

巨量資料與金融科技實務

期末報告

以規模、帳面市值比、動能作為因子，運用 AI 機器學習建構投資組合

學生姓名：陳柏瑋
班 級：財精二 C
學 號：11155312
指導老師：鄭宏文

2024 年 6 月 15 日

目次

目次	2
壹、前言	3
貳、動機	3
參、研究方法	3
一、投資架構流程圖	4
肆、資料來源	5
一、台灣經濟新報資料庫（Taiwan Economic Journal，TEJ）介紹	5
二、資料收集步驟	5
伍、資料處理	7
陸、程式撰寫	13
柒、成果展現與分析	20
一、勝率分析	20
二、三種機器學習方法在不同等份投資組合之累計報酬分析	21
三、在不同等份之投資組合下三種機器學習方法的表現分析	24
捌、結論	26
玖、心得	26

壹、前言

人工智能的發展，為各個行業提供各種提升效率及準確性的工具，金融領域也不例外。金融市場的歷史悠久，留下了龐大的金融數據，有足夠多的資料供人工智能使用。隨著各種機器學習方法的研發，使更多人在量化交易及優化投資組合上使用人工智能。

其中，因子投資在人工智能的發展中逐漸受到大眾的關注。自 1960 年代發表資本資產定價模型（Capital Asset Pricing Model，CAPM），及 1993 年提出的三因子模型（Fama-French Three-Factor Model）後，人們發現股票報酬與特定因子存在一定的相關性，使得因子投資使用的比例有顯著的成長。

貳、動機

傳統的股市投資人靠著自身對市場的見解，以及長年的經驗累積來建構自己的投資組合。然而，比起電腦的運算，人們容易受到各種情緒、各種消息的影響，而做出不理性的投資，透過電腦運算的結果進行決策的依據能夠避免掉這方面的影響。此外，利用演算法及機器學習的方法對過往的歷史數據進行分析，能夠發掘出資料間難以以肉眼發現的相關性，進而找到投資機會。

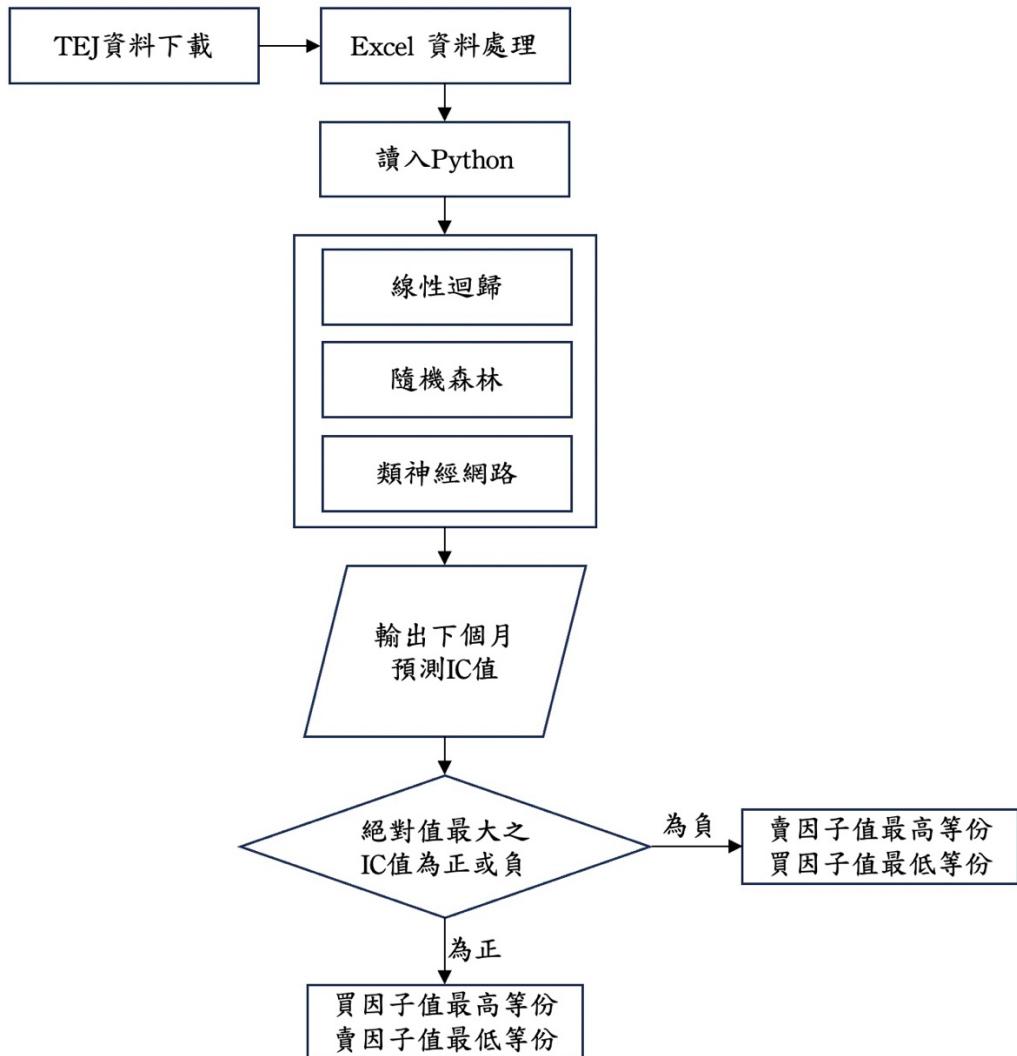
此項研究利用機器學習的方法，透過歷史資料建立預測下個月 IC 值（Information Coefficient）的模型來建構投資組合，希望能夠建構穩定且勝率高的投資組合。

參、研究方法

本研究以規模（Size）、帳面市值比（Book-to-Market）及動能（Momentum）做為因子，分別利用當月三種因子與下個月月報酬計算 IC 值，並且切分 2000 年 1 月到 2013 年 11 月資料作為訓練集，以此分別利用線性迴歸、隨機森林及類神經網路三種機器學習方法來建立模型預測下月 IC 值。

利用模型預測出的下個月 IC 值，挑選出絕對值比較大的 IC 值，以該因子作為建立投資組合的依據。我們將依照因子值的高低分把所有股票分成 10 等份、20 等份、50 等份、100 等份、200 等份、300 等份、450 等份及 900 等份作為投資組合。若是 IC 值預測為正，則採取買進排名最高等分，賣出排名最低等分的策略。若是 IC 值預測為負，則採取買進排名最低等分，賣出排名最高等分的策略，並且與大盤月報酬做比較。以此方式分別用三種機器學習方法做比較，比較勝率較高的方法。

