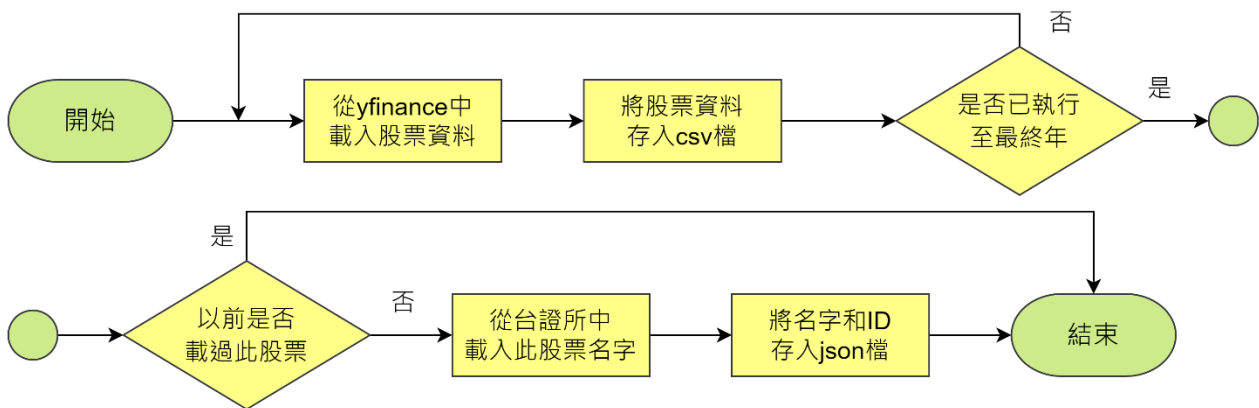
a.original\_data/原資料 csv 檔。

b.stock\_name/name&ID json 檔。

流程圖如圖 11。

圖 11

stock\_yfinance 流程圖



創資料:

1.original\_data:原資料檔(csv)

2.stock\_name:name&ID(json)

## (3)arrange

目的與用途為，將股票資料平移。

創建資料：

data/整理後資料 csv 檔。

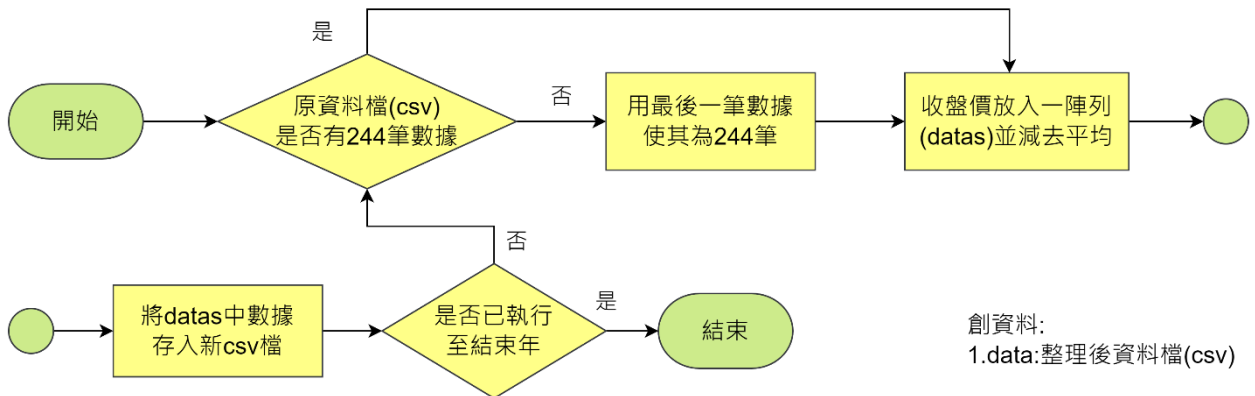
所需資料：

original\_data/原資料 csv 檔(from stock\_yfinance)。

流程圖如圖 12。

圖 12

arrange 流程圖



(4)polyfit

目的與用途為，將整理後股票資料擬合並繪圖。

創建資料：

- a.coef/擬合多項式資料 json 檔。
- b.picture/擬合多項式疊圖 png 檔。

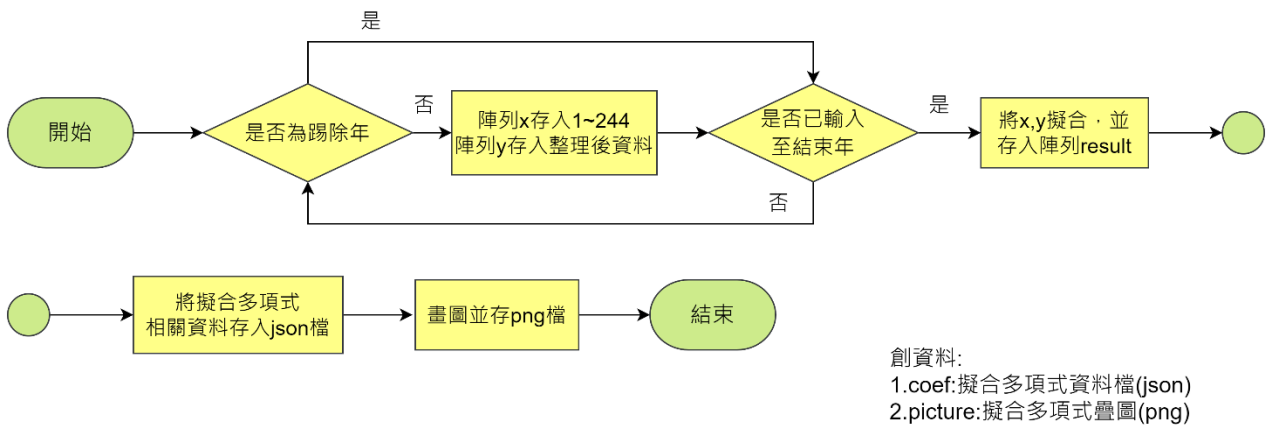
所需資料：

- a.stock\_name/name&ID json 檔(from stock\_yfinance)。
- b.data/整理後資料 csv 檔(from arrange)。

流程圖如圖 13。

圖 13

polyfit 流程圖



### (5)analysis

目的與用途為，計算擬合多項式之判定係數。

創建資料：

credibility/分析資料 json 檔。

所需資料：

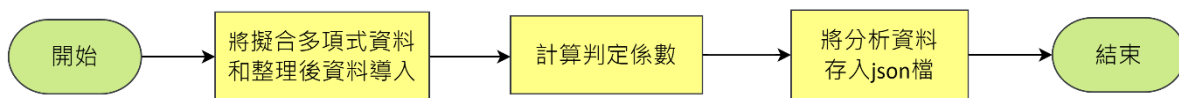
a.data/整理後資料 csv 檔(from arrange)。

b.coef/擬合多項式資料 json 檔(from polyfit)。

流程圖如圖 14。

圖 14

#### analysis 流程圖



創資料:  
1.credibility:分析資料檔(json)

