Machine Learning 2021 FALL

G-Research Crypto Forecasting

組別：第11組

組員：

R10921098 楊仁傑

R10921057 鄭凱鴻

B06505003 陳祐融

Outline

- Recap of Problem 問題回顧
- Introduction 理論介紹
- Data Preprocessing 資料預處理
- Model Structure 模型架構
- Experiment & Discussion 實驗討論
- Conclusion 結論
- Future work 未來展望

Outline

- Recap of Problem 問題回顧
- Introduction 理論介紹
- Data Preprocessing 資料預處理
- Model Structure 模型架構
- Experiment & Discussion 實驗討論
- Conclusion 結論
- Future work 未來展望

Recap of Problem

- 隨著**加密貨幣的重要性日益增加**，其交易量也快速上升，如果我們能精準預測其未來的走勢並以此作為投資與否的標準就可以在**最小化風險的情況下最大化收益**。
- 由於在**官方規定的虛擬環境下無法以其指標做為評估模型的標準**，我們在經過討論以後決定在期末報告上調整題目為**預測加幣貨幣的走勢關係**。

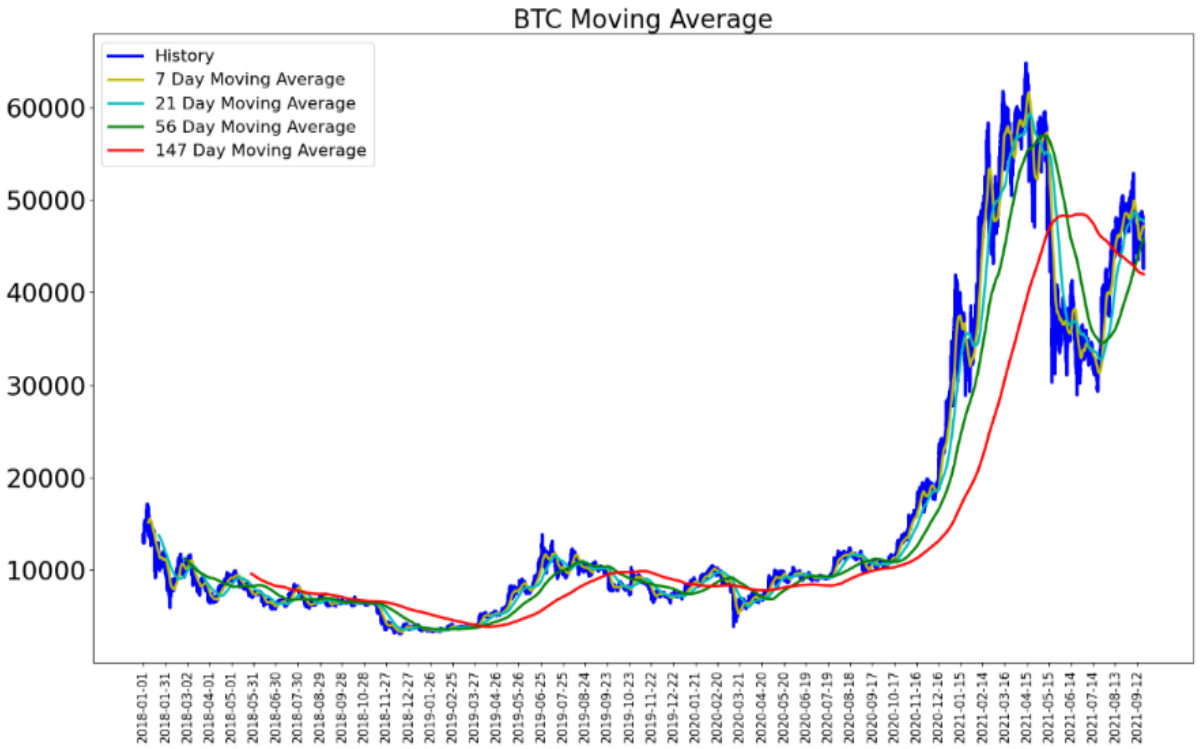
Outline

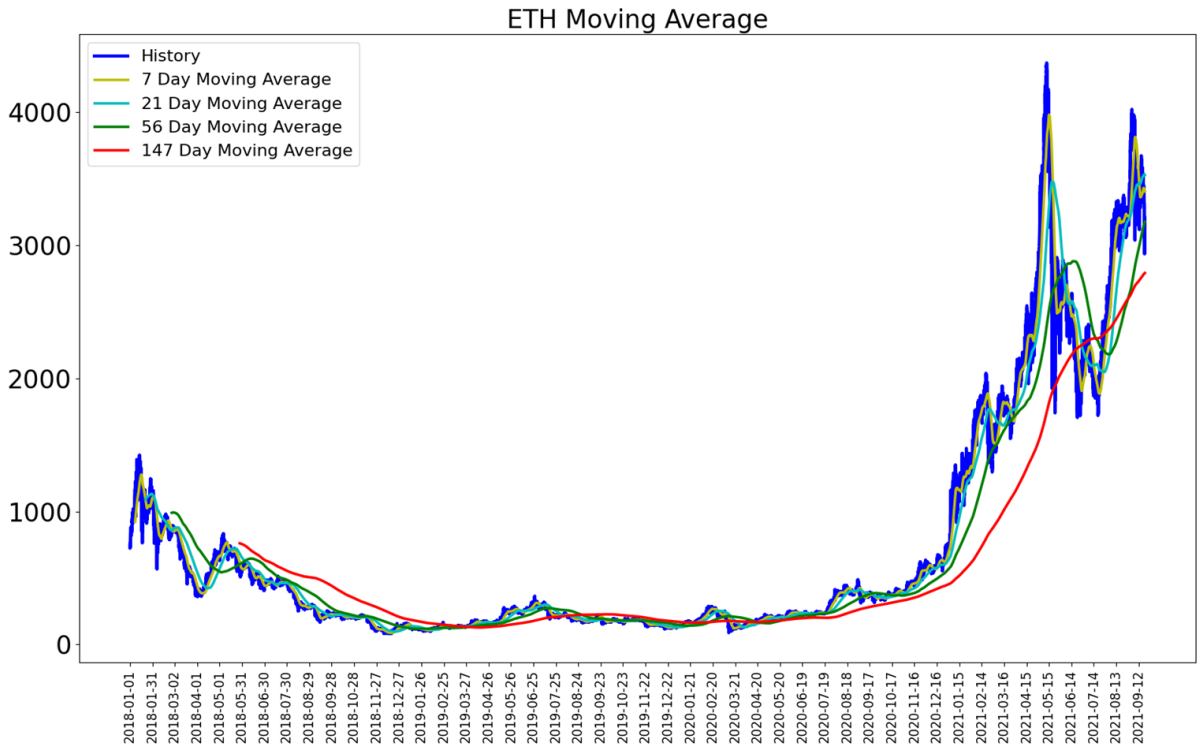
- Recap of Problem 問題回顧
- Introduction 理論介紹
- Data Preprocessing 資料預處理
- Model Structure 模型架構