一、投資架構流程圖



肆、資料來源

一、台灣經濟新報資料庫（Taiwan Economic Journal, TEJ）介紹

台灣股市每天產生許多交易資訊，除了股價波動的資訊外，也有各種公司公告的消息，營收、獲利、股利等消息。這些資訊雖然在網路上都能夠取得，然而每天收穫這些資料是非常曠日費時的，且資料的品質也難以保證。

為此 TEJ 收集了大量的台股資料，並且定期有研究員對資料進行清洗與校閱，提供了乾淨、方便、可分析的資料。滿足了量化交易人的需求。

二、資料收集步驟

1. 在 TEJ 中點選「未調整股價（月）」。

The screenshot shows the TEJ Pro software interface. On the left, there's a sidebar with 'Smart Inquiry', 'Data Search' (which is circled in red), and 'Event Research'. The main area displays 'TQuant Lab 線上工作台' with the sub-section '一起進入量化金融的' and the date 'Apr 12 2024'. On the right, a detailed list of data series is shown, with '未調整股價(月)' also circled in red.

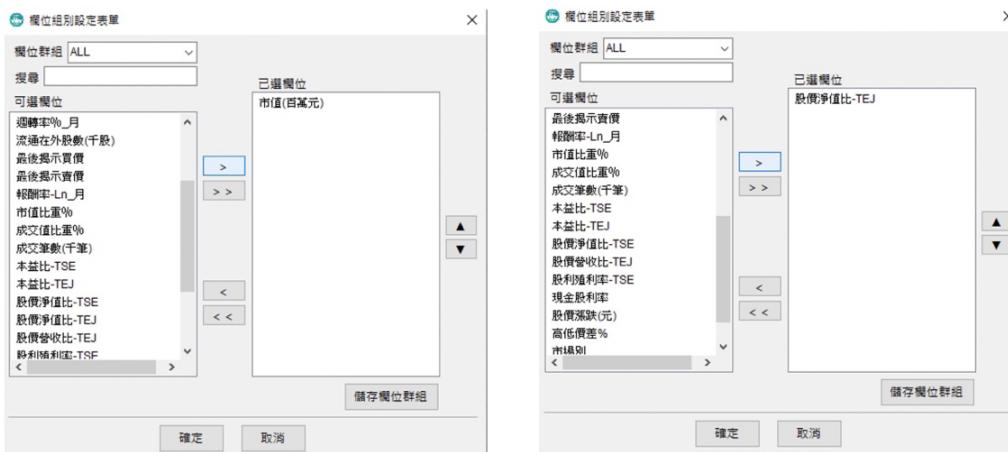
代號	名稱
1	元大台灣50
2	元大中型100
3	富邦科技
4	元大電子
5	元大MSCI金融
6	元大高股息
7	富邦摩台
8	元大寶滙深
9	元大MSCI台灣
10	永豐臺灣加權
11	富邦上延
12	元大上證50
13	復華滙深
14	富邦台50
15	富邦上延+R
16	元大台灣50正2
17	元大台灣50反1
18	富邦上延正2
19	富邦上延反1
20	元大S&P黃金
21	國泰中國A50

2. 在欄位中點選「收盤價」，並且調整時間為 2000/01/01 至 2023/12/31。

This screenshot shows two dialog boxes. The left one is '欄位組別設定表單' with a list of fields like '開盤價(元)', '最高價(元)', etc., and '收盤價(元)' highlighted with a red circle. The right one is '進階日期設定' for selecting a date range from '20000101' to '20231231', with the entire date range field also circled in red.

3. 按下「匯出」，即可得到 2000/01/01 至 2023/12/31 所有台股上市上櫃公司每月的收盤價資料。

4. 利用同樣方法在欄位中選擇「市值」及「股價淨值比」，並且調整時間為 2000/01/01 至 2023/12/31。(由於 TEJ 的帳面市值比資料較少，因此我們利用股價淨值比資料再取其倒數，以此獲得帳面市值比資料)



5. 按下「匯出」，獲得 2000/01/01 至 2023/12/31 所有台股上市上櫃公司每月的市值與股價淨值比資料。

伍、資料處理

在分析資料前，我們先在 Excel 內將資料處理成方便分析的形式。首先我們需要清理資料，抽出沒有值且股票代碼非四碼的股票。接著我們要計算下個月月報酬、市值、動能及帳面市值比的資料並且將資料中缺值的部分補上。最後，透過計算出來的結果與下個月月報酬計算 IC 值。

步驟：

一、刪除在 2005 年 1 月當下收盤價沒有值的公司

- 全選 2005/1 整行，使用「尋找與搜尋」來找出沒有值的儲存格。



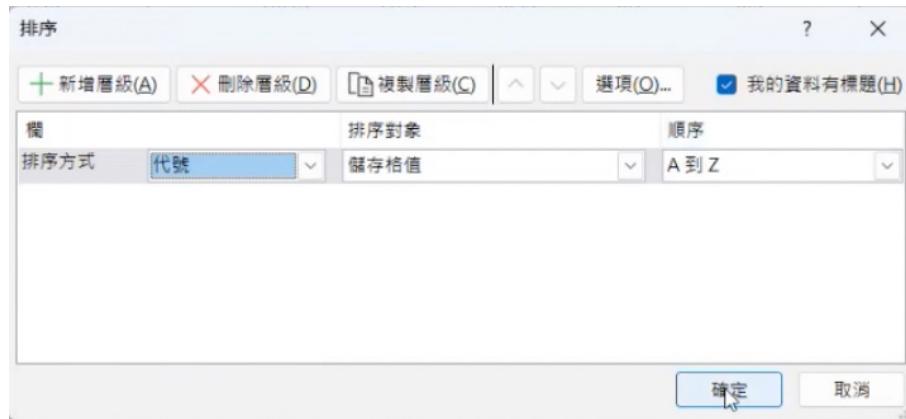
- 刪除在 2005 年 1 月當下沒有值的列

二、刪除股票代碼非四碼股票

因為本研究僅探討台灣 900 多家上市上櫃公司股票，因此我們將刪除股票代碼非四碼的股票。

1. 全選整張工作表

2. 點選排序，排序方式設為代號，排序對象設為儲存格值，且順序為遞增排序。



3. 按下確定。股票代碼非四碼的股票即出現在工作表的最下方。

三、將市值與股價淨值比的表頭調整成相同的排序。

1. 複製調整完順序的收盤價工作表的表頭，在新開的工作表上貼上。

代號	名稱	2000/01	2000/02	2000/03	2000/04	2000/05	2000/06	2000/07	2000/08	2000/09	2000/10	2000/11
1101	台泥											
1102	亞泥											
1103	嘉泥											
1104	環泥											
1108	幸福											
1109	信大											
1110	東泥											
1201	味全											
1203	味王											
1210	大成											
1213	大飲											
1215	卜蜂											
1216	統一											
1217	愛之味											
1218	泰山											
1219	福壽											
1220	台榮											
1225	福懋油											
1227	佳格											
1229	聯華											
1231	聯華食											
1232	大統益											
1233	天仁											
1234	黑松											