### (6)vs\_averaging1

目的與用途為，用於模擬分析，計算並比較定期定額和多項式投資方法的總資產和報酬率。

創建資料：

a.vs\_picture/總資產比較圖 png 檔。

b.vs\_picture/累積報酬率比較圖 png 檔。

所需資料：

a.original\_data/原資料 csv 檔(from stock\_yfinance)。

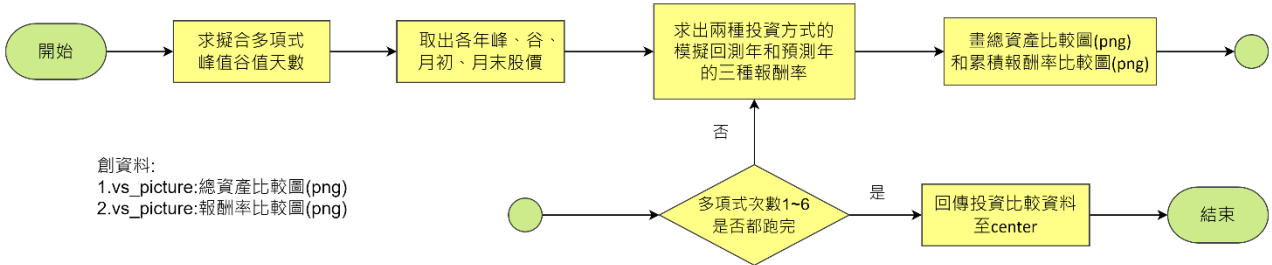
b.stock\_name/name&ID json 檔(from stock\_yfinance)。

c.coef/擬合多項式資料 json 檔(from polyfit)。

流程圖如圖 15。

圖 15

vs\_averging1 流程圖



(7)vs\_averging2

目的與用途為，用於真實分析，計算並比較定期定額和多項式投資方法的總資產和報酬率。

創建資料：

- a.vs\_picture/總資產比較圖 png 檔。
- b.vs\_picture/累積報酬率比較圖 png 檔。

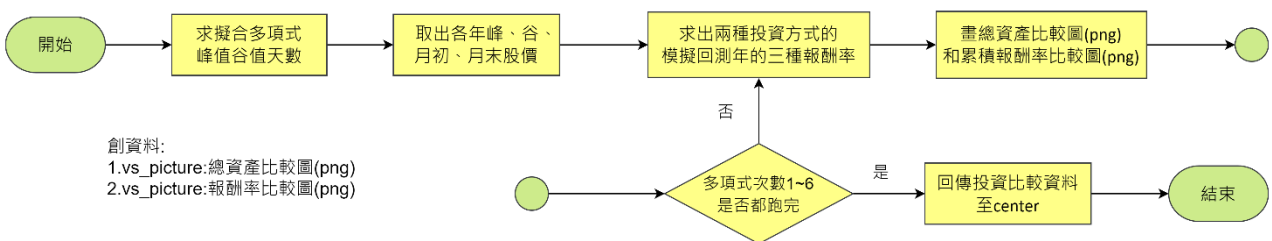
所需資料：

- a.original\_data/原資料 csv 檔(from stock\_yfinance)。
- b.stock\_name/name&ID json 檔(from stock\_yfinance)。
- c.coef/擬合多項式資料 json 檔(from polyfit)。

