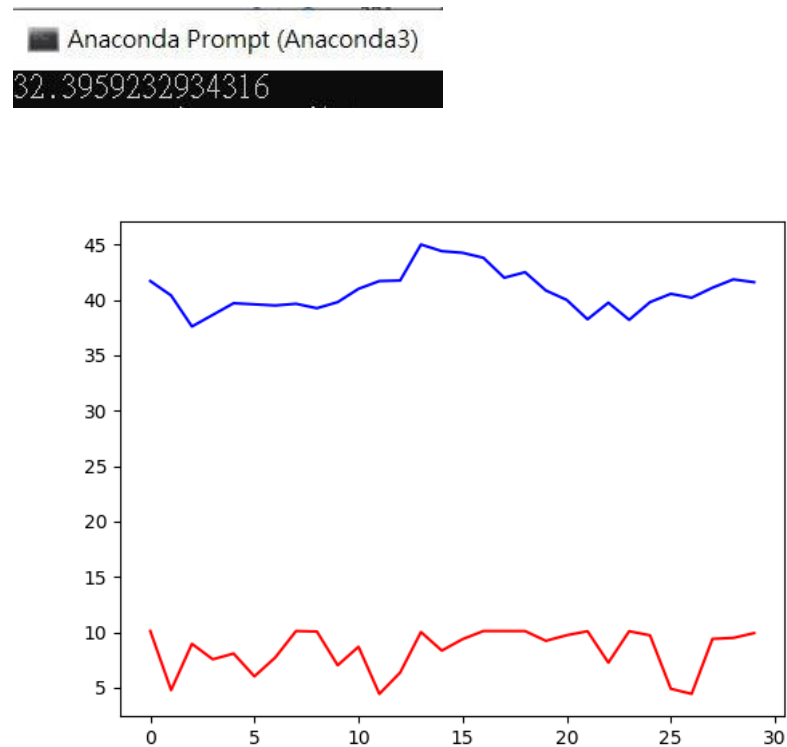


此次作業是用 RNN 進行股票預測，因為股票預測是可以透過過去的歷史資料和同產業的股價走勢去做未來股價的預測，需要的模型是將之前計算完成的輸出，再回傳給自己作為輸入，所以這次作業是修改利用 tensorflow 去建立的 LSTM 的模型，增加準確率。LSTM 的特性是具三個控制記憶的機制，分別是 Input Gate、Output Gate 與 Forget Gate。

一開始，我只有將助教範例程式的路徑改掉，跑看看出來的預測結果如何，跑出來的 acc 約為 32.4，可以明顯看出準確率欠佳，從圖形也可看出預測所代表的紅線和實際代表的藍線有明顯的差異。

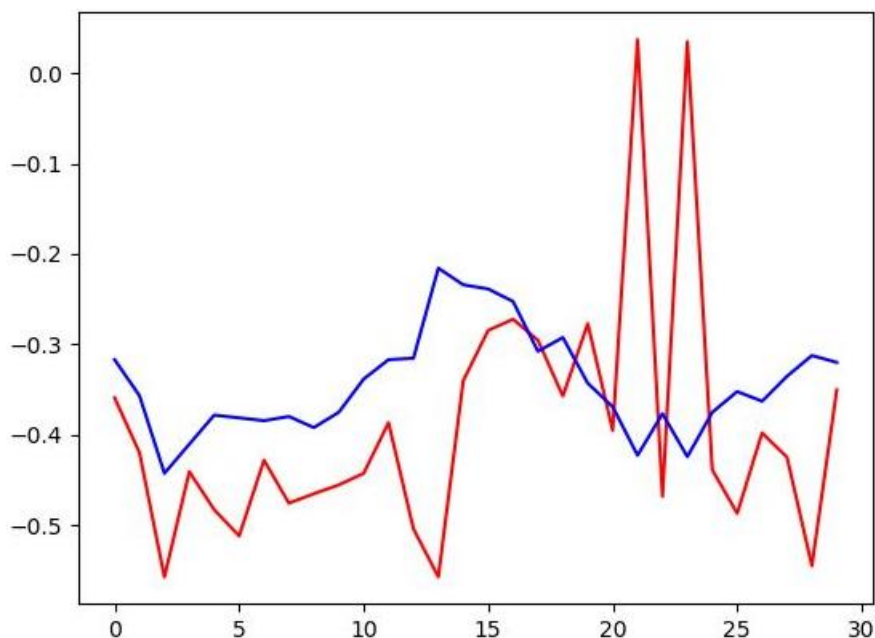


我覺得增加訓練次數和更改每幾步保存一次模型會影響準確率，故下一次訓練我將 for 迴圈的執行次數增加到 500，改成 10 步保存一次模型；此次結果的 acc 為 0.16，圖形也稍稍的相近了一些。

```

0.1640509558385717
real predict
0 [-0.31701623238766324] -0.411887
1 [-0.3568930085628182] -0.279341
2 [-0.4427814495554593] -0.406288
3 [-0.4105732841832189] -0.488283
4 [-0.37836511881097834] -0.448693
5 [-0.3814325631321442] -0.433690
6 [-0.38450000745331] 1.261725
7 [-0.37989884097156135] -0.495119
8 [-0.39216861825622434] -0.450107
9 [-0.37529767448981277] -0.408211
10 [-0.3384883426358236] -0.422508
11 [-0.31701623238766324] -0.373659
12 [-0.31548251022708046] -0.328620
13 [-0.21579056978919334] -0.405172
14 [-0.23419523571618792] -0.104177
15 [-0.23879640219793652] -0.272199
16 [-0.2525999016431825] -0.233133
17 [-0.30781389942416604] -0.279426
18 [-0.29247667781833725] -0.326912