五、計算下個月月報酬

因為對數報酬具有良好的可加性，我們將使用對數報酬來進行本研究。

對數報酬公式：

$$R_{t+1} = \ln \left(\frac{S_{t+1}}{S_t} \right)$$

其中 R_{t+1} 為時間點 $t+1$ 時的對數報酬值， S_{t+1} 與 S_t 分別為時間點 $t+1$ 以及時間點 t 時的收盤價（ t 的單位為月）。。

指令：IFERROR(LN(收盤價!D2/收盤價!C2),"缺值")

六、計算市值

指令：IFERROR(LOG(市值!C2*1000000),"缺值")

因為從 TEJ 下載的市值資料單位為百萬元，因此我們在指令中將儲存格值乘上 1000000 已呈現原始的數字。接著，我們對市在值資料取對數，得到對數市值的資料。最後，因為有些公司可能在某些資料中沒有值，我們透過 IFERROR 的指令，讓儲存格裡沒有值使對數計算出現 Error 的儲存格放入”缺值”。

Rank IC 計算方式：

$$\text{Rank IC}_t = \text{Corr}(Y_t, R_{t+1})$$

其中 Y_t 為各個因子值在時間點 t 的排序值，而 R_{t+1} 為下個月月報酬在時間點 $t+1$ 時的排序值。我們將兩者的資料計算相關係數而得到 Rank IC 值。

指令：CORREL(size_rank!C\$2:C\$972,下個月月報酬_rank!C\$2:C\$972)

A	B	C	D	E	F	G	H	I	J	K	L	M
1 time	2000/01	2000/02	2000/03	2000/04	2000/05	2000/06	2000/07	2000/08	2000/09	2000/10	2000/11	2000/12
2 RIC	0.02238597	0.20452065	0.04948759	0.29253982	0.04274884	0.1998853	0.13893743	0.03723683	0.06424808	0.22219426	-0.0337692	0.15425745

陸、程式撰寫

使用 Excel 將資料處理完畢後，接者使用 Python 讀取 Excel 中的資料並作出建立模型作出預測。以下使用三種機器學習的方法，分別為線性迴歸（Linear Regression）、隨機森林（Random Forest）以及類神經網路（Artificial Neural Network，ANNs）。

三種方法都依循以下步驟進行，其中最大的差別在於引入的套件以及模型的建立。因此在步驟一以及步驟三會將三種機器學習方法的程式碼分別說明，其餘步驟將以線性回歸為例說明。

步驟一、引入套件

Python 中有許多分析資料以及機器學習的套件，因此我們一開始先引入後面會使用到的套件以方便之後的分析。且因三種機器學習各自有自己的套件，以下將分別進行說明。

線性迴歸（Linear Regression）

```
""" Step 1 引入需要的套件 """
import os
import pandas as pd
import numpy as np
import openpyxl
from sklearn.linear_model import LinearRegression
```

隨機森林（Random Forest）

```
""" Step 1 引入需要的套件 """
import os
import pandas as pd
import numpy as np
import openpyxl
from sklearn.ensemble import RandomForestRegressor
```

類神經網路（Artificial Neural Network，ANNs）

```
""" Step 1 引入需要的套件 """
import os
import pandas as pd
import numpy as np
import openpyxl
from keras.layers import Dense
from keras.models import Sequential
import tensorflow as tf
from tensorflow.keras.optimizer import Adam
```

套件功能：

os: 提供了操作系統中檔案的方法，可以針對檔案進行重新命名、編輯、刪除等相關操作

pandas: 提供許多工具進行數據的轉換、分割等等處理資料的功能

numpy: 支援高維度陣列及矩陣運算，也具有大量數學與統計函式庫，方便進行數據分析。

openpyxl: 用於讀取、寫入和修改 Excel 文件中的數據，同時還可以處理工作表、單元格格式、圖表等內容。

sklearn.linear_model: 提供多種線性模型的運算，而我們使用其中的「LinearRegression」來完成線性迴歸的建模。

sklearn.ensemble: 專門用於各種集成學習的方法，而我們使用其中「RondomForestRegression」來完成隨機森林的建模

keras.layers: 是 keras 中層（Layer）的模組，內含多種神經網路層。而我們將使用其中的「Dense（全連接層，Fully Connected Layer）」來進行建模。

keras.models: 是 keras 中模型的模組，提供建構及訓練模型的工具。而我們將使用其中的「Sequential（順序模組）」來進行建模。

tensorflow: 用於建構以及訓練各種深度學習模型

tensorflow.Keras.optimizer: 用於選擇與配置訓練過程中的優化算法。而我們將使用其中的「Adam」作為優化器。

步驟二、定義路徑

利用 os 套件中 chdir 指令來改變當前資料夾的路徑

```
""" Step 2 定義路徑 """
os.chdir(r'/Users/chenpowei/Documents/巨量資料')
```

步驟三、建立模型

線性迴歸（Linear Regression）

利用 Scikit-Learn 中的線性迴歸模型「LinearRegression」

```
""" Step 3 建立模型[線性迴歸] """
reg = LinearRegression()
```

隨機森林（Random Forest）

利用 Scikit-Learn 中的隨機森林模型「RandomForestRegressor」，設定每顆決策樹最大深度為 2 層，以及決策樹的數量為 100 顆。並且設定 random_state=0，可以確保每次執行程式時都可以獲得相同的隨機森林模型。

```
""" Step 3 建立模型[RandomForest] """
rf = RandomForestRegressor(max_depth=2, random_state=0, n_estimators=100)
```