流程圖如圖 16。

圖 16

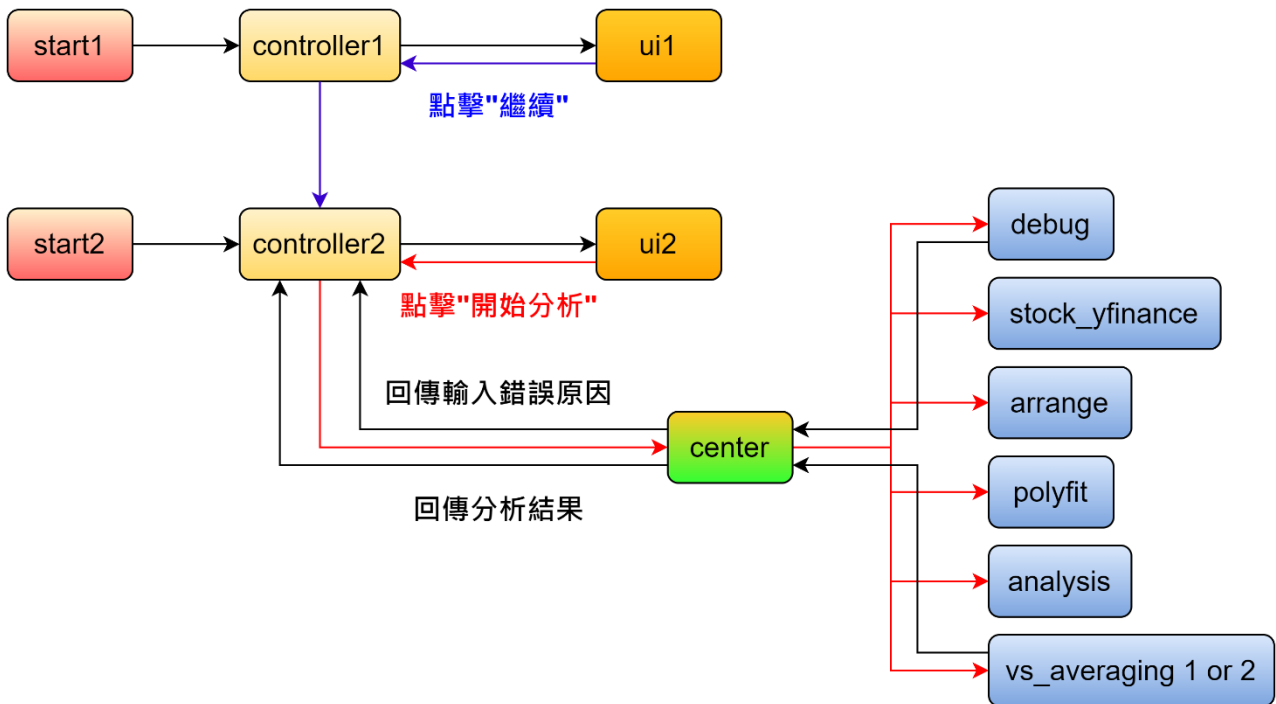
vs\_averging2 流程圖



#### (四)開發結果

圖 17

##### 最終的程式架構



#### 五、部署與維護

##### (一)定義角色

###### 1.使用者

使用者指使用本系統之不特定瀏覽民眾，這個角色主要可以進行系統之功能。

###### 2.維護者

此角色用於程式程序設計與系統維護管理，這個角色可以接收使用者回饋進而修改程式與偵錯。

##### (二)部署

###### 1.存放

我們將最終的「股價年週期循環技術分析應用程式」儲存於 Google 雲端硬碟，並開放存取權為「知道連結的任何人」，目的是有個公開平台，可以讓使用者自由下載。



## 2.接收回饋

我們建立一個 Google 表單，使用者可以在裡面輸入建議、報錯或給予回饋。

### (三)維護

維護者根據 Google 表單的建議或報錯，不定期修正或優化此應用程式，並上傳更新版本至 Google 雲端硬碟。

## 肆、研究結果

### 一、擬合多項式投資

運用我們所開發出的「股價年週期循環技術分析應用程式」，先求出各次數的峰值與谷值；接著在谷值用所有現金購入股票，並且只要尚未達到下個峰值，有資金就立刻購入股票，直到達到峰值當天，將所有股票賣出，並且持續持有現金，等到下次谷值再用所有現金購入股票；重複以上行為，所形成的投資方式。目的為求出最適次數並顯示其相關結果。

### 二、分析介紹

此分析方式分成兩個階段，分別是「模擬分析」以及「真實分析」。

#### (一)模擬分析

模擬分析的目的是，讓使用者假設自己在模擬當年運用此擬合多項式技術分析，並將此分析結果運用在實際市場，也就用狹義擬合多項式投資，從模擬當年直到模擬結束年的結果，使用者可以根據不同模擬擬合起始年、模擬當年與模擬預測結束年的相關分析結果(最適次數、判定係數、峰值、谷值和三種報酬率) 與線圖，評估擬合多項式技術分析是否適合這支股票，如果判斷適合，再進行真實分析。

#### (二)真實分析

當使用者已經評估擬合多項式技術分析適合這支股票，就會進行真實分析。在真實分析時，真實當年應是使用者分析當下的年分，而真實欲測年無完整資料可以分析，在此分析系統中是無意義的，所以也統一輸入使用者分析當下的年分。

### 三、下載

存放股價年週期循環技術分析應用程式的連結請見附錄。

#### 四、操作說明

(一)請點擊程式 start1 開始執行本「股價年週期循環技術分析應用程式」。

(二)稍後片刻，螢幕會顯示出使用說明，閱讀完畢請點擊「繼續」。

(三)稍後片刻，螢幕會顯示出分析股票的介面，如圖 18。

圖 18

#### 使用者介面 2

輸入股票代碼    
\*台灣上市股票

輸入擬合起始年   
\*請注意該股票最低起始年非上市年分

輸入當年年分   
\*同時為預測起始年  
\*擬合結束年為前一年

輸入預測結束年   
\*若當年和預測結束年皆為現年，為真實分析  
\*若非真實分析，請填入現年之前的年分

輸入踢除年   
\*請輸入非增減資、股利、彈性面額的意外年  
\*請輸入擬合有包括的年份  
\*若一次輸入多個年份，請以空白鍵分隔  
\*若無請輸入“0”

分析結果1:  
\*擬合多項式1-6次中，投報率最高的多項式

多項式次數:  
判定係數:  
多項式峰值:  
多項式谷值:

分析結果2:  
\*與定期定額的報酬率比較  
\*擬合多項式 VS 定期定額  
\*回測年:從擬合開始年到擬合結束年  
\*預測年:從預測開始年到預測結束年

模擬/真實回測年累積報酬率:  
模擬/真實預測年累積報酬率:  
模擬/真實回測年年化報酬率:  
模擬/真實預測年年化報酬率:  
模擬/真實回測年內部報酬率:  
模擬/真實預測年內部報酬率:

狀態欄:請輸入股票相關資料

若對於本應用程式有任何建議及問題，請掃描QRCode填寫表單讓我們知道!

將顯示投報率最多項式的擬合結果圖

將顯示與定期定額報酬率比較圖

(四)請根據介面要求，輸入分析股票的股票代號，以及測試的擬合起始年、當年、預測結束年與踢除年分(輸入的詳細介紹與限制，請見 ui1 的使用說明)。

(五)輸入完畢請點擊「開始分析」按鈕，左下方狀態欄若顯示「運作中，請稍後...」，表示程式已經在執行了。

(六)此時需要注意，左下方狀態欄的上方是否有出現「輸入錯誤」的提醒，如果有，將會伴隨一句加底線的錯誤原因，如圖 19，請直接根據錯誤原因更正輸入的資料，然後重複(四)、(五)步驟。如果沒有出現「輸入錯誤」的提醒，請確保狀態欄是顯示「運作中，請稍後...」，並靜待程式分析完成。