- Experiment & Discussion 實驗討論
- Conclusion 結論
- Future work 未來展望

Moving Average (MA，移動平均線)

- 定義：MA在技術分析中常常被用來輔助觀察對趨勢的判斷，其將一段時期內的股票價格平均值連成曲線以代表過去一段時間裡的平均成交價格。
- 目的：判斷與預測市場當前及未來可能的走勢
- 計算方式：將N個時段的收盤價加總後取平均，作為第N個時段的值
- 缺點：落後指標、進場訊號較模糊、不適用上下行情整理的盤型

Moving Average (MA，移動平均線)





Recurrent Neural Network(RNN, 循環神經網路)

- RNN是一種以序列數據為輸入資料且所有節點依照其順序連接與傳遞的神經網路。在進行預測（或回歸）時，**不僅要考慮到當前的輸入，還要考慮上一個時刻的輸入**，與之前提到的MA有類似的概念。
- **缺點：只能有一種記憶疊加的方式**
- 我們在本次報告中會使用Long Short-Term Memory networks (LSTM, 長短期記憶網路)，其主要特色是能依使用者決定是否紀錄當前信息，較能運用在「長時間記憶」的任務上。

Outline

- Recap of Problem 問題回顧
- Introduction 理論介紹
- Data Preprocessing 資料預處理
- Model Structure 模型架構
- Experiment & Discussion 實驗討論
- Conclusion 結論
- Future work 未來展望

題目微調

- **在Kaggle上遇到的問題：**

  由於用來測試的資料(2021/06/01-2021/09/21)已經包含在給定的檔案中，所以只要將這部分的資料上傳正確率就可以達到0.999，因此沒辦法但從分數判斷模型的表現。