類神經網路（Artificial Neural Network，ANNs）

在建立類神經網路模型之前，我們需要先設定隨機種子才能讓每次的結果保持一致性及可重複性。

接著，我們利用 keras 中的順序模型（sequential）來建立模型。模型中一共有四層全連接層（Dense），在第一層中設定為 32 個隱藏神經元，並給定輸入數據為一個維度為 971 的向量。第二層設定為 16 個隱藏神經元，第三層為 8 個隱藏神經元，最後一層輸出層為 1 個隱藏神經元。其中前三層都利用 relu（Rectified Linear Unit）激活函數，使模型能夠學習更複雜的函數

最後，設定學習率為 0.001，並使用 Adam 優化器及均方誤差（Mean-Square Error）作為損失函數，以此在模型訓練過程中更新權重並提高預測精度。

```
""" Step 3 建立模型[類神經網路] """
seed = 40
np.random.seed(seed)
tf.random.set_seed(seed)
model = Sequential([
    Dense(32, activation='relu', input_dim = 971),
    Dense(16, activation='relu'),
    Dense(8, activation='relu'),
    Dense(1)
])
adam = Adam(learning_rate=0.001)
model.compile(optimizer = adam, loss= 'mean_squared_error')
```

步驟四、定義預測 IC 值函式

我們定義一個函式來對模型進行訓練以及測試並且在最後輸出預測的 IC 值。在此，三種機器學習方法個自有自己的指令對於模型進行訓練，其中類神經網路多了一個驗證集來訓練資料，因此需要再將資料進行切割。以下利用線性迴歸的程式碼為例並標明其他方法不相同的部分：

以線性迴歸（Linear Regression）為例

```
""" Step 4 預測IC值 """
#定義函式：factor, IC
def facIC(f,i):
    c = pd.DataFrame()
    if f=='mom':
        n1,n2 = 154,274
    else:
        n1,n2 = 166,286
    df_x = pd.read_excel(f+'.xlsx',sheet_name=f+'補值').T
    df_y = pd.read_excel(f+'.xlsx',sheet_name=f+'_'+i).T
    df_x.columns = df_x.iloc[0,:]
    df_y.columns = df_y.iloc[0,:]
    df_x = df_x.iloc[2:n2+3,:]
    df_y = df_y.iloc[1:,:]
    for n in range(n1,n2):
        X = df_x.iloc[:n,:].values
        y = df_y.iloc[1:n+1,:].values.ravel()
        test_X = df_x.iloc[n:n+1,:].values
        reg.fit(X,y)
        pred = reg.predict(test_X)
        b = pd.DataFrame(np.array(pred)[0:])
        c = pd.concat([c,b],axis = 0,sort=False)
    c.columns = [f]
    return c
#計算並合併三個因子結果
#for 迴圈
factor = ['bm','size','mom']
IC = ['RIC', 'NIC']
for i in IC:
    if i == 'RIC':
        RIC = pd.DataFrame()
    else:
        NIC = pd.DataFrame()
    for f in factor:
        if i == 'RIC':
            RIC = pd.concat([RIC,facIC(f,i)],axis = 1)
        else:
            NIC = pd.concat([NIC,facIC(f,i)],axis = 1)
```

紅框部分為模型的建模，三種方法程式碼不相同。以下為隨機森林以及類神經網路的程式碼：

隨機森林（Random Forest）

```
rf.fit(X,y)
pred = rf.predict(test_X)
```

類神經網路（Artificial Neural Network，ANNs）

```
#model.fit不支持 ndarray,故須轉成float64
X = X.astype('float64')
y = y.astype('float64')
test_X = test_X.astype('float64')
#使用validation_split = 0.3 作為驗證及自動化分的比例
model.fit(X,y,validation_split=0.3,batch_size=10,epochs=20)
pred = model.predict(test_X)
```

步驟五、讀取及寫入 Excel

在此我們將先前製做好準備值的 Excel 檔，並且放入所預測 IC 值。

```
""" Step 5 讀取及寫入Excel """
#讀取Excel
Predict = openpyxl.load_workbook('predict_OLS.xlsx',data_only=True)
#寫入Excel
RIC = np.array(RIC)[0:]
NIC = np.array(NIC)[0:]
for k in range(0,3):
    for j in range(0,120):
        for i in IC:
            Predict[i].cell(k+2,j+2).value = eval(i)[j][k]
#儲存Excel
Predict.save('predict_OLS.xlsx')
```

(此處僅需將讀取的 Excel 檔案改成其他機器學習方法的檔案即可)

步驟六、讀入因子值以及下個月月報酬數據

因為接下來要計算我們建構的投資組合的報酬率，我們先將各個因子值以及下個月月報酬的資料讀入 Python。

```
""" Step 6 讀入數據 """
#讀取bm (bm補值、bm下個月月報酬)
bm = pd.read_excel('bm.xlsx',sheet_name="bm補值").T
bm = np.array(bm.iloc[169:,:])
bm_return = pd.read_excel('bm.xlsx',sheet_name="下個月月報酬補值").T
bm_return = np.array(bm_return.iloc[169:,:])

#讀取mom (mom補值、mom下個月月報酬)
mom = pd.read_excel('mom.xlsx',sheet_name="mom補值").T
mom = np.array(mom.iloc[157:,:])
mom_return = pd.read_excel('mom.xlsx',sheet_name="下個月月報酬補值").T
mom_return = np.array(mom_return.iloc[157:,:])

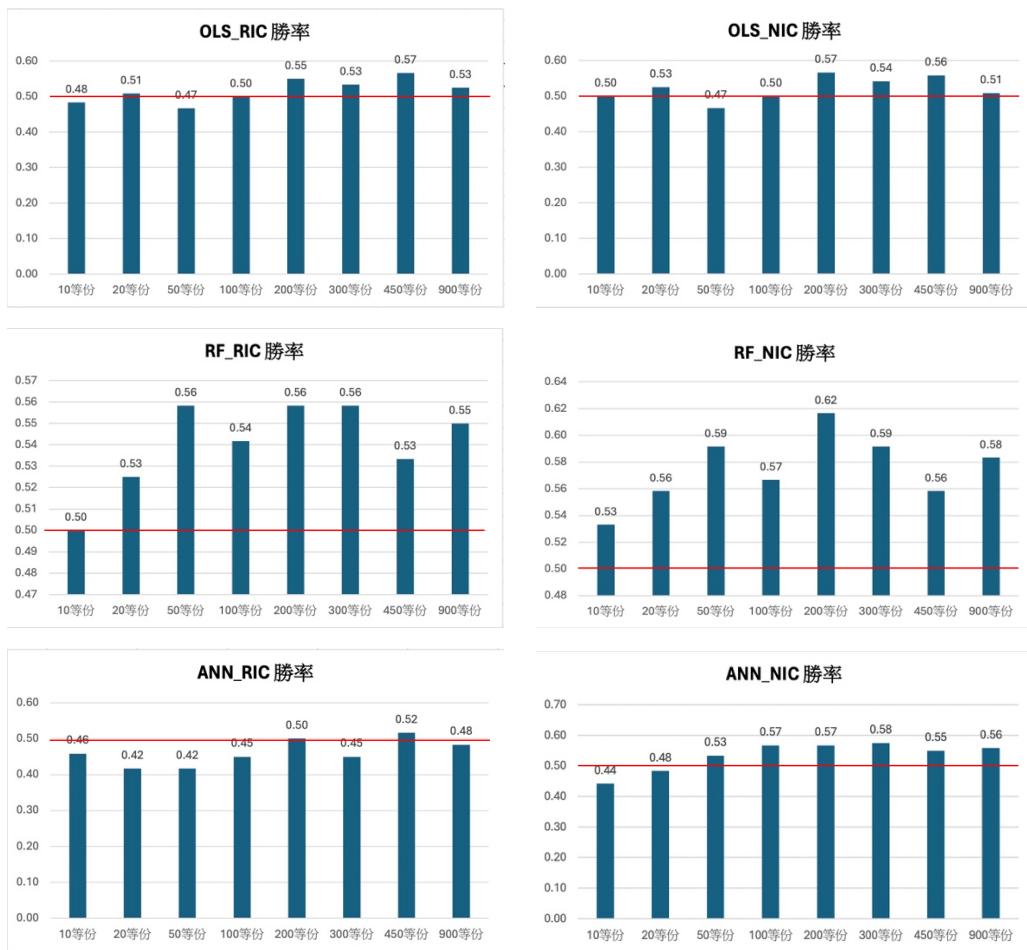
#讀取size (size補值、size下個月月報酬)
size = pd.read_excel('size.xlsx',sheet_name="size補值").T
size = np.array(size.iloc[169:,:])
size_return = pd.read_excel('size.xlsx',sheet_name="下個月月報酬補值").T
size_return = np.array(size_return.iloc[169:,:])
```

(此處三種機器學習方法程式碼一致)

柒、成果展現與分析

有了各個投資組合的報酬率之後，我們也會比較各個投資組合的每個月報酬贏過大盤月報酬的勝率。同時，我們將比較各種機器學習預測 RIC 值以及 NIC 值在不同等份之投資組合下的累積報酬率，並且跟大盤報酬率做比較。除此之外，我們也將比較在等份為 10 等份、20 等份、50 等份、100 等份、200 等份、300 等份、450 等份及 900 等份下各種機器學習方法透過預測 RIC 值以及 NIC 值所建構投資組合之累積報酬率。

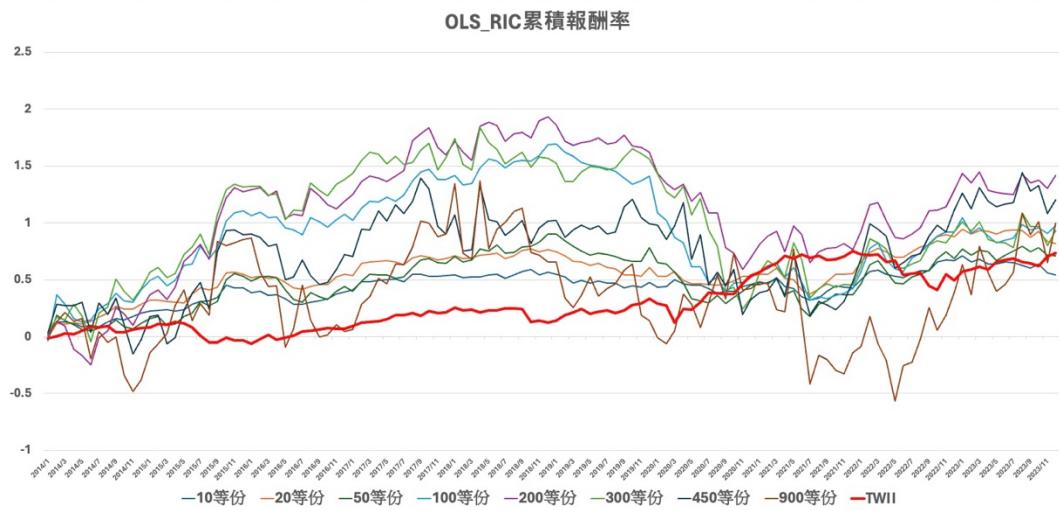
一、勝率分析



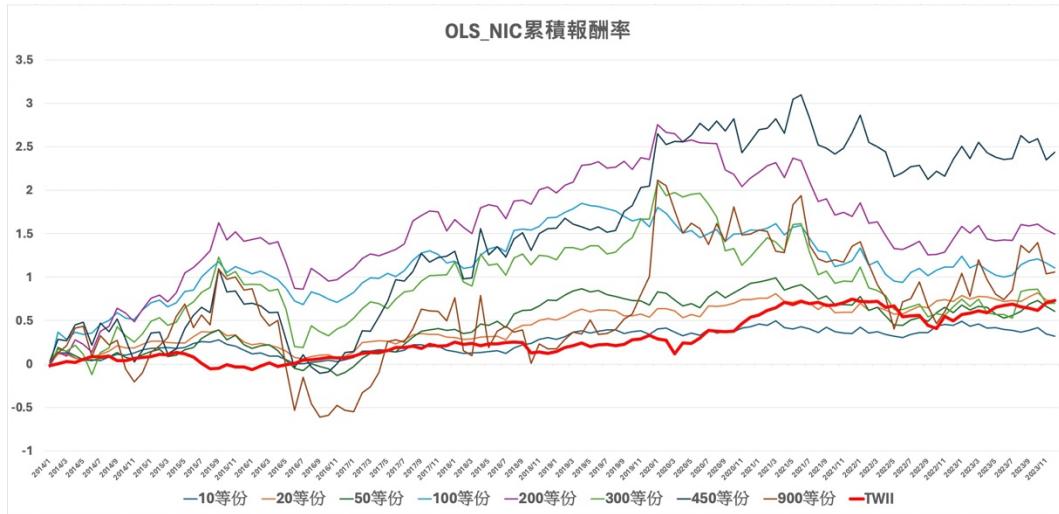
由上表可知，大部分投資組合勝率都能夠超越 50%，意即有超過一半的機會能夠勝過大盤。其中又以隨機森林所建構之投資組合表現最好，無論多少等份之投資組合的勝率都能夠超過或與大盤持平。

二、三種機器學習方法在不同等份之投資組合累計報酬分析

線性迴歸

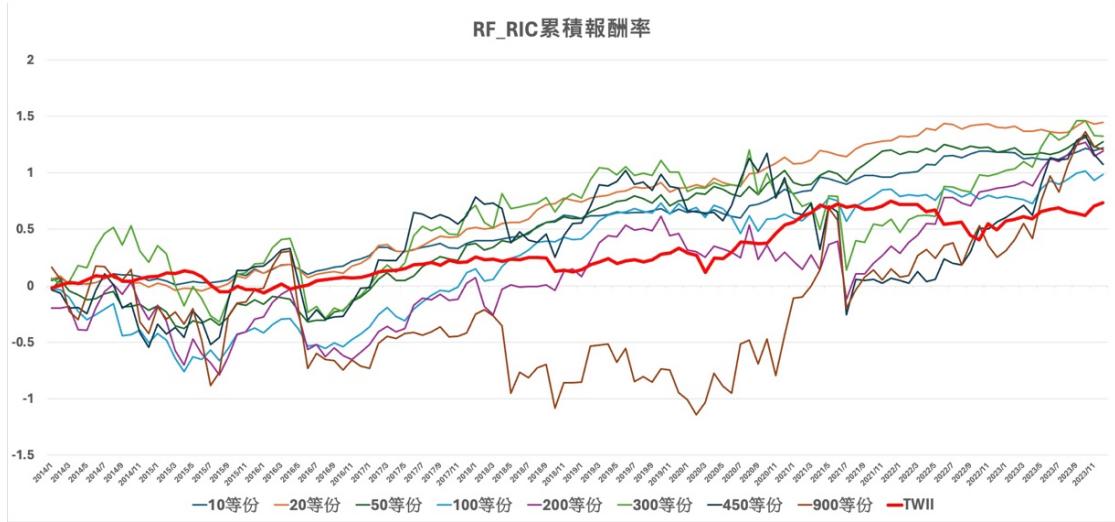


利用線性迴歸預測 RIC 值所建構投資組合之累計報酬率多數都能夠超越或與大盤持平，僅有以十等份所建構之投資組合稍微低於大盤。

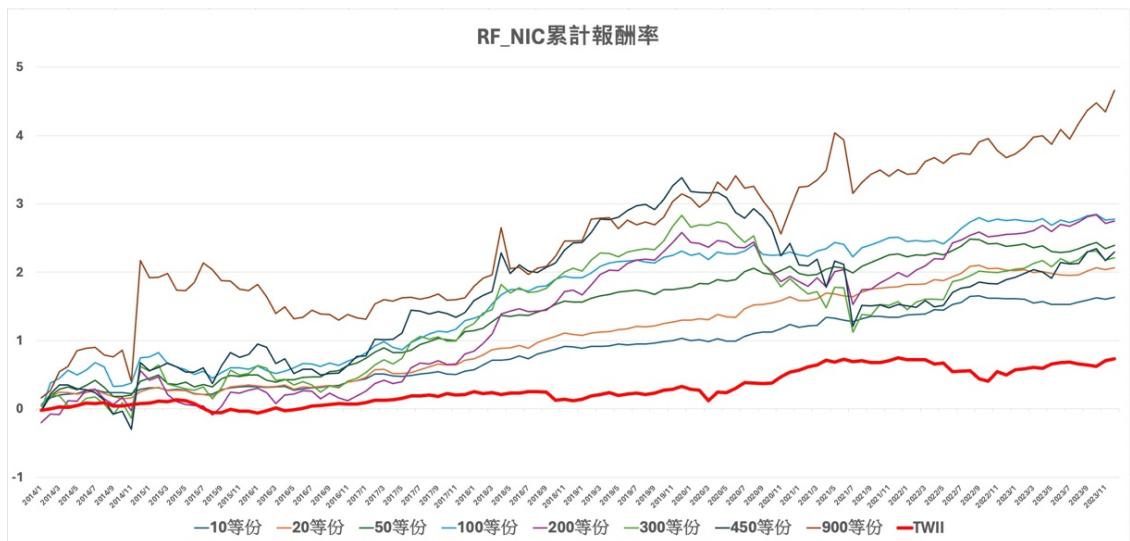


同樣的，利用線性迴歸預測 NIC 值所建構投資組合之累計報酬率多數都能夠超越或與大盤持平，其中又以以 450 等份建構之投資組合表現最好。僅有以十等份所建構之投資組合低於大盤。

隨機森林

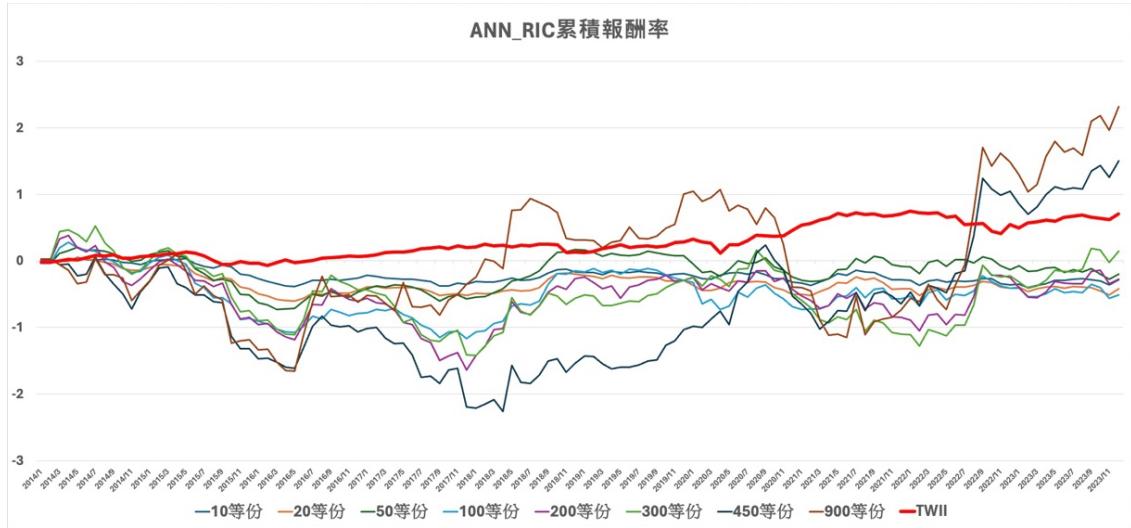


利用隨機森林預測 RIC 值所建構之投資組合累計報酬率雖然超出大盤的幅度不大，但是全都能超過大盤。

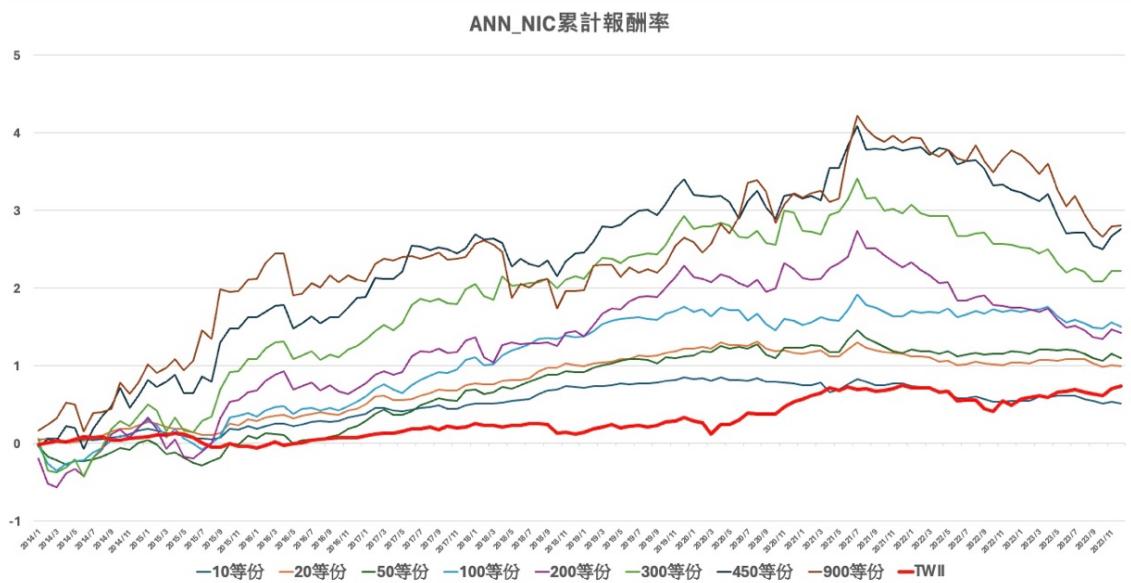


利用隨機森林預測 RIC 值所建構之投資組合累計報酬率也同樣的全都能夠超越大盤。值得注意的是各種等份之投資組合超出大盤都有一定的幅度。其中又以 900 等份建構之投資組合表現最好。

類神經網路



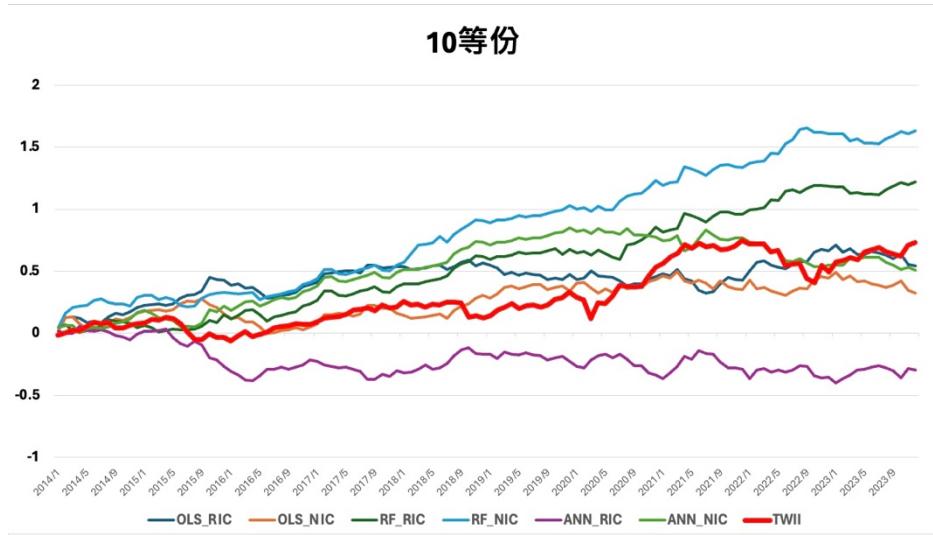
利用類神經網路預測 RIC 值所建構之投資組合累計報酬率相較之下表現並不突出，僅有 10 等份以及 900 等份所建構之投資組合能夠超越大盤，其餘皆落後大盤。



然而，利用類神經網路預測 NIC 值所建構之投資組合累計報酬率大部分都能夠超越大盤，僅有以十等份所建構之投資組合稍微低於大盤。

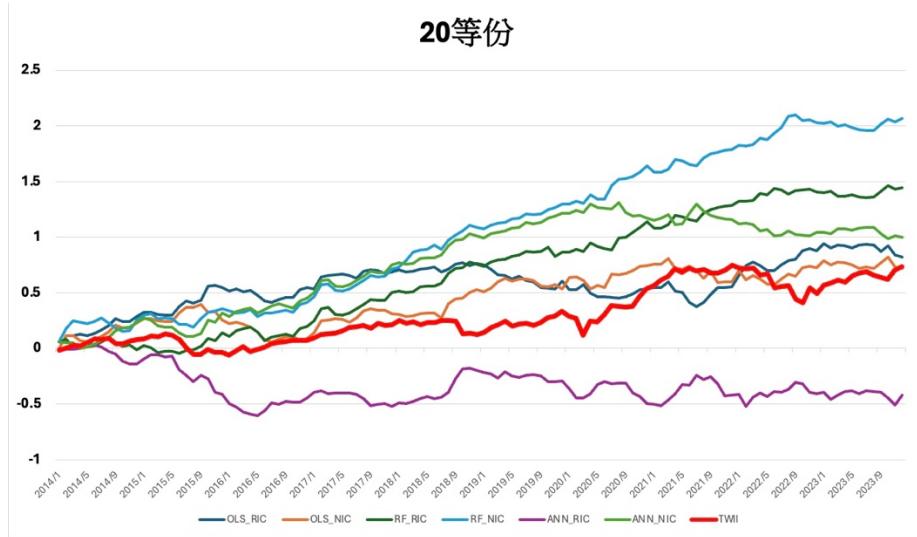
三、在不同等份之投資組合下三種機器學習方法的表現分析

10 等份



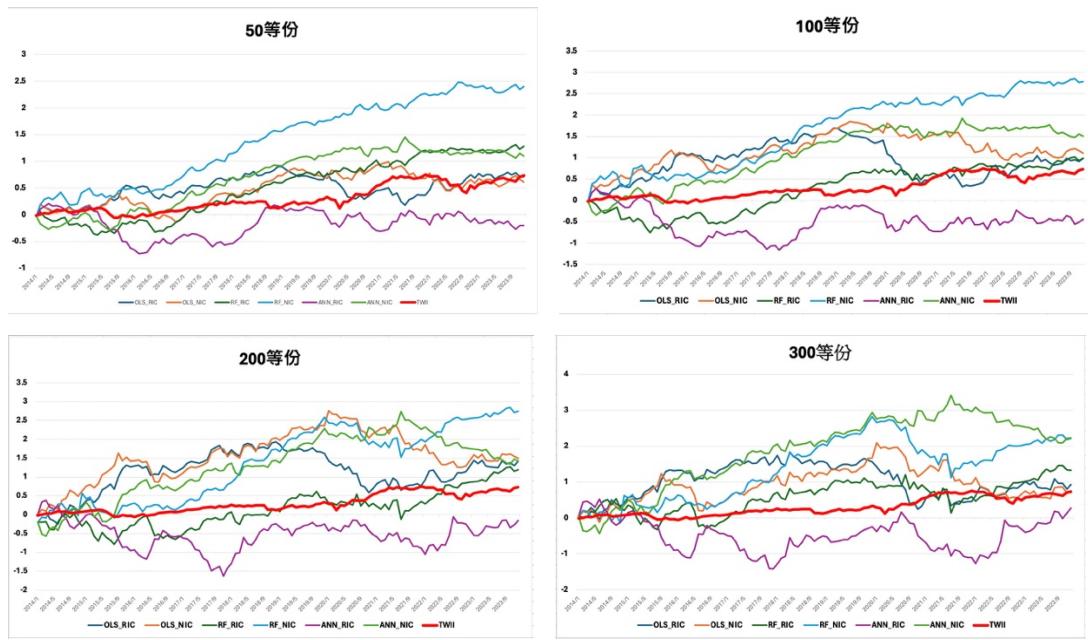
在 10 等份下，僅有隨機森林所建構之投資組合能夠超越大盤，其餘皆落後大盤。其中以預測 NIC 值所建構之投資組合表現最佳。

20 等份



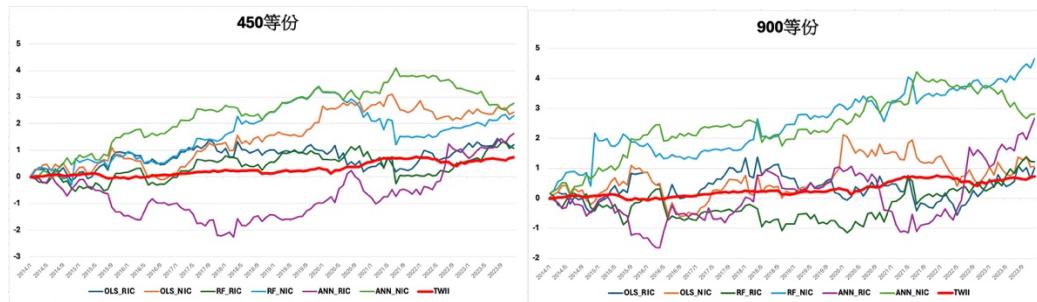
在 20 等份下，僅有類神經網路與側 RIC 值所建構之投資組合落後大盤，其餘皆能超越大盤。其中又以隨機森林預測 NIC 值所建構之投資組合表現最佳。

50 等份、100 等份、200 等份、300 等份



如同 20 等份，在 50 等份、100 等份、200 等份、300 等份下都是僅有類神經網路與側 RIC 值所建構之投資組合落後大盤，其餘皆能超越大盤。其中也都是以隨機森林預測 NIC 值所建構之投資組合表現最佳。

450 等份、900 等份



在 450 等份、900 等份下，所有投資組合和都能夠超越大盤。其中依舊以隨機森林所建構之投資組合表現最為突出。

捌、結論

在各個投資組合中，以隨機森林預測 NIC 值所建構之投資組合不但勝率全都能超過大盤，並且累計報酬率也都高出大盤許多。除此之外，各等份的投資組合相較於其他機器學習方法表現都比較好。因此，我認為用隨機森林以市值、帳面市值比、動能三種因子預測 Normal IC 值所建構之投資組合能夠很好的提升投資勝率以及穩定性。

玖、心得

心得一、

在這堂課中有很多內容都是我以前沒有接觸過的，包括資料的抓取、巨量資料的處理、機器學習的運用、投資組合的建構等等。因此在完成這份報告的過程中我從一開始懵懵懂懂地跟著大家的步調慢慢做，自己搜尋了許多資料，了解這些動作的理由，直到後來才漸漸進入狀況。起初，我連 Excel 的指令都不太熟悉，需要反覆聽助教講解這些公式的意思，直到多次的練習後才慢慢熟悉。後來到了程式撰寫的部分，我也都需要自己搜尋每列程式碼的功能，才能夠完全掌握程式碼的意義。雖然過程很緩慢，但我在過程中學習到了許多。因此，在此次課程當中的收穫除了課程本身之外，也有很大的一部分是在搜尋資料解決自己不懂的地方時所獲得的。

心得二、

我認為用線上課程的方式來進行這堂課程有很大的幫助，雖然線上課程可能讓人產生惰性，但這堂課主要利用 PPT 講解課程的原理以及帶著同學實際操作，我認為使用線上課利用投影的方式講解讓我滿容易就跟上課程的腳步。除此之外，課程也都會留下上課的影片，讓我也能夠反覆觀看先前上課的內容，這對我也很有幫助，我常常遇的不懂的地方就會找之前上課的片段再做複習。因此，我認為以線上課來進行這堂課對我來說是利大於弊。