圖 19

使用者介面顯示輸入錯誤



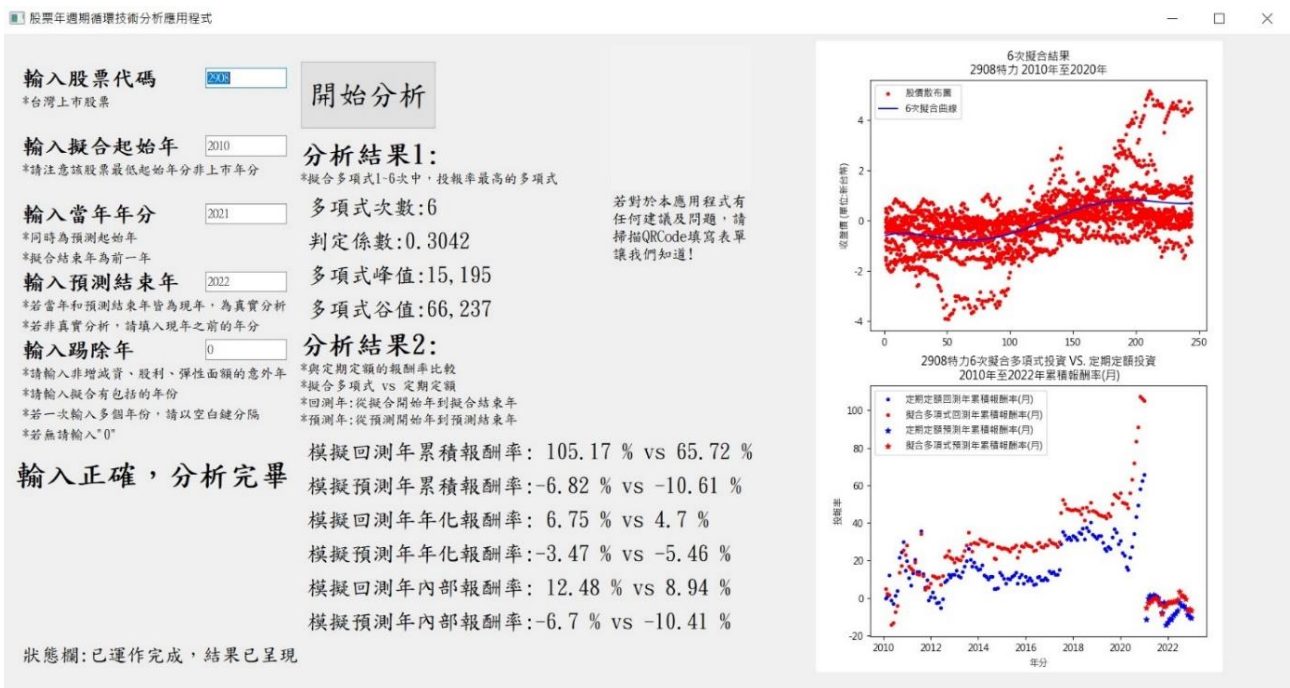
將顯示投報率最多項式的擬合結果圖

將顯示與定期定額報酬率比較圖

(七)程式分析完成後，狀態欄會顯示「已運作完成，結果已呈現」，並且程式會將分析結果顯示於分析股票的介面，此時分析完畢，如圖 20。

圖 20

使用者介面顯示分析結果



輸入正確，分析完畢

(八)若想進行下一次分析，可直接重複(四)到(七)步驟，無須重啟程式，但請確保這次分析已經結束，才可進行下一次分析。

## 伍、討論

### 一、Python 做法相對於 Octave 做法的改進

#### (一)用於輸入及顯示結果的互動平台

在 Octave 做法中，沒有一個方便使用者輸入、明確顯示最後分析結果的使用者介面，因此在 Python 做法中，我們建立了使用者介面 2 已達成此目的。

#### (二)股票資料的平移並儲存

在 Octave 做法中，我們是手動算出股票資料的平均，並將其減去，使其達到平移的效果，且儲存方面也是手動儲存，因此在 Python 做法中，我們製作了 `arrange` 程式來解決此問題。

#### (三)繪製結果圖片

在 Octave 做法中，我們是將平移後數據放入 Octave 後，使其計算出擬合多項式，並經由疊圖後截圖下來，最後再手動加上標題和圖標，因此在 Python 做法中，我們直接於程式中自動繪圖並儲存圖片，以解決此問題。

#### (四)將擬合程度及預測結果量化

在 Octave 做法中，沒有將擬合程度及預測結果量化的動作，導致只能利用肉眼去觀察，因此在 Python 做法中，我們利用判定係數來使擬合程度量化，再利用三個投資報酬率來將預測結果量化。

#### (五)多項式次數、擬合年分及預測年份

在 Octave 做法中，我們將多項式次數固定為三次，擬合年分固定為 2011 至 2018 年，預測年份為 2019 年，這樣無法看到不同年份或是不同次數的結果，因此在 Python 做法中，我們將這三項改為讓使用者輸入，使其彈性更高。

### 二、此應用程式的尚待改進之處

#### (一)股票代號只能輸入台灣上市公司

由於此應用程式原本的構想只是在於研究台灣公司的股價，再加上 Yahoo 經調整收

盤價只記錄已上市的公司，因此本應用程式的股票代號只能輸入台灣上市公司。

### (二)擬合起始年分最早只能輸入 2000 年

因為 Yahoo 經調整收盤價於 2000 年才開始收集，因此在 2000 年以前就已上市的公司，無法取得其所有上市期間的股價。

### (三)運作速度慢

因為在分析過程中有六個程式需要執行，加上下載股票資訊也需要時間，因此若要分析尚未分析過的股票，在使用者介面 2 中點擊「開始分析」後，需要大約 5 至 10 秒的時間運作，才會顯示最終結果。

## 三、股票分析類別及本研究定位

分析股票一般分成四大分析，分別為基本、技術、籌碼與消息分析，本研究屬於技術分析。技術分析是以「歷史會重演」的概念為基礎，一般分成型態學(觀察圖形)及指標法(分析數據)兩種，而本研究屬指標法的範疇。表 3 為一些股票常見的分析方式及本研究的定位。

表 3

股票常見的分析方式及本研究的定位

觀察對象	分析類別	觀察重點
公司 Q1 財報	基本分析	公司經營狀況
三大法人買賣超	籌碼分析	資金流動狀況
公司 CEO 換人	消息分析	消息對公司的影響
壓力、支撐線	技術分析-形態學	股價可能反轉
MA 線	技術分析-指標法	特徵：平均買入價格
本研究	技術分析-指標法	特徵：年週期循環

技術分析指標法，係運用公式與歷史數據，計算出股價會反覆出現的一種特徵以供觀察；而本研究，是運用多項式擬合技術，觀察的特徵是「股價的年週期循環」。

如果投資者根據分析結果，評估某支股票不具年週期趨勢，則可以避免被來路不明的資訊困惑，用科學的方式得到資訊；若投資者評估結果可能具有年週期趨勢，對投資者就是一個重要的參考依據。根據研究過程或方法中，觀察技術分析的結果，有別於其它技術分析，本研究努力在觀察股價年週期循環特徵的領域做出貢獻，我們相信，無論使用者評估該股票是否有年週期循環的現象，此技術分析都可以作為使用者理性且客觀的投資工具。

## 陸、結論

### 一、程式

本研究觀察市面上股票技術分析的方式後，提供了一種新的股票分析方式，即「擬合多項式技術分析」。這種分析方法基於技術分析，透過擬合多項式模型對股票價格進行預測，並在合適的時機點進行買進或賣出操作。相對於傳統的股票分析方式，它可以更直接地捕捉股票價格年週期性的波動，對於投資者來說，具有一定的參考價值。

為了能夠更好且更直觀的使用此分析方式，我們採取迭代式開發，最終用 Python 程式語言建立了「股價年週期循環技術分析應用程式」，我們透過提供最適擬合多項式次數、判定係數、峰值谷值及比較擬合多項式投資與定期定額投資的報酬率等方式，讓使用者能獲得更多資訊，進而判斷此分析方式是否適合運用於此股票，以達到我們最初的研究目的。