```



下一次訓練，想說增加欣興電子(3037)歷史資料集(3037_2010_2019_csv)來輔助預測股價，然後再將資料進行標準化，再來將 rnn_unit 改成 20，batch_size 改成 5，因為是將欣興電子的資料 append 在南亞電子後，想說資料增加勢必會拉長訓練時間，故將迴圈先降成 300 次。

```

f1=open(r'C:\大四\大四下學期\人工智慧與機器學習\HW2\3037_2010_2019_csv.csv')
df_1=pd.read_csv(f1) #讀入股票數據
df_1=df_1.dropna(axis=0,how='all') #刪除表中全部為NaN的行
data=np.array(df[['close']]) #close為收盤價，獲取收盤價
data_1 = np.array(df_1[['close']]) #close為收盤價，獲取收盤價
normalize_data=(data-np.mean(data))/np.std(data) #標準化
normalize_data_1=(data_1-np.mean(data_1))/np.std(data_1) #標準化

```

```

time_step=10      #看10天預測第11天
rnn_unit=20      #hidden layer units
batch_size=5     #每一批次訓練多少個unit
input_size=1     #輸入層維度
output_size=1    #輸出層維度
lr=0.0006       #學習率

```

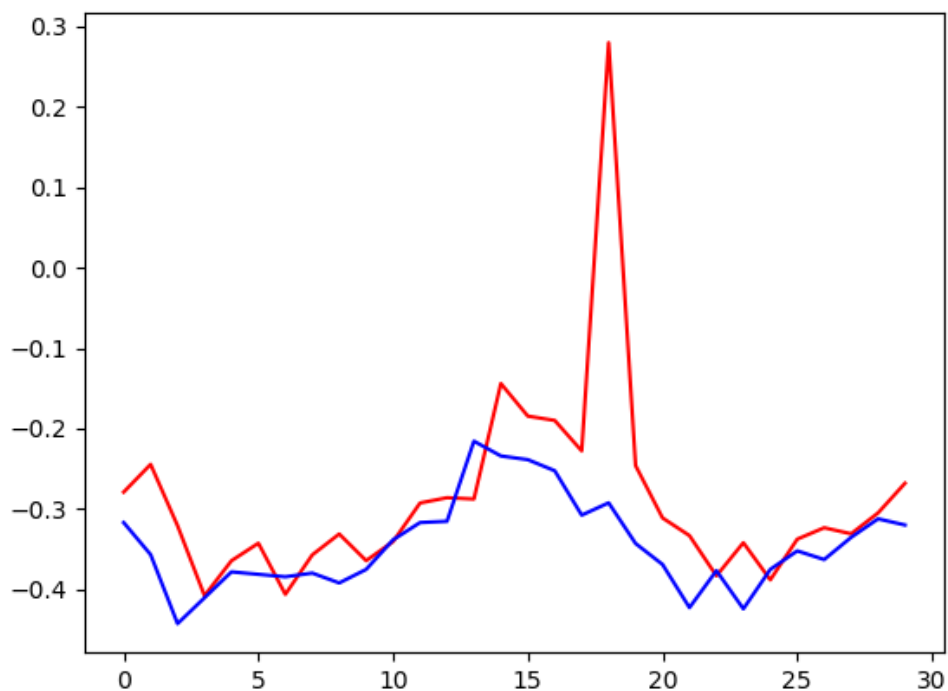
此次訓練完，acc 為 0.063 左右，從圖中可看出，加入另外一家的股票是能有效提升準確率，預測所代表的紅線和實際代表的藍線，除了在 15 到 20 中間有明顯的誤差之外，可看出剩下差異已沒那麼大。

```

acc: 0.06314610101080378

```

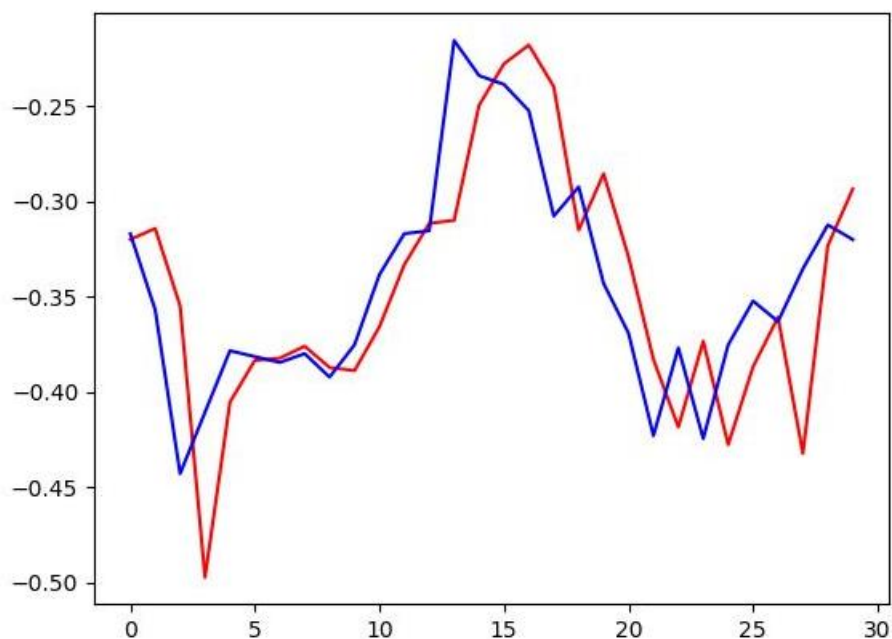
	real	predict
0	[-0.31701623238766324]	-0.279149
1	[-0.3568930085628182]	-0.244624
2	[-0.4427814495554593]	-0.320983
3	[-0.4105732841832189]	-0.408052
4	[-0.37836511881097834]	-0.364536
5	[-0.3814325631321442]	-0.342701
6	[-0.38450000745331]	-0.406314
7	[-0.37989884097156135]	-0.357240
8	[-0.39216861825622434]	-0.331251
9	[-0.37529767448981277]	-0.364417
10	[-0.3384883426358236]	-0.340652
11	[-0.31701623238766324]	-0.292699
12	[-0.31548251022708046]	-0.286127
13	[-0.21579056978919334]	-0.287770
14	[-0.23419523571618792]	-0.144102
15	[-0.23879640219793652]	-0.184627
16	[-0.2525999016431825]	-0.190005
17	[-0.30781389942416604]	-0.227865
18	[-0.29247667781833725]	0.279963
19	[-0.3430895091175722]	-0.246231
20	[-0.3691627858474812]	-0.311265
21	[-0.4228430614678819]	-0.333279
22	[-0.37683139665039556]	-0.383317
23	[-0.4243767836284647]	-0.342083
24	[-0.37529767448981277]	-0.388312
25	[-0.3522918420810696]	-0.337694
26	[-0.36302789720514955]	-0.323491
27	[-0.3354208983146578]	-0.331023



原本想要再進行兩次訓練，迴圈次數都設為 1000 次，每 5 次保存一次模型，第一次構想為沒加欣興電子的股票去做預測，看準確率如何，第二次訓練為加欣興電子的股票去做預測，最後比較有加與沒加的準確率差異；不過由於迴圈次數設為 1000 次，跑一次就得花費一整天的時間，在時間不足下，只進行構想中的第一次訓練。

在此次訓練完，acc 為 0.033 左右，由圖形可看出藍線和紅線的相似程度是所有四次訓練中最高的一次，我認為如果有把欣興電子加進去協助預測，準確率應該能再上升一些。

```
Anaconda Prompt (Anaconda3)
acc: 0.03399324763323622
real    predict
0 [-0.31701623238766324] -0.320036
1 [-0.3568930085628182] -0.314289
2 [-0.4427814495554593] -0.354831
3 [-0.4105732841832189] -0.497118
4 [-0.37836511881097834] -0.405345
5 [-0.3814325631321442] -0.383408
6 [-0.38450000745331] -0.382252
7 [-0.37989884097156135] -0.375995
8 [-0.39216861825622434] -0.387134
9 [-0.37529767448981277] -0.388792
10 [-0.3384883426358236] -0.365839
11 [-0.31701623238766324] -0.333259
12 [-0.31548251022708046] -0.311621
13 [-0.21579056978919334] -0.310033
14 [-0.23419523571618792] -0.249787
15 [-0.23879640219793652] -0.227996
16 [-0.2525999016431825] -0.218188
17 [-0.30781389942416604] -0.240038
18 [-0.29247667781833725] -0.315132
19 [-0.3430895091175722] -0.285622
20 [-0.3691627858474812] -0.329148
21 [-0.4228430614678819] -0.382879
22 [-0.37683139665039556] -0.418361
23 [-0.4243767836284647] -0.373330
24 [-0.37529767448981277] -0.427605
25 [-0.3522918420810696] -0.386724
26 [-0.36302789720514955] -0.360813
27 [-0.3354208983146578] -0.432177
```



此次作業，我覺得自己應該要在第一個禮拜就開始進行，因為此次作業，進行一次訓練，往往半日甚至一日就過了，所需要的時間成本很高，如果自己於第一個禮拜就開始，相信會更有餘裕進行更多次的嘗試。