- **我們的作法：**

  1. 依照Train 70%, Valid 15%, Test 15%的比例切割，這樣就沒有測試資料包含在訓練資料中的問題。
  2. 由於不用在Kaggle上比準確率，所以為了簡化分析，我們將問題簡化為：

只看預測收盤價與歷史收盤價的MAE、RMSE

Data Preprocessing

- 由於從Kaggle提供的train.csv中的資料是將所有的幣種的資料放在一起，而每個幣種的價格區間不同，因此我們先將各個幣種的資料分開，得到右表描述的資料集。

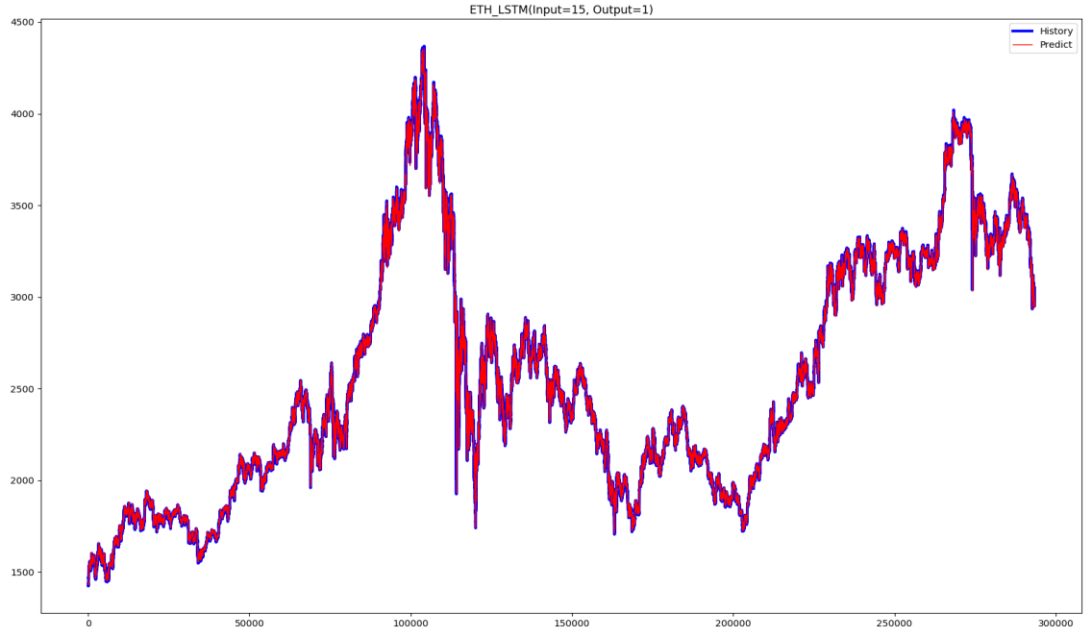
	timestamp	Asset_ID	Count	Open	High	Low	Close	Volume	VWAP	Target
0	1514764860	2	40.0	2376.580000	2399.5000	2357.1400	2374.590000	19.233005	2373.116392	-0.004218
1	1514764860	0	5.0	8.530000	8.5300	8.5300	8.530000	78.380000	8.530000	-0.014399
2	1514764860	1	229.0	13835.194000	14013.8000	13666.1100	13850.176000	31.550062	13827.062093	-0.014643
3	1514764860	5	32.0	7.659600	7.6596	7.6567	7.657600	6626.713370	7.657713	-0.013922
4	1514764860	7	5.0	25.920000	25.9200	25.8740	25.877000	121.087310	25.891363	-0.008264
5	1514764860	6	173.0	738.302500	746.0000	732.5100	738.507500	335.987856	738.839291	-0.004809
6	1514764860	9	167.0	225.330000	227.7800	222.9800	225.206667	411.896642	225.197944	-0.009791
7	1514764860	11	7.0	329.090000	329.8800	329.0900	329.460000	6.635710	329.454118	NaN
8	1514764920	2	53.0	2374.553333	2400.9000	2354.2000	2372.286667	24.050259	2371.434498	-0.004079
9	1514764920	0	7.0	8.530000	8.5300	8.5145	8.514500	71.390000	8.520215	-0.015875

➡

Asset_ID	Asset_Name	Asset_Shape
0	**Binance Coin (BNB)**	(1942619, 10)
1	**Bitcoin (BTC)**	(1956282, 10)
2	Bitcoin Cash	(1953537, 10)
3	**Cardano(ADA)**	(1791867, 10)
4	Dogecoin	(1156866, 10)
5	EOS.IO	(1955140, 10)
6	**Ethereum (ETH)**	(1956200, 10)
7	Ethereum Classic	(1951127, 10)
8	IOTA	(1592071, 10)
9	Litecoin	(1956030, 10)
10	Maker	(670497, 10)
11	Monero	(1701261, 10)
12	Stellar	(1778749, 10)
13	TRON	(1942619, 10)

Data Preprocessing

- 接著，我們使用pandas根據Asset_ID提取BTC、 ETH、BNB和ADA四種加密貨幣作為此次的測試。
- 下圖為利用LSTM模型對BTC與ETC分別做預測的結果，其中紅線代表預測資料、藍線代表實際資料，可以看到將不同幣種單獨預測的結果還不錯。



Data Preprocessing

- **Dataset Split 劃分資料集**：將原本的訓練資料依照Train: Valid: Test = 0.7: 0.15: 0.15的比例進行分割，這三者分別對應到模型訓練、選擇最佳驗證模型和模型效能評估。

- **MinMaxScaler 最大最小標準化**：我們藉由以下公式將資料等比例縮放到 [0, 1] 區間中以減少收斂所需時間並使不同特徵值有相近程度的貢獻。

$$X_{nom} = \frac{X - X_{min}}{X_{max} - X_{min}} \in [0, 1]$$

Outline

- Recap of Problem 問題回顧
- Introduction 理論介紹
- Data Preprocessing 資料預處理
- Model Structure 模型架構
- Experiment & Discussion 實驗討論
- Conclusion 結論
- Future work 未來展望

RNN, LSTM 架構

- 我們採用類似移動平均概念以利用前m天的收盤價來預測後n天的價格，此處n>1，表示往後預測n天並加總取平均作為當下的預測結果。
- 在兩個模型中，設置了4層網絡，第一層、第二層為RNN/LSTM層(維度；64)，第三層為RNN/LSTM層(維度；32)，第四層為Dropout層(0.01，用於防止過擬合)，第四層為全連接層(神經元個數為n，用於預測未來n筆資料)
- optimizer使用Adam，lr_rate = 0.01, epoch = 50, batch_size = 1024。

```
# create model
model = Sequential()
model.add(LSTM(64, return_sequences=True, input_shape=(x_train.shape[1:])))
model.add(LSTM(64, return_sequences=True))
model.add(LSTM(32))
model.add(Dropout(0.1))
model.add(Dense(output))

model.compile(optimizer=Adam(learning_rate=0.01), loss='mae', metrics=['acc'])
```

RNN, LSTM 架構

- 此外，我們也設定了兩個回呼函數：
- 一是只儲存最佳val_loss的模型作為使用，此設定可以幫助我們使用最佳val_loss的訓練模型
- 二是設定動態學習率，若當在訓練過程中模型經過三個回合都沒有取得更好的val_loss，則就會將學習率縮小0.1倍，最小值為1e-7。

```
callBack = [ModelCheckpoint(f'./model/BTC_LSTM(Input={time_stamp}, Output={output}).h5',
                            monitor='val_loss', verbose=0, save_best_only=True),
            ReduceLROnPlateau(monitor='val_loss', patience=3, factor=0.1, min_lr=1e-7)]
```

Outline

- Recap of Problem 問題回顧
- Introduction 理論介紹
- Data Preprocessing 資料預處理
- Model Structure 模型架構
- Experiment & Discussion 實驗討論
- Conclusion 結論
- Future work 未來展望

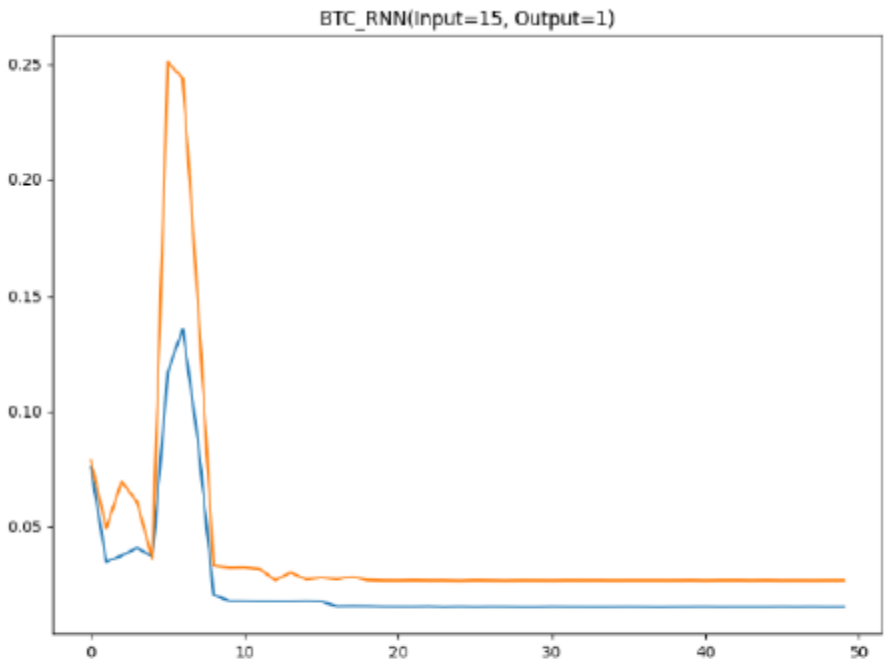
Comparison of RNN, LSTM

- 使用了相同資料(BTC)、相同架構但是分別為RNN跟LSTM的模型做為訓練，將m設定為15，n設定為1，並且經過50論訓練。
- 訓練結果如右表所示，可以看到不管是在每回合平均的訓練時間，MAE, RMSE上，RNN的表現結果比起LSTM而言都非常的差。

BTC(15/1)	each_epoch time(s)	MAE	RMSE
RNN	152.49	939.009	1154.830
LSTM	22.88	75.676	104.557

表 6.1 不同模型的訓練結果

Comparison of RNN, LSTM



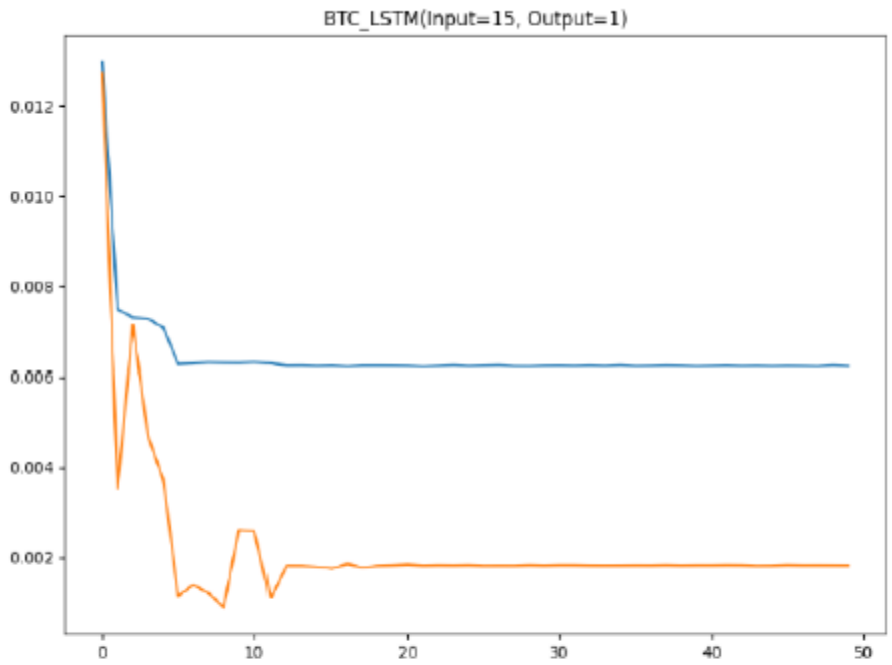
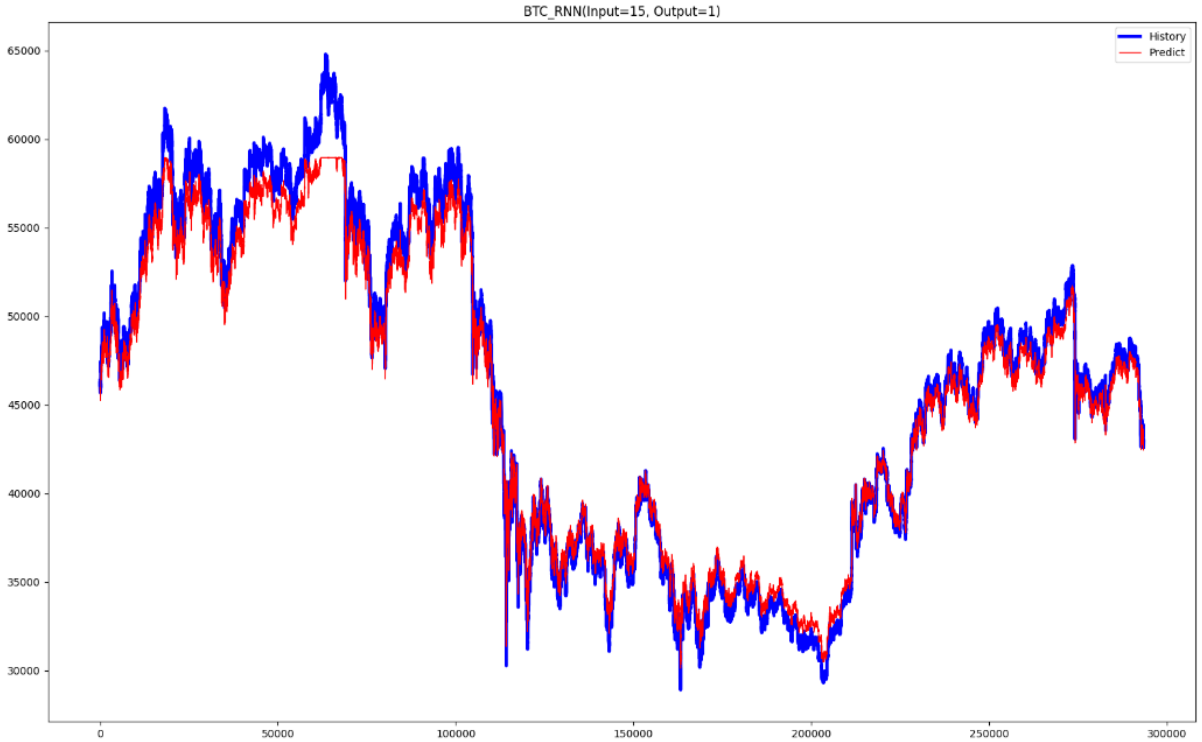


圖 6.1 使用 RNN 的訓練過程的 Loss(左)

圖 6.2 使用 LSTM 的訓練過程的 Loss(右)

Comparison of RNN, LSTM



RNN預測結果與實際收盤價的比較圖(左)

LSTM預測結果與實際收盤價的比較圖 (右)

LSTM for a single currency

- 以探討當利用前m天的收盤價來預測後n天的價格，如果當m, n不同的情況下會有甚麼結果。此處的MAE和RMSE是用於討論預測的收盤價與實際上的收盤價的差異。

BTC(m/n)	each_epoch time(s)	MAE	RMSE
15/1	22.88	75.676	104.557
15/5	22.80	95.287	143.041
15/10	22.82	118.789	181.931
30/1	32.15	59.811	86.017
30/5	32.25	91.916	139.361
30/10	32.12	125.373	187.727
60/1	52.19	58.034	84.799
60/5	52.16	92.033	139.963
60/10	52.41	125.067	187.641

表 6.2 使用 BTC 數據做為 LSTM 模型輸入及預測

ETH(m/n)	each_epoch time(s)	MAE	RMSE
15/1	22.56	4.643	7.041
15/5	22.82	7.200.	11.411
15/10	22.91	8.910	14.469
30/1	32.30	4.118	6.541
30/5	32.24	9.062	13.230
30/10	32.32	9.095	14.637
60/1	52.27	6.342	8.894
60/5	52.22	7.202	11.465
60/10	52.27	10.169	15.654

表 6.3 使用 ETH 數據做為 LSTM 模型輸入及預測

Pre-trained model with LSTM

- 上述的作法，只討論單一幣種使用自身收盤價來訓練，但實際上，不同幣種間的相關性也很大，也許我們同時考慮不同幣種的趨勢，訓練出來的模型效能會更好。
- 為了實驗這個想法，我們將BTC, ETH個別做MinMaxScaler，再將這些資料合併成更大的訓練集，用此訓練出來的預訓練模型，比較與使用自身收盤價訓練的模型(BNB、ADA)，看看有甚麼差異。

Pre-trained model with LSTM

BNB(m=15, n=1)	MAE	RMSE
Model: BTC+ETH	0.785	1.197
Model: BNB	0.913	1.368

表 6.4 預訓練模型與 BNB 模型的 MAE, RMSE 比較

ADA(m=15, n=1)	MAE	RMSE
Model: BTC+ETH	0.00347	0.00585
Model: ADA	0.00416	0.00646

表 6.5 預訓練模型與 ADA 模型的 MAE, RMSE 比較

Weight model with LSTM

- 考慮**某種貨幣的影響力大於其他幣種**，我們嘗試將選定的幣種個別做MinMaxScaler後，再加權平均，產生新的自定義虛擬貨幣，並討論看看這樣訓練出來的模型效果如何。
- 但是由於Kaggle上的”train.csv”所提供的資料中，每個幣種的資料的時間並沒有對齊。因此我們另外下載了各幣種的歷史數據[4]，其中時間範圍由2018/01/01至2022/01/22。
- 將2018/01/01至2021/12/31作為訓練(共1462筆)，2022/01/01至2022/01/20作為測試(共20筆)。

```
# 加權係數
weight = dict(BTC=6.78, ETH=5.89, BNB=4.31, ADA=4.40)
```

圖 6.5 在 Kaggle 上所提供的各幣種加權係數

Weight model with LSTM

BTC(m=10, n=1)	MAE	RMSE
Model: Weight	43003.666	43008.331
Model: BTC	42880.182	42883.013

表 6.6 加權模型與只使用 BTC 模型的 MAE, RMSE 比較

BNB(m=10, n=1)	MAE	RMSE
Model: Weight	468.174	468.271
Model: BTC	466.593	466.718

表 6.8 加權模型與只使用 BTC 模型的 MAE, RMSE 比較

ETH(m=10, n=1)	MAE	RMSE
Model: Weight	3315.563	3316.981
Model: BTC	3299.258	3300.431

表 6.7 加權模型與只使用 BTC 模型的 MAE, RMSE 比較

ADA(m=10, n=1)	MAE	RMSE
Model: Weight	0.825	0.871
Model: BTC	0.823	0.866

表 6.9 加權模型與只使用 BTC 模型的 MAE, RMSE 比較

Weight model with LSTM

- 加權係數模型的表現並沒有比單一使用BTC作為訓練的模型略遜一籌。
- 我們認為可能有以下兩個原因：
- 一是比起原先資料集(以分鐘為單位)有上百萬條的訓練資料，新的資料集(以日為單位)只有一千多筆資料，因此沒有辦法使模型有足夠的訓練，進而使表現略遜於單一使用BTC作為訓練的模型；
- 二是由於在加密貨幣本身是以BTC為主體的市場，因此BTC的走勢會帶動其他的幣種的走勢，所以加權係數的模型自然表現會略差一點。

Outline

- Recap of Problem 問題回顧
- Introduction 理論介紹
- Data Preprocessing 資料預處理
- Model Structure 模型架構
- Experiment & Discussion 實驗討論
- Conclusion 結論
- Future work 未來展望

Conclusion 結論

- 預測時間序列是一個非常有趣的領域。雖然我們在使用LSTM可以取得不錯的預測結果，但是由於加密貨幣市場跟一般金融市場的實際情況十分接近，並不是所有投資者都符合經濟學意義上的理性人假設。仍就會有投資人的不理性操作**(幫我還套在山頂的EOS哭哭)**，因此即使我們有不錯的預測結果仍舊應該視為輔助/參考用，並不應該完全相信模型的，自身仍舊需要有更多的思考能力，Machine learning is just a tool, not everything.

Outline

- Recap of Problem 問題回顧
- Introduction 理論介紹
- Data Preprocessing 資料預處理
- Model Structure 模型架構
- Experiment & Discussion 實驗討論
- Conclusion 結論
- Future work 未來展望

Future work 未來展望

- 我們在這次報告中有把BTC, ETH的收盤價訓練出來的模型做為預訓練模型，並且收到不錯的成效，這代表著：或許不同幣種間的漲跌是相關的，因此在未來有機會的話，我們希望去探討選擇那些幣種作為預訓練模型，例如選擇BTC, ETH, BNB, ADA，看看怎樣的選擇方式可以讓模型最好。
- 並且希望可以取得更為詳細的資料，完成對於加權係數模型的探討，理解說為甚麼在本次報告中的加權係數模型表現並沒有比較好。並且從中得到新的啟發，進而設計出表現更加優秀的模型。

THANKS