完成應用程式後，我們發現了其優點和缺點。股票年週期循環技術分析應用程式的主要優點是其簡單性和易用性，使用者只需輸入一些股票基本的資訊，就可以進行完整的分析。

然而，此應用程式也存在一些缺點。首先，由於這種方法主要基於歷史股價數據進行分析，因此雖然可以預測到未來股票價格的趨勢，但是無法精準預測。其次，這種方法還存在著一定的風險，特別是在市場出現劇烈波動時，股票價格的趨勢可能會遭到改變，此時擬合多項式技術分析會失去其預測能力，因此投資者也必須觀察其他的環境因素，謹慎將其作為投資決策的參考。

且如同上面的討論所述，此應用程式依然有許多的缺陷等待著我們去完善，我們會依照上面討論所提到的缺點，以及使用者使用完本應用程式後提出的反饋，進行修正及繼續研究，並將新的應用程式版本傳至提供下載的地方，供使用者進行更新的動作。

綜上所述，我們觀察了市面上股票技術分析，研究了一種分析方法叫做「擬合多項式技術分析」，並製作出「股票年週期循環技術分析應用程式」，使此分析方法可以更直觀的操作，我們也希望可以讓此應用程式更完善，以提供一個更良好的分析環境。

存放股價年週期循環技術分析應用程式的連結請見附錄。



## 二、未來展望

- (一)股價加權：考慮趨勢、時間、通膨等因素，使其更接近真實。
- (二)延伸比較：把預測對象擴展到基金、期貨、債券等其他金融商品。
- (三)心理因素：分析風險以符合投資人的需求。
- (四)優化使用者介面：優化使用者介面，使互動性更高、功能更強大。
- (五)更換程式語言：改善 Python 本身速度慢、維護困難等缺點。
- (六)函數延伸：將擬合曲線從多項式不同次數，延伸至擬合不同函數的比較。

## 柒、參考文獻資料

- 1.金鐵英、黃盈智(2016)。台灣股市農曆月份效應之研究。財金論文叢刊，25(1)，33-61。  
<http://finance.cyut.edu.tw/var/file/25/1025/img/576/2016v2-3.pdf>
- 2.許繼元(2022年1月30日)。景氣循環股是什麼?如何判斷景氣循環股的買賣點。  
**Mr.Market 市場先生**。<https://rich01.com/what-is-cyclical-stocks/>
- 3.黃益松、朱曉萍、張旭玲(2007)。臺灣股票市場元月及春節效應探討。臺中教育大學學報：數理科技類，21(2)，7-19。<https://doi.org/10.7037/JNTUMST.200712.0007>
- 4.a Leader in Agile Training (2019, March 2). *Agile's 3 categories: Iterative, Incremental, and Evolutionary*. Agile Notion. <https://agilenotion.com/agile-categoriesiterative-incremental-evolutionary/?fbclid=IwAR1-9xz0e12KSYAHZwSratzz0fQdiy6QpPZzHNm5XSu7EjdFYmpxj9x5vLo>
- 5.Nugroho, S., Waluyo, S. H., Hakim, J. (2017). Comparative Analysis of Software Development Methods between Parallel, V-Shaped and Iterative. *International Journal of Computer Applications*, 169(11), 7-11. <https://doi.org/10.5120/ijca2017914605>

## 捌、附錄

存放股價年週期循環技術分析應用程式的連結：[https://drive.google.com/drive/folders/1E-Ixq\\_D2DeaBk-gqx3ToylGvOKUAqSUC?usp=sharing](https://drive.google.com/drive/folders/1E-Ixq_D2DeaBk-gqx3ToylGvOKUAqSUC?usp=sharing)

## 【評語】 052509

此作品觀察市面上股票技術分析的方式後，提供了一種新的股票分析方式，即「擬合多項式技術分析」。這種分析方法基於技術分析，透過擬合多項式模型對股票價格進行預測，並在合適的時機點進行買進或賣出操作。相對於傳統的股票分析方式，此作品可以更直接地捕捉股票價格年週期性的波動，對於投資者來說，有提供一些參考價值。

在未來建議可以探索以下的問題：

哪一個國家的股市趨勢？

在台灣古視訊練出來的模型可以應用到其它國家嗎？

準確率？

這些需要有客觀的數據來證明。

另外，需要特徵為何？需輸入何種資料？此部分需要詳細的討論。



# 作品海報

股價年週期循環

技術分析應用程式

## 壹、前言

### 一、研究動機

可能導致股價年週期循環的原因有：淡旺季、資金、環境影響等。所以最直接的分析方式為針對股價進行分析，然而目前市場上相關的分析工具仍不普及，且綜觀目前幾十甚至幾百種技術分析，幾乎找不到可以有效率看出股價是否具年週期循環現象的方式，於是出於好奇與熱情，我們決定開發一個分析股價年週期循環的應用程式。

### 二、研究目的

建立一個應用程式。該應用程式可以運用多項式擬合技術，顯示出我國個股股價年週期循環趨勢，使投資型使用者可以藉此進行技術分析，研究型使用者可以藉此取得相關資料與數據。

### 三、文獻回顧

1. 軟體開發生命週期(SDLC):意指軟體從需求性開始，逐漸發展到成熟階段的過程，常見的有瀑布式、敏捷式與螺旋式等開發模型，最終我們選擇了**敏捷式**中的**迭代開發模型**。
2. 年週期循環類似概念:有景氣循環和月份效應，和本報告研究不同之處分別為:景氣循環的週期長度不只一年;月份效應的研究對象為整個市場。

## 貳、研究設備及器材

一、**硬體**:筆記型電腦

二、**軟體**:Python、Qt Designer、Visual Studio Code、Excel、Octave、CMD

## 參、研究過程與方法

### 一、需求分析

(一)名詞釋義

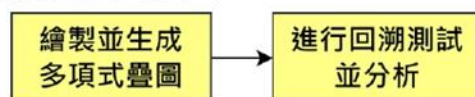
1. 預測:指未來趨勢，非股價的精確值。
2. 擬合多項式技術分析:是利用歷史股價與擬合多項式，分析股價年週期循環的方式。

(二)觀察技術分析

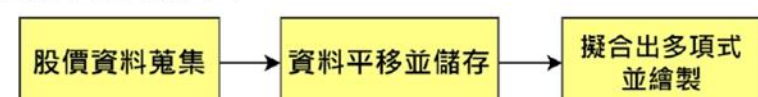
1. 大多數技術分析都會以線圖呈現，方便使用者觀察。
2. 以年為週期分析未來股價趨勢的方式尚不普及。

(三)**討論需求**

1. 產出結果需求:



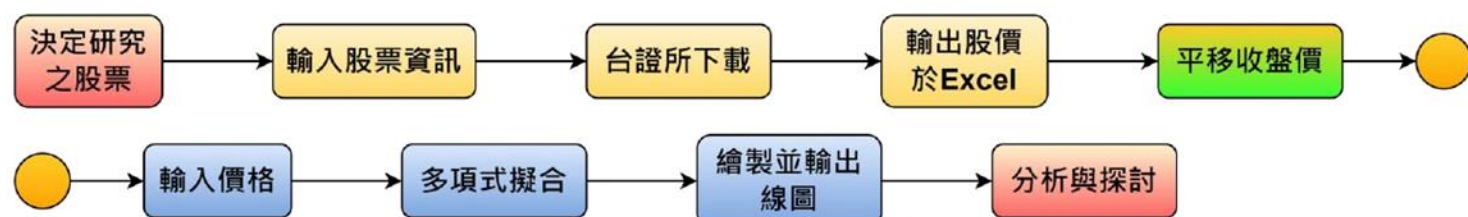
2. 應用程式需求:



### 二、Octave 做法

(一)研究架構

如下圖，先決定研究之股票，米白色部分表示使用 Python 執行，綠色部分表示使用 Excel 執行，藍色部分表示使用 Octave 執行，最後分析與探討。



(二)研究對象

我們選擇分析我國上市各類股，市值前十大的股票。原因是其趨勢和波動更穩定，較不會因小資金流動而有明顯變化，炒作門檻也較高。

(三)研究方式

先將 2011 年到 2018 年八年的股價平移，接著將八年相同交易日的平移後價格取算術平均數，然後再擬合成一個三次多項式，並與 2019 年平移後股價擬合成的三次多項式疊圖。選擇 2011 年是因為台證所資料限制;選擇 2019 年是因為其年末 COVID-19 疫情爆發，可能造成產業結構改變而使股票失去原趨勢。

(四)研究分析

經過分析與整理後，本研究將所有線圖分成線圖貼近、相同起伏、部分起伏符合、毫不相干和異常五類，線圖貼近和相同起伏為符合規律線圖，部分起伏符合和毫不相干為不符合規律線圖。結果發現，各類股的股票皆為部份符合規律，部分不符合規律，以此判斷此分析方式的確可行。

(五)**討論需求與變因**

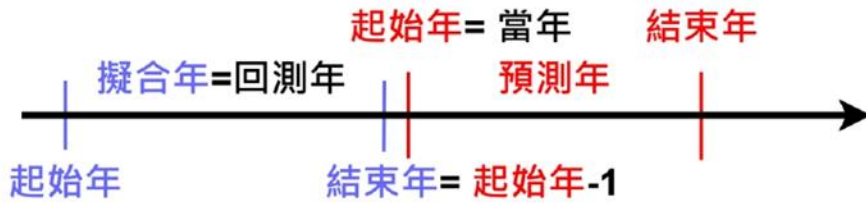
1. 缺乏**互動平台**用於輸入及顯示結果
2. 缺乏**自動平移**數據、**繪圖儲存**功能
3. 無法將**擬合程度**量化，只能以肉眼觀測
4. 沒有**對照組**和本分析方式做比較
5. 多項式次數固定為 3 次
6. 擬合年份固定為 2011 年至 2018 年
7. 預測年份固定為 2019 年
8. 台證所資料不足，無 2011 年前的股價



### 三、Python 做法

#### (一)名詞釋義

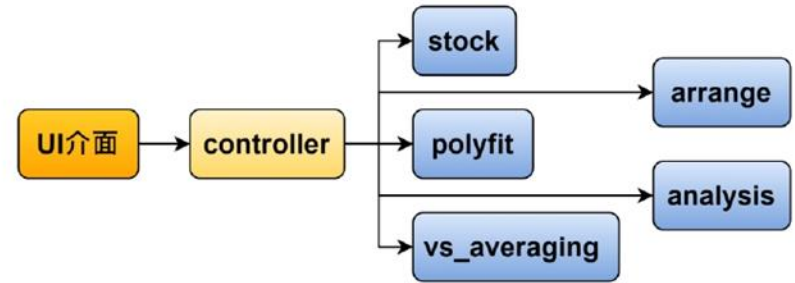
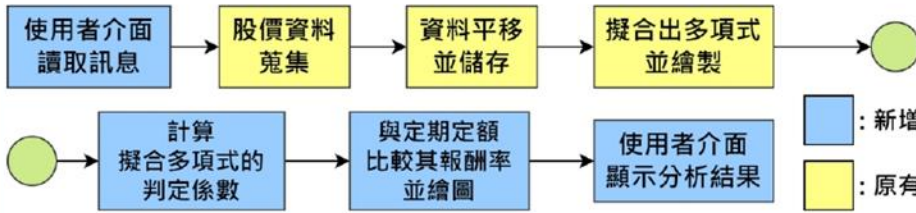
1. 模擬分析&真實分析:都是擬合多項式技術分析的一部分，在進行此技術分析時，先模擬分析，再真實分析。
2. 擬合年&回測年&當年&預測年:
3. 擬合多項式投資:



即運用我們最終「股價年週期循環技術分析應用程式」所得出的結果，進行投資的方式。其中「最適次數」，是運用擬合多項式投資，分析出所有擬合多項式次數後，所得出回測年報酬率最高的多項式次數。

#### (二)研究架構

1. 設計流程:根據 Octave 做法的討論，流程增加為七階段
2. 程式架構:以設計流程為基礎，設計出以下架構



#### (三)開發之程式介紹

##### 1. 使用者介面程式

start1、controller1 和 ui1 管理**使用者介面 1**，使用者介面 1 包含使用說明及一按鈕「繼續」，點按即可跳至使用者介面 2。start2、controller2 和 ui2 管理**使用者介面 2**，使用者介面 2 包含分析介面及一按鈕「開始分析」，點擊後即開始分析，分析完後會顯示分析結果。

##### 2. 中央處理程式:center

**統整**執行除錯、資料處理、資料分析和對照組比較**程式**，並顯示回傳分析結果，將其導至 controller2。

##### 3. 除錯程式:debug

判定介面中所輸入之**資料是否正確**，是則繼續運作，否則回傳錯誤原因。

##### 4. 資料分析程式:polyfit & analysis

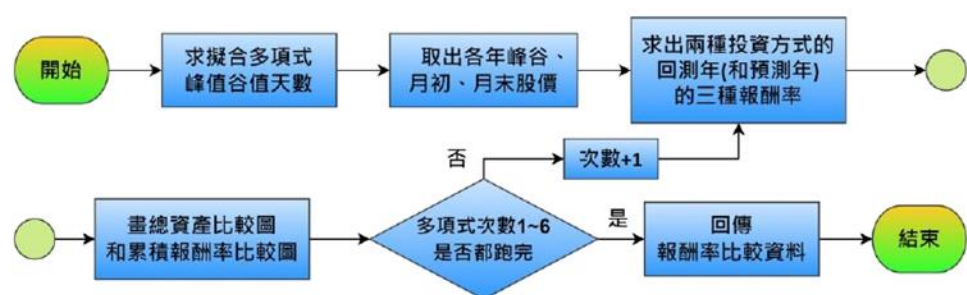
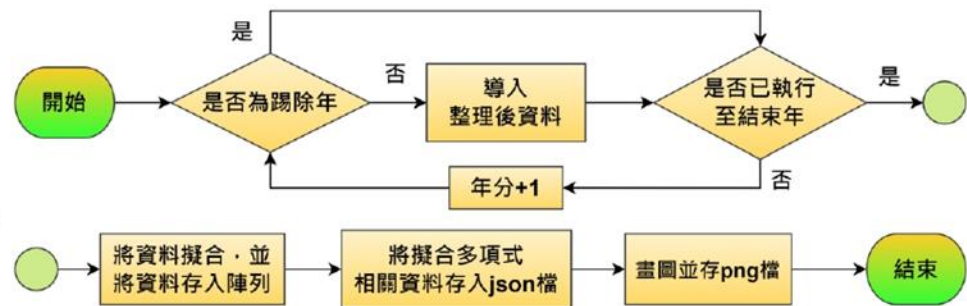
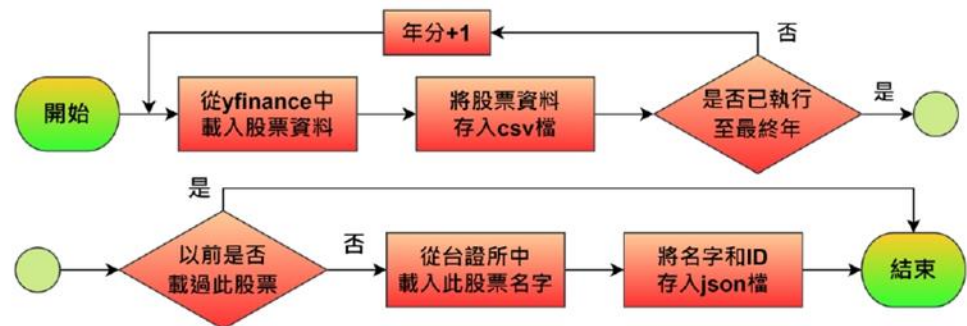
polyfit 的用途為**擬合出多項式並繪圖**，生成擬合多項式和疊圖，流程圖如右圖(黃)。analysis 的用途為**計算擬合多項式之判定係數**，也就是將擬合結果量化。

##### 5. 對照組比較程式: vs\_averaging1 & 2

vs\_averaging1 用於**模擬分析**，**計算並比較定期定額和多項式投資方法**回測年與預測年的總資產和**報酬率**。vs\_averaging2 用於**真實分析**，相較於 vs\_averaging1 來說，缺少預測年的比較。二者流程圖如右圖(藍)。

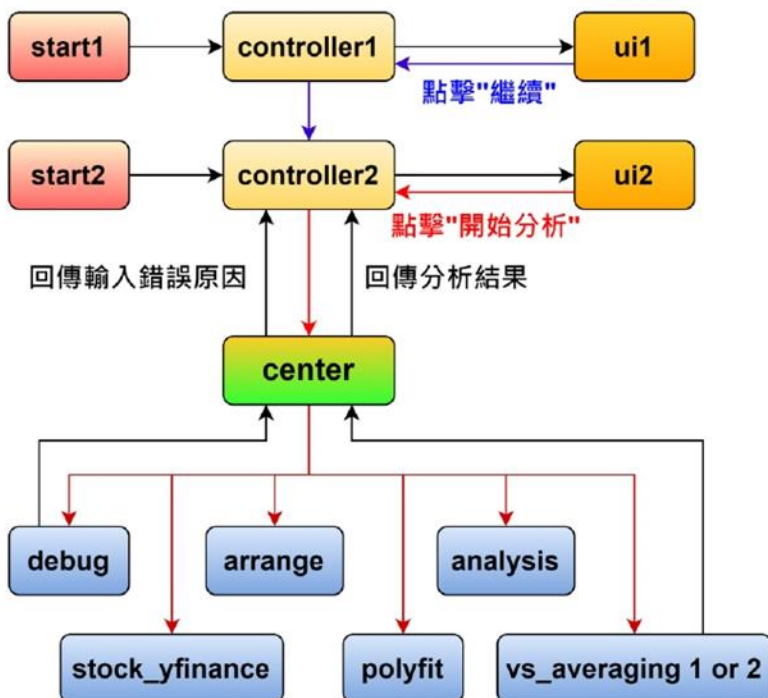
##### 6. 資料處理程式:stock\_yfinance & arrange

stock\_yfinance 的用途為**下載調整後股價**以及股票的中文名。流程圖如下(紅)。arrange 的用途為將**股價平移**，並在平移後存入一新檔案



#### (四)開發結果

##### 1. 最終程式架構



### 四、部署與維護

#### (一)定義角色

「使用者」指使用本應用程式之不特定瀏覽民眾，這個角色主要可以進行應用程式之功能。而「維護者」用於程式程序設計與應用程式維護管理，可以接收使用者回饋進而修改程式與偵測錯誤。

#### (二)部署

我們將最終的「股價年週期循環技術分析應用程式」**存放於 Google 雲端硬碟**，並開放存取權為「知道連結的任何人」，目的是有個公開平台，可以讓使用者自由下載。

而接收回饋的部分，我們建立一個 **Google 表單**，使用者可以在裡面輸入建議、報錯或給予回饋。

#### (三)維護

維護者根據 **Google 表單**的建議或報錯，不定期修正或優化此應用程式，並上傳更新版本至 **Google 雲端硬碟**。



## 肆、研究結果

### 一、擬合多項式投資

先求出**擬合多項式的峰值與谷值**；接著在谷值用所有現金購入股票，並且只要尚未達到下個峰值，有資金就立刻購入股票，直到達到峰值當天，將所有股票賣出，並且持續持有現金，等到下次谷值再用所有現金購入股票，目的為**求出最適次數及得到相關結果**。

### 二、分析方式

#### (一) 模擬分析

目的為評估此技術分析**是否適合**這支股票。分析方式為：讓使用者假設自己在模擬當年運用此擬合多項式技術分析，並將此分析結果運用在實際市場投資。

#### (二) 真實分析

真實分析的前提為模擬分析中評估此分析適合這支股票，而分析方式為得出**真實投資的分析結果**，並對其進行最後的評估。

### 三、使用者介面 2

#### (一) 輸入區

輸入股票代碼   
\*台灣上市股票

輸入擬合起始年   
\*請注意該股票最低起始年非上市年分

輸入當年年分   
\*同時為預測起始年  
\*擬合結束年為前一年

輸入預測結束年   
\*若當年和預測結束年皆為現年，為真實分析  
\*若非真實分析，請填入現年之前的年分

輸入踢除年   
\*請輸入非增減資、股利、彈性面額的意外年  
\*請輸入擬合有包括的年份  
\*若一次輸入多個年份，請以空白鍵分隔  
\*若無請輸入"0"

#### (二) 狀態區

##### 輸入錯誤：

股票代碼輸入錯誤!

狀態欄:請重新輸入股票相關資料

#### (三) 分析結果區

##### 分析結果1:

\*擬合多項式1-6次中，投報率最高的多項式

多項式次數:

判定係數:

多項式峰值:

多項式谷值:

##### 分析結果2:

\*與定期定額的報酬率比較

\*擬合多項式 VS 定期定額

\*回測年:從擬合開始年到擬合結束年

\*預測年:從預測開始年到預測結束年

模擬/真實回測年累積報酬率:

模擬/真實預測年累積報酬率:

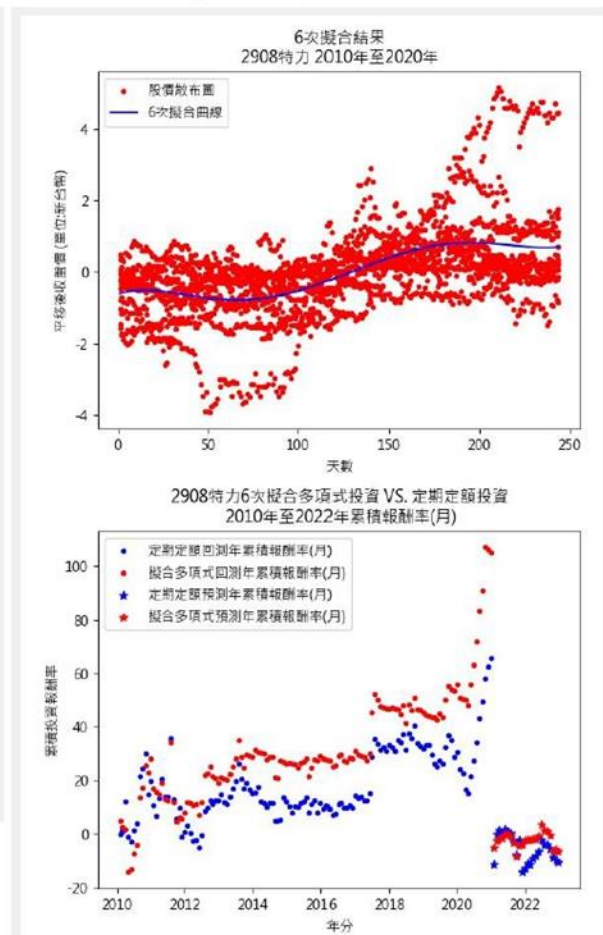
模擬/真實回測年年化報酬率:

模擬/真實預測年年化報酬率:

模擬/真實回測年內部報酬率:

模擬/真實預測年內部報酬率:

#### (四) 圖示區:多項式疊圖&對照組比較圖



## 伍、討論

### 一、Python 做法相對於 Octave 做法的改進

1. 互動平台:使用者介面 2
2. 自動平移數據:程式 arrange
3. 繪圖儲存:matplotlib 模組
4. 分析結果量化&對照組:判定係數&定期定額
5. 次數&年分靈活度:輸入區
6. 資料來源&股價波動: yfinance 的調整後股價

### 二、本程式的尚待改進之處

1. 分析對象限制
2. 資料年份限制
3. 運作速度

### 三、股票分析類別及本研究定位

分析股票分成基本、消息、籌碼與技術分析，本研究屬於技術分析中的指標法，有別於其他指標法，本研究努力在觀察「**股價年週期循環特徵**」做出貢獻。我們認為無論分析的股票是否有年週期循環的現象，此分析都可以作為投資者理性且客觀的分析工具

## 陸、結論

本研究成功開發了「**股價年週期循環技術分析應用程式**」，提供了一個新的技術分析方式，即「**擬合多項式技術分析**」，可以直接捕捉股票價格年週期性的波動，並輔以相關資料，達到最初的研究目的。此分析亦有優缺點，缺點為未進行風險評估;優點為其直觀、合理且創新的分析方式。

我們的未來展望分為六點，分別為:股價加權、延伸比較、心理因素、優化使用者介面、更換程式語言以及函數延伸。我們希望讓此應用程式更完善，以提供一個更良好的分析環境。

## 柒、參考資料

1. 金鐵英、黃盈智(2016)。台灣股市農曆月份效應之研究。財金論文叢刊, 25(1), 33-61。  
<http://finance.cyut.edu.tw/var/file/25/1025/img/576/2016v2-3.pdf>
2. 許繼元(2022年1月30日)。景氣循環股是什麼?如何判斷景氣循環股的買賣點。Mr.Market 市場先生。  
<https://rich01.com/what-is-cyclical-stocks/>
3. 黃益松、朱曉萍、張旭玲(2007)。臺灣股票市場元月及春節效應探討。臺中教育大學學報:數理科技類, 21(2), 7-19。  
<https://doi.org/10.7037/JNTUMST.200712.0007>
4. a Leader in Agile Training (2019, March 2). Agile's 3 categories: Iterative, Incremental, and Evolutionary. Agile Notion. <https://agilenotion.com/agile-categories-iterative-incremental-evolutionary/?fbclid=IwAR1-9xz0e12KSYAHZwSratz0fQdjy6QpPZzHNm5XSu7EjdfYmpxj9x5vLo>
5. Nugroho, S., Waluyo, S.H., Hakim, J. (2017). Comparative Analysis of Software Development Methods between Parallel, V-Shaped and Iterative. International Journal of Computer Applications, 169(11), 7-11.  
<https://doi.org/10.5120/ijca2017914605>