台灣ETF價格預測競賽

設計文件

隊伍: 乾這個難

成員：白家誠，方彥鈞，陳勃文，林嵩儒，樊恩宇

# 摘要

[請簡單說明本次比賽所使用過之特徵、演算法以及訓練模型的方式]

本次比賽我們使用XGBoost的機器學習模型，XGBoost是一種gradient boosting regression tree，並且在預測股價及預測漲跌上我們使用不同的模型，在預測股價上，我們的模型做的是linear regression的問題，而在預測漲跌上，我們的模型是做binary regression，因為大部分ETF的漲跌都跟大盤一致，因此我們在預測漲跌時是預測大盤的漲跌，在預測股價時，我們則是對每隻ETF有個別的model來預測價格。在選取特徵上，我們使用了台股大盤的漲跌，美國股市指數漲跌（ex: DowJones指數, NASDAQ指數, S&P500指數, 費城半導體指數)，以及用台股大盤指數製作的技術指標(ex: KD線, B-bands等等)，台股及外資買賣超均線ETF個股漲跌作為特徵。

# 環境

[請說明本次比賽所使用的系統平台、程式語言、函式庫]

Platform: Ubuntu 16.04

Language: Python 3.5

Python package:

[機器學習相關]

matplotlib

numpy

pandas

xgboost

talib

Pickle

[爬蟲相關]

bs4

requests

lxml

Html5lib

Pandas\_datareader

fix\_yahoo\_finance

# 特徵

[請說明本次比賽所使用的特徵]

1. 美國股市指數漲跌（ex: DowJones指數, NASDAQ指數, S&P500指數, 費城半導體指數)（note:美股禮拜五收盤對下週台股指數影響頗大，因此加入美股當feature，可讓準確率提高）

2. 台股大盤漲跌

3. 台股大盤技術指標轉換 (ex: KD線, B-bands)

4. 台股大盤N日均線

5. 台股大盤外資買賣超均線 (取近N日平均）

6. 台股大盤多頭空頭走勢

7. ETF個股漲跌

資料來源：

美股: https://finance.yahoo.com/

台股大盤: http://www.twse.com.tw/zh/

爬蟲程式參考: https://www.finlab.tw/Python-%E6%99%82%E9%96%93%E5%BA%8F%E5%88%97%E5%AF%A6%E4%BD%9C%EF%BC%81/

# 訓練模型

[請說明本次比賽所使用的訓練模型、參數]

本次比賽我們使用XGBoost的套件，針對漲跌預測跟股價預測我們設計兩種不同的模型，

預測漲跌

xgb\_params = {

'eta': 0.15,

'max\_depth': 5,

'subsample': 0.8,

'colsample\_bytree': 0.7,

'objective': 'binary:logistic',

'eval\_metric': 'error',

'silent': 1

}

上面是我們在預測漲跌上所使用的參數，max\_depth、subsamplecol、sample\_bytree是用來避免overfit的參數，在預測漲跌時，我們把”漲”當成1，”跌”當成0，並以binary:logistic當objective function預測，另外我們觀察到ETF的漲跌跟大盤的漲跌大致上是一致，所以我們用上個section中的特徵1~6來預測大盤漲跌當作每個ETF的漲跌。

預測價格

xgb\_params = {

'eta': 0.15,

'max\_depth': 5,

'subsample': 0.8,

'colsample\_bytree': 0.7,

'objective': 'reg:linear',

'eval\_metric': 'mae',

'silent': 1

}

上面是我們在預測價格上所使用的參數，大部分的參數我們都跟預測漲跌一樣，但是兩個model我們使用不同的objective function，在預測價格時，我們以reg:linear當objective function作預測，預測的label為(當日價格/上禮拜五價格)，如此可以確保我們的模型不論在多空頭時，training時的label差異都不會太大。

因為ETF漲跌的幅度不一定與大盤一致，所以我們對每支ETF個別有一個模型，用上個section的特徵1-7做訓練，另外也有一個模型用來預測大盤漲跌的幅度，最後再用ETF模型與大盤模型預測出來的結果相加除二得到預測結果。

# 訓練方式及原始碼

[請說明本次比賽答案的產出方式並提供有效之原始碼(連結亦可)]

Model:

1. 漲跌

檔案：up\_rise\_example.py

2. 價格（建立兩個獨立model 做Ensemble Learning）

檔案: price\_example.py

檔案: etf\_regression.py

Output:

檔案: ETF\_output.py

輸出: output\_example.csv

Finance\_data\_example 資料夾:

檔案: 整理好的data

執行:

python3 main.py

連結: https://github.com/michaelm60710/ML-ETF-Prediction-Example/

# 結論

[請簡易說明本次競賽後所得的結論]

我們主要都是以預測大盤為主，理由是分析大盤與18個ETF股，相似度很高，普遍在70~90%左右（假設大盤漲/跌，ETF也會跟著一樣漲/跌的比例)，如果再加上考慮有時股價在整理，漲跌幅度非常小，幾乎可略過，其實70~90%的相似度是非常高的。且大盤相對比較穩，雜訊也比單純看個股還小很多，如果單純拿ETF和個股的資料來training，我們認為不可行，因為雜訊太多且資料太少。股市的影響因素非常多(ex:國際情勢，政治，投資者心理狀態等等)，只能免強的從統計學和機率的角度來分析，計算出未來大概是漲是跌的機率是多少。

Model部分我們不用主流的RNN,LSTM，因為用neural network太多參數和細節須考慮，而我們如果用XGBoost，我們只需花精力調整feature，讓feature有把過去和現在日期的資料都包含在一起，例如嘗試用EMA(Exponential Moving Average)和技術指標產生feature，或著再把這些資料取微分(取斜率）產生新的feature。

而觀察實驗結果時注意的有趣問題，發現我們的模型在上漲時的期間預測比較準確，但在下跌期間時卻比較弱，認為這跟股市近10年是大多頭也有點關係，造成Model訓練後變得比較”樂觀”。

預測一隻ETF未來的”隔日”是漲是跌其實是很困難的，未來可變因素實在太多（禮拜五整體股市是漲是跌影響分數最重，卻也是最難的部分 ），但比賽卻讓漲跌的分數影響全局。如果把評分重點擺在預測整體的價格趨勢是往上或往下，可能會比較好。