Personal Report

第五組 40847016S 謝尚恆

專題形式：

組合探勘模組

資料集：

Stock Exchange Data

<https://www.kaggle.com/mattiuzc/stock-exchange-data?select=indexInfo.csv>

實作目的：

建構出一個可以藉由當日最高價、當日最低價，來預測收盤價的模型。

採用模組：

sklearn.ensemble的RandomForestRegressor

sklