Python資料分析與機器學習應用 期中報告

組別：J

組員：吳仲皓、丁大洸、許時融、張景皓、鍾承恩

目錄：

壹、Project Overview

貳、Background

☐ 專案動機

不管是在股票或是加密貨幣的市場中，往往有著許多不同的指標，不管是單純的 K 線圖，各種技術分析指標，或是各種短中長期的量化交易策略，市場中都有許多量化的指標在影響著人們的交易決策。而我們認為在這之中，有一塊缺少的但也相當重要的資訊，那就是質性的資料，不管是知名投資 KOL 或是普通鄉民在網路針對投資相關話題發表的看法，都代表著市場上某部分的情緒，但這樣的資料相對於 OHLC 、交易量等資訊較難以數字的形式呈現，因此也是較少人關注的部分。

而再來則是市場的選擇，我們認為相對於成熟的傳統金融股票、債券、期貨市場，新興波動大的加密貨幣市場較可能因為這些市場的情緒而影響幣價，而甚至會因為一些非常個人行為層面的決定而對市場的情緒產生重大影響進而影響幣價的漲跌幅。最後幣圈相對於傳統金融有著波動大的性質，許多市場參與者都屬於投機交易者，平均交易頻率相較傳統股票市場高出許多，因此我們認為在這樣的市場中，掌握市場的情緒，是個有效判斷整體市場情況以及不同幣種短期可能走勢的好方法。例如在之錢，僅僅是因為馬斯克將推特的頭貼換成狗狗幣的圖片就讓狗夠幣的幣價上升近 40%，能清楚顯示出幣圈是個受市場短期情緒或是突發事件極大影響的交易市場。

我們意識到市場資訊可能很難蒐集，導致投資人無法有效到掌握市場對於加密貨幣的情緒。因此我們想藉此作品來幫助投資人解決此問題，除了即時了解市場情緒，更能利用機器學習來預測未來走勢。

☐ 作品的重要性

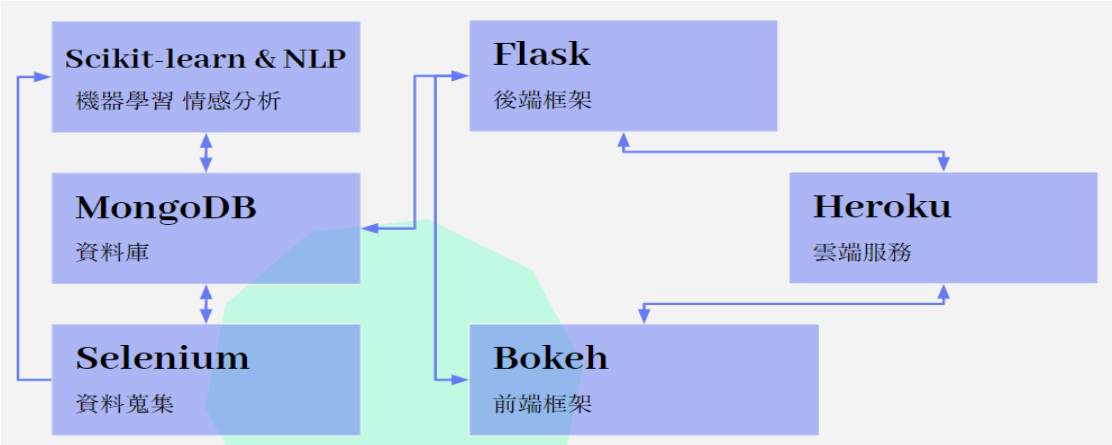
- 在相對不成熟的加密貨幣市場，市場消息往往是影響市場情緒的一大因素，而市場情緒是影響幣價的主要原因。
- 為使用者擷取並整理Twitter使用者對虛擬貨幣的情緒指標，讓使用者快速掌握虛擬貨幣價格的可能走勢。

☐ 作品帶給使用者的便利性

- 讓無法及時觀察市場的交易人，也能輕鬆得知市場的情緒指標，省去主動爬找資料與質化資料不好分析的問題！
- 開發出圖像化的 UIUX 介面，讓使用者可以輕易的查看想要尋找的資訊以及一目了然的看出加密貨幣市場上的情緒指標與各項資訊。

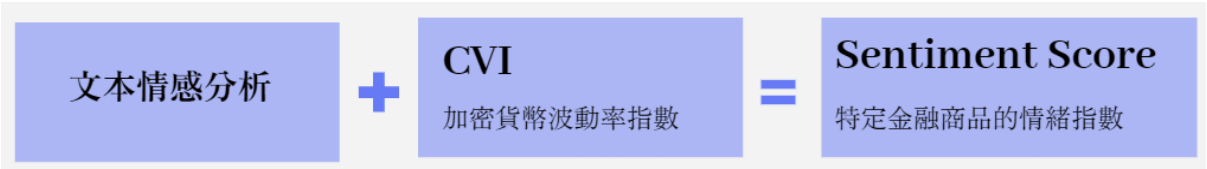
參、Target Dataset & Packages and Tools

利用twitter上的資料來分析虛擬貨幣市場的現今情緒，並利用Python爬蟲技術，擷取社群媒體Twitter對於”Bitcoin”、”乙太幣”等關鍵字，以觀察市場對於虛擬貨幣的市場情緒、社群媒體熱門話題、價格和交易量走勢等資訊，提供使用者做交易上的參考。



☐ NLP - twitter RoBERTa

- 透過 NLP - twitter RoBERTa 情緒分析模型，針對「社群媒體之貼文」、「討論串的標題」與「留言」進行情緒分數分析。
- 結合文本「情緒分數」分析結果與「加密貨幣波動率指數(CVI)」，產生針對特定金融商品的情緒指標(Sentiment Score)。



☐ Scikit-Learn

- 得到即時的「Sentiment Score」之後，我們結合「歷史的貨幣價格走勢」成為貨幣價格之特徵值，並且藉由使用 Scikit Learn 套件，得出預測之貨幣價格走勢。
- 再藉由使用多種學習模型以即調整參數之後，我們決定使用 KNN 模型來成為預測價格走勢之工具，而原因如下：

  1. 虛擬貨幣的漲跌可能是具有非線性的複雜關係，而KNN 在處理數據時因為對於數據的分佈並沒有假設，所以能更好地捕捉其變異

  2. KNN 在調參上較為簡便

  3. 在經過多個模型比較後，KNN 在測試資料中的表現最好，r-squared為最高的，代表這個模型對數據的擬合程度最高

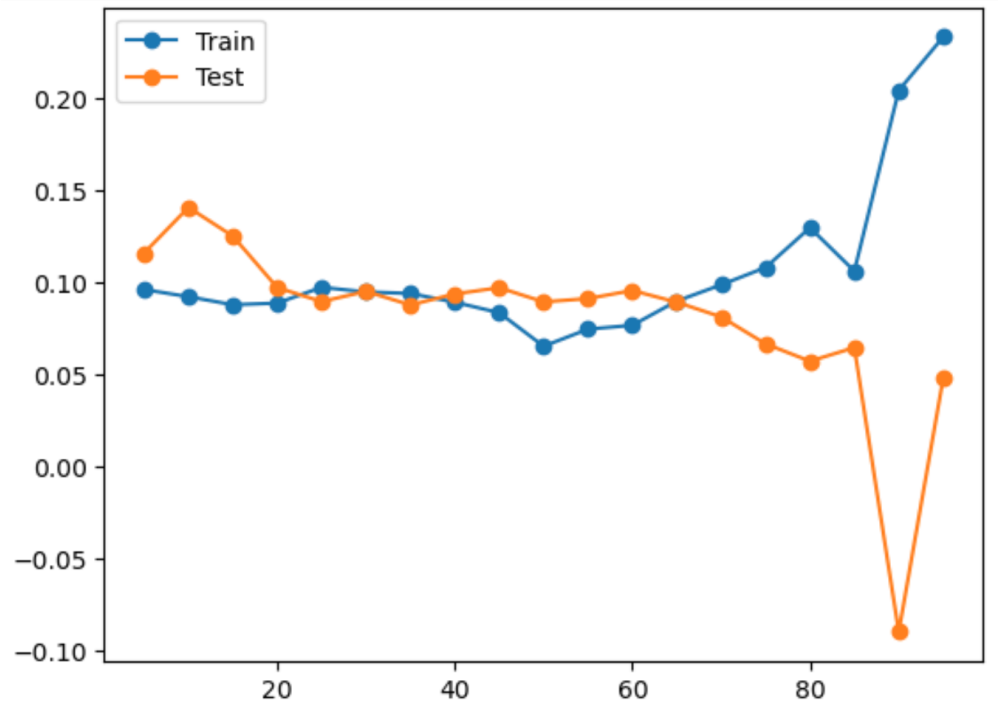
Target and Feature

我們想拿今日的資料來去預測隔日CVI的漲跌，因此Target為t+1期的cvi漲跌幅。而Feature則有t期的cvi漲跌幅、t-1期的cvi漲跌幅、t期的情感分數以及t期的收盤價。放入t-1期的資料是因為價格可能是具有較長的時間延遲效應，能讓我們更好預測t+1期的波動。

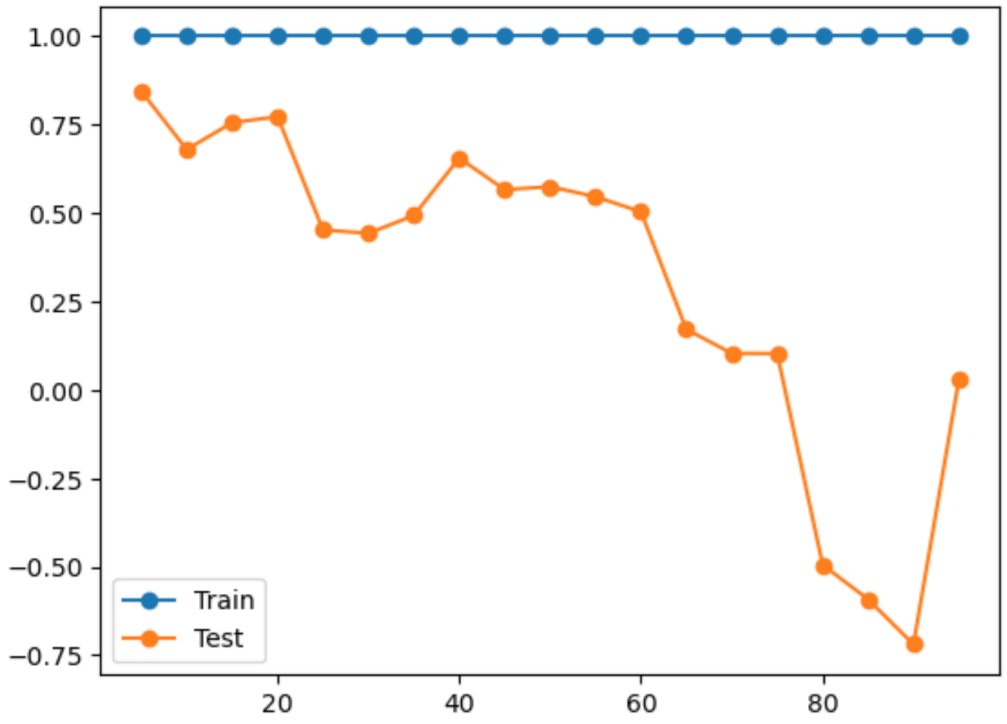
Chosing Model

從老師所介紹的模型中，我們把每個模型都拿來試看看，以下為一些結果，y軸為r^2的分數，x軸為測試資料的比例。

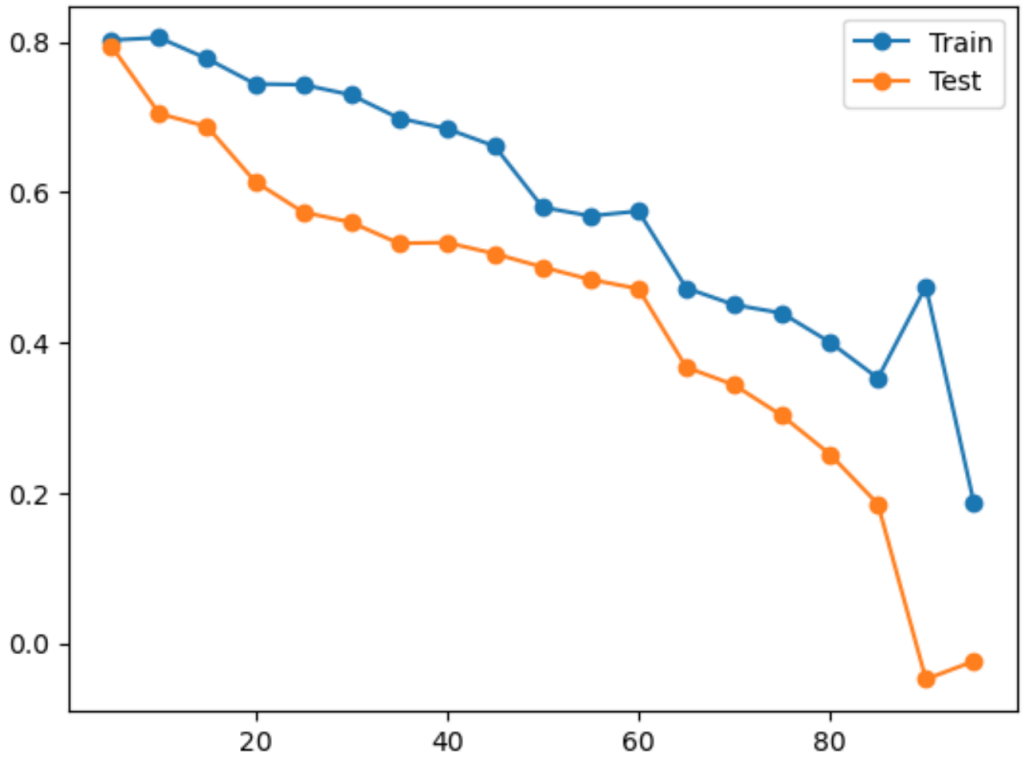
linear model



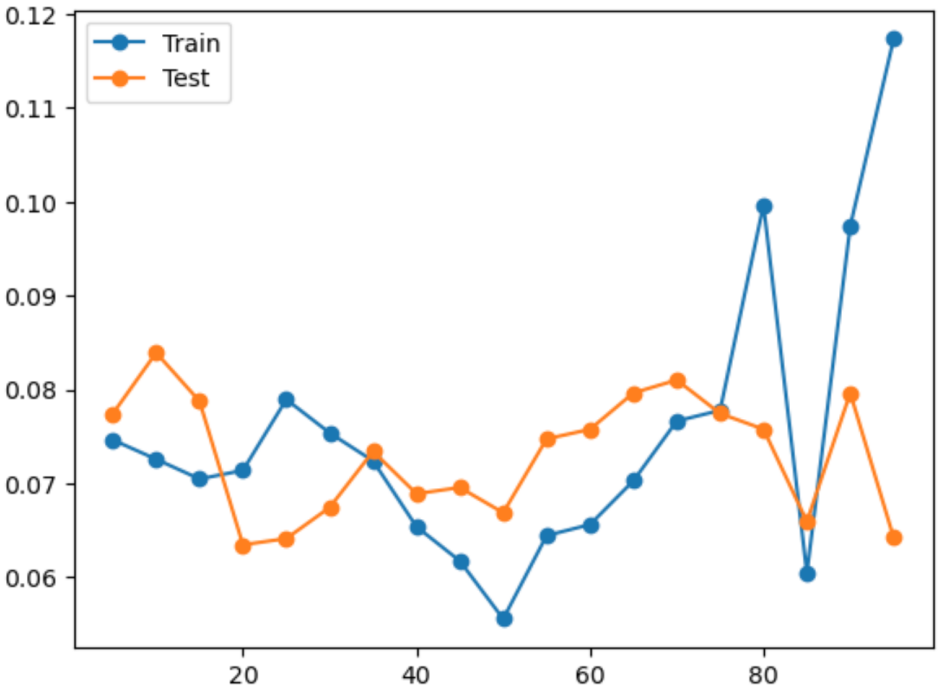
Decision Tree



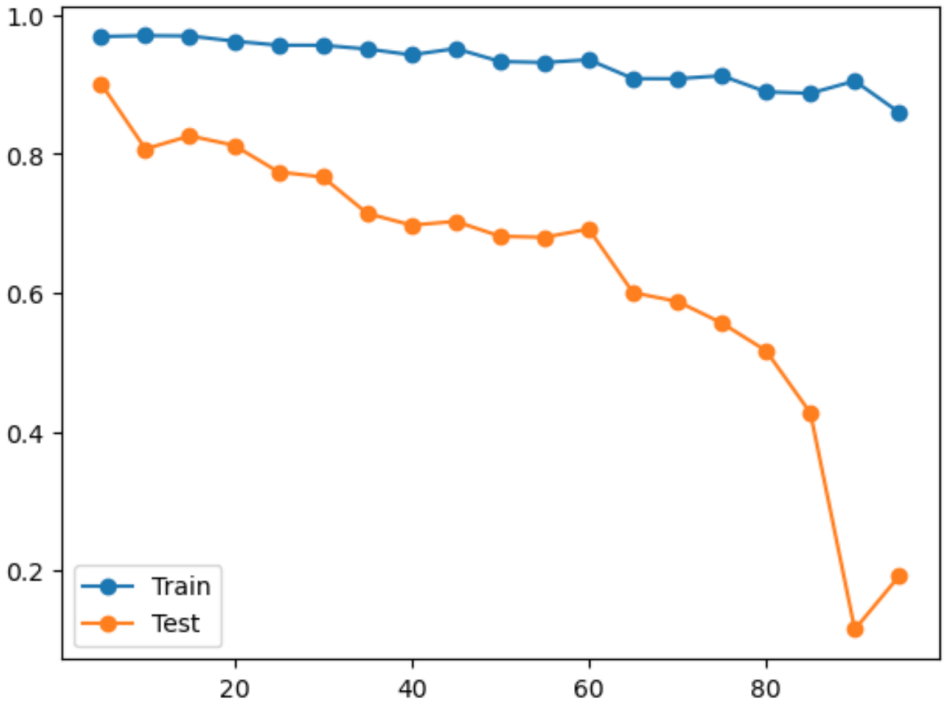
KNN



SVR



Random Forest



綜上來看，decision tree, KNN, random forest較能擬合資料，具有比較好預測的能力。

Training

挑出這三個模型decision tree, KNN, random forest來進行訓練，因為其在未調參的表現時就已不錯，在調完後應該會有更好的表現。調整時會著重於R^2的分數，我們會希望這個分數越高越好，因為這樣就代表我們這個模型較能解釋依變數的變異。

Decision Tree

```
for k in ratiovalues:
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = k/100, random_state=2023)
    for i in depthvalues:
        for j in leafvalues:
```

參數主要調整為depth value以及leaf value，這兩個參數對於模型的發展較有進步的空間，其中depth越深會更有過度擬合的風險，而葉子的節點數量越小也會有過度擬合的風險。
其結果為下圖：

```
best depth: 20 best min_sample_leaf: 2
Testing r2: 0.8683254269387181 testing mse: 6.2545055091925645
test_ratio: 5
```

KNN

```
for k in ratiovalues:
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = k/100, random_state=2023)
    print(k)
    for j in neighborvalues:
        for i in weights_value:
            for a in algorithm_value:
                for t in p_value:
```

參數主要調的是neighbor，若此值越小，代表所選擇的鄰居越少，若太小的話會有過度擬合的問題，而剩下的參數調整對於模型的進步較沒有進步的空間，但還是調看看。
其結果為下圖：

```
best n_neighbors: 4 best weights: distance best algorithm: ball_tree best p: 1
Testing r2: 0.9445219469039781 testing mse: 2.6351920546334853
test_ratio: 5
```

Random Forest

```
for k in ratiovalues:
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = k/100, random_state=2023)
    print(k)
    for j in depthvalues:
        for i in estimatorvalue:
```

參數的調整主要為depth跟estimator，depth越深的話會有過度擬合的問題，而estimator則是能增加樹的量，若越多的樹則一樣會有過度擬合的問題。

其結果為下圖:

```
best n_estimators: 130 best max_depth: 18
Testing r2: 0.9028250055240712 testing mse: 4.61578514496181
test_ratio: 5
```

所以從結果來看，我們應選擇knn的模型來做為我們預測模型的工具，應其有較高的R^2值，但使用上，我們仍需注意過度擬合的問題，礙於我們資料量少，沒有更多能用的測試資料，較無法看出其在預測未來時的效能為何。

☐ Selenium

- 取得CVI(加密貨幣波動率指數)資料

```python
def getWordCloud():
    path = os.path.join(os.path.dirname(__file__), 'ML_Sentimental/Crawler/twitter_data_new.csv')
    df = pd.read_csv(path)

    stop_words = STOPWORDS.update(["https", "co", "RT", "reddit", "bitcoin", "WWW", "com", "message",
    # twitter
    wordcloud = WordCloud(stopwords = stop_words).generate(''.join([i for i in df['text']]))

    plt.imshow(wordcloud, interpolation='bilinear')
    plt.axis("off")

    img_path = os.path.join(os.path.dirname(__file__), "example1.png")
    plt.savefig(img_path)
    encode = img_to_base64(img_path)
    return encode

def img_to_base64(img_path):
    with open(img_path, 'rb') as f:
        img = f.read()      You, 13 minutes ago • add backend and requirements.txt …
        img_base64 = base64.b64encode(img).decode('utf-8')
    return img_base64

if __name__ == '__main__':
    print(getWordCloud())
```

☐ Wordcloud

- 對當日的與比特幣相關之推特貼文做斷詞斷句後繪製成圖，方便讀者掌握當天最多的關鍵字

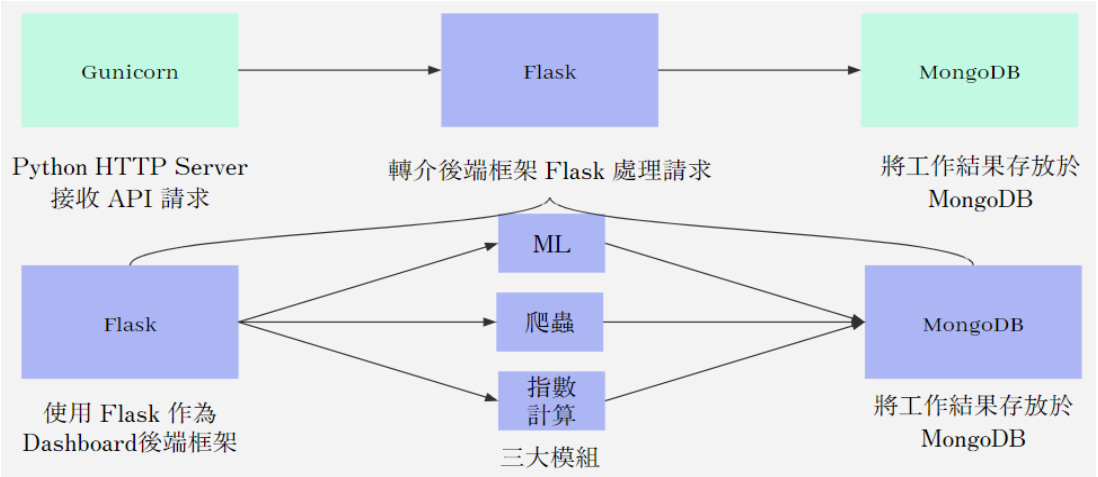
```python
def getWordCloud():
    path = os.path.join(os.path.dirname(__file__), 'ML_Sentimental/Crawler/twitter_data_new.csv')
    df = pd.read_csv(path)

    stop_words = STOPWORDS.update(["https", "co", "RT", "reddit", "bitcoin", "WWW", "com", "message",
    # twitter
    wordcloud = WordCloud(stopwords = stop_words).generate(''.join([i for i in df['text']]))

    plt.imshow(wordcloud, interpolation='bilinear')
    plt.axis("off")

    img_path = os.path.join(os.path.dirname(__file__), "example1.png")
    plt.savefig(img_path)
    encode = img_to_base64(img_path)
    return encode

def img_to_base64(img_path):
    with open(img_path, 'rb') as f:
        img = f.read()      You, 13 minutes ago • add backend and requirements.txt …
        img_base64 = base64.b64encode(img).decode('utf-8')
    return img_base64

if __name__ == '__main__':
    print(getWordCloud())
```

☐ Pandas、NumPy

- 清洗虛擬貨幣歷史價格、歷史CVI指數... 等資料

肆、System Structure and Modules

☐ 後端 - Flask



Step 1 : 對推特發送api請求，並且接收到api請求後的處理

Step 2 : 串接API後，藉由"查詢Twitter資料"取得有關比特幣的貼文內容

☐ 前端 - Bokeh

此次專案的前端全部由 Bokeh 這個是覺化套件完成，這個套件提供了強大的視覺化及互動式工具，因此使用起來也較為方便。此次的成果主要分成四個區塊 :

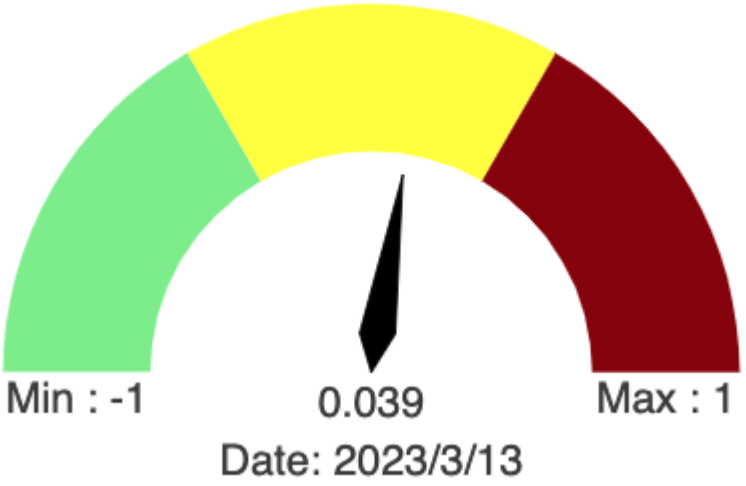
1. Wordcloud :
   此部分為透過與後端 API 互動回傳一個根據當天最新資料產生的 Base64 字串，在經過 Python 內建套件的轉換，將圖片存成 PNG 方式在 database 中儲存，最後由套件將 png 圖片轉為 rgba 並且用 bokeh 的 image_rgba 將圖片呈現在

前端中。

(3.) Wordcloud
What are the keywords that appear on Twitter most frequently in the past 24 hours?

2. Sentiment Score :
此部分透過與後端 API 互動取得最新的市場情緒指標，再透過事先設定好的不同市場情緒區段定義出當天透過推特的數據判斷出為過熱、適中、或是偏冷。並在前端中以圓弧圖及指針表示。

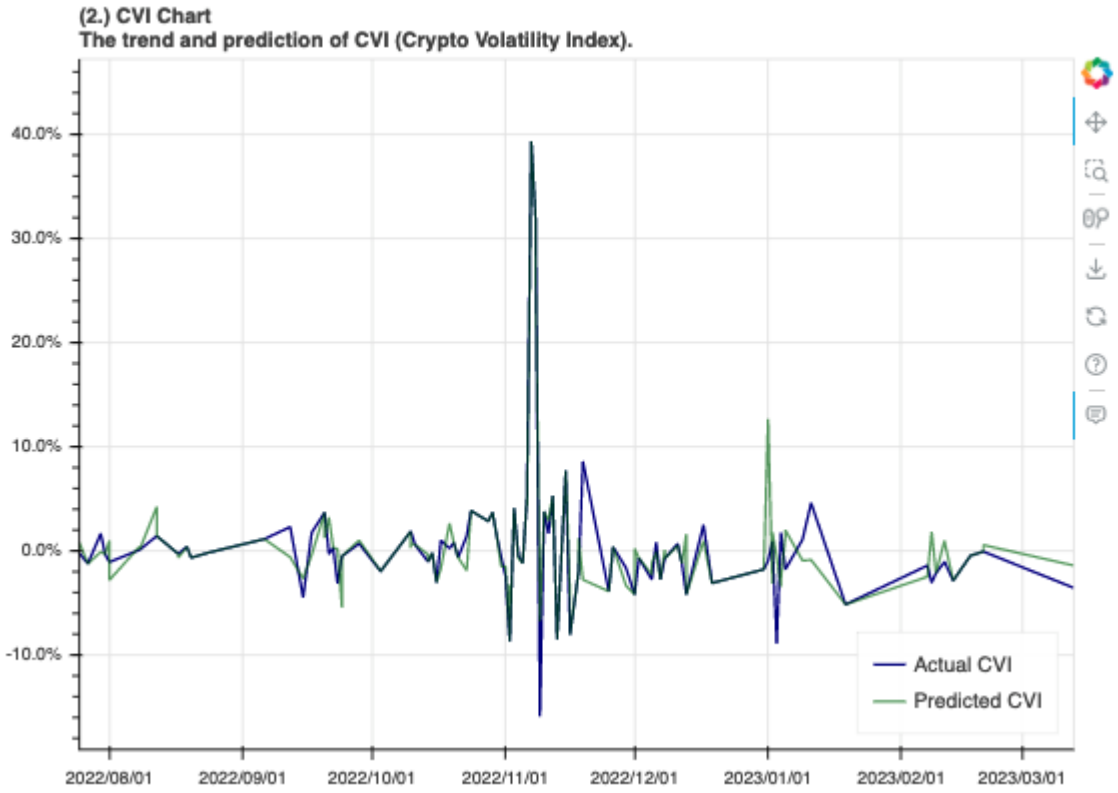
(4.) Sentiment Score
What is the sentiment score in the past 24 hours?



3. Price Chart :
此部分為提供使用者在查看市場預期波動率及情緒指標時能夠配合市場最新的價格走勢一同查看，因此我們在這個區塊提供了目前市場市值前十大的幣種的歷史 k 現價格圖，搜集了從 20200101~使用者使用的時間的最新市場 15 分鐘數據，並提供不同的時間區段供使用者選擇，包括 15分鐘、半小時、一小時、四小時、一天、一週、一個月等不同的時間間隔。而提供的幣種涵蓋 BTC, ETH, BNB, TRX, DOGE, SHIB, MATIC, SOL, XRP, ADA。
此段程式也會持續透過與 Binance API 對接持續更新最新的市場價格資訊，同

時也會在圖片中顯示成交量的資訊。

4. CVI Chart :
   此部分為透過呼叫透端的 API, 取得歷史全部實際及我們透過機器學習預測出的 CVI 指標，代表我們對這天市場價格波動性的預測。再透過 bokeh 將兩條線各自畫在前端上，並且可以提供查詢歷史不同時期的 CVI 實際及預測值。



伍、Total Evaluation

☐ 原計畫：

- 專案四月初開始動工，執行順序為資料搜集及前處理，模型建制、訓練、優化，前後端工程及最後的整合。預計在五月底時完成。

☐ 執行結果：

- 資料搜集及前處理，模型建制、訓練、優化，前後端工程及最後的整合等程式皆按時完成。
- 專案於5/30日完成。
- 除了完成原來的計畫(只有做到情緒分數)，我們又更額外的使用模型來預測未來 **CVI** 走勢。

☐ 在設立時遇到的困難 - **Twitter API** 改為收費制

- 無法取得 2023/03/14 日之後之市場動向以及情緒指標，導致 Wordcloud, sentiment score, 以及歷史sentiment score停止更新
- 只能以 2019/12/11 ~ 2023/03/13日之資料訓練模型
- 只要能付費解鎖Twitter API，以上功能將能持續更新，為使用者提供最完整的市場資訊

陸、Teamwork Assignments

☐ **Data Processing & Model training**

B10703024吳仲皓、B09305043 許時融、B09602062 鍾承恩

☐ **Front-Backend**

B09704076 張景皓、B09703074 丁大洸

柒、Feelings

- B10703024 財金二 吳仲皓：

這次的機器學習期末報告讓我對市場情緒對加密貨幣價格的影響有了更深入的研究。我們通過使用Twitter API截取市場對加密貨幣的情緒數據，並結合情緒指標和歷史CVI指數，來預測未來一天的CVI指數。我主要用到的工具包括上課教到的pandas、NumPy、以及scikit-learn。而我的主要收穫有三。

首先，這次研究讓我深刻理解到市場情緒對加密貨幣價格的重要性。除了關注最新事件，更要了解市場對這些事件的反應和看法，以制定更明智的投資策略。情緒指標的應用在預測市場趨勢方面具有潛在價值，並可能為交易者和投資者提供有益的參考。

其次，這次的研究讓我意識到有效的團隊合作和溝通對研究成功至關重要。要把自己產出的東西讓隊友使用並不是一個容易的工作，但這次經歷讓我學到了如何與隊友協調，讓合作創造更大的價值。同時我也透過跟隊友的合作了解更多套件，像是Bert、Flask、selenium等，還有更熟悉前後端的應用、假設、以及融合。

最後，就是我們這次的報告遇到了Twitter api突然要收費的困境。但我相信儘管我們無法獲取到最新的數據，我們的研究仍為加密貨幣領域的未來發展提供了一個基礎。這次經驗也讓我明白到研究不僅僅是結果，更是一個學習和成長的過程。

總結而言，這次的學習經驗讓我對市場情緒對加密貨幣價格的影響有了更深入的理解。我通過研究和合作與小組成員的互動，提高了我的研究能力和團隊合作技能。這次研究雖然有些意外插曲，但我相信這次的研究對未來的學習和發展將有所啟發。

- B09703074 財金三 丁大洸：

在這個專案中，我們的目標是利用Twitter上的資料來分析虛擬貨幣市場的情緒，並利用機器學習的技術來預測虛擬貨幣的價格走勢。我們使用了Python作為主要的程式語言，並使用了各種相關的工具和套件，如NLP情緒分析模型、Scikit-Learn、Selenium、Wordcloud、Pandas和NumPy等。

此次專案中我主要負責前端的工作，也很感謝這次專案讓我更熟悉 Bokeh 這個套件，也在跟組員開會討論的過程，逐步發現怎麼樣更好使用，以及更好的呈現資訊。整個專案中讓我最印象深刻及學習的就是學習建構 K 線圖，要怎麼樣讓前端不會一次塞太多資料導致卡卡的，怎麼樣分配呼叫來的資料以及讀取都是很重要的功課。
最後則是感謝這個專案讓我們組員更熟習怎麼使用 Github 進行專案管理，在每一次開會討論 merge branch 的過程，都收穫滿滿。

最後雖然最後的專案中我不是主要負責機器學習的部分，但藉由觀摩同學的程式碼以及討論的過程，也讓我比起學期初更加熟悉 Python 中的機器學習套件，算是這堂課最大的收穫之一。

- B09704076 國企三 張景皓 ：

在這個專案中，我負責後端系統的部分，也很感謝這次專案讓我更熟悉 Python 中的相關套件，如 Scikit-Learn、Pandas 和 NumPy 等。我們的目標是利用 Twitter 上的資料來分析虛擬貨幣市場的情緒，並利用機器學習的技術來預測虛擬貨幣的價格走勢。

在專案中，我們使用了各種相關的工具和套件，如 NLP 情緒分析模型、Selenium 和 Wordcloud 等。在使用這些工具和套件的過程中，我學習到了如何使用 Python 中的機器學習套件，這是我這堂課最大的收穫。

最後，雖然我不是主要負責機器學習的部分，但在觀摩同學的程式碼和討論的過程中，我了解了如何使用 Python 中的機器學習套件，這對我在未來的學習和工作中都很有幫助。感謝教授與這堂課讓我們有機會實行這次專案，讓我學到了很多新知識和技能。

- B09602062 生工三 鍾承恩：

這次的期末報告關於虛擬貨幣走勢分析的主題對我而言比較陌生，對於Python的接觸經驗較少，透過這次報告使我更認識Python在金融和虛擬貨幣上的應用，也對Python如何從社群軟體擷取相關的關鍵字進行市場情緒分析有更深的了解，讓我知道了更多的未來延伸應用，比如:輿情掌握、網路討論度分析等，同時也認識到很多本次使用到的

Flask, Selenium, Wordcloud, Bokeh等Python套件, 很感謝組員的規劃與合作, 克服過程中遇到的困難, 而這次專題報告引起自己對Python更大的興趣, 學期結束後應持續學習, 累積相關的程式經驗。

- B09305043 經濟二 許時融：

這次期末報告我主要負責的是資料處理以及機器學習的部分。原先對於資料處理完全不熟悉, 還是第一次以python來做, 非常感謝組員願意包容我慢慢來, 在機器學習的方面上, 一開始調參時完全不知該如何調, 幸好chat gpt幫助我了解各個參數的意義, 以及可以怎麼調, 讓我度過難關。我也很慶幸我有把統計學好, 對於挑選變數時有更多的想法, 以及對於結果及演算法能更正確地理解其意義。而我也深知我對於機器學習還有更多能學習的地方, 希望有朝一日我能再繼續學習相關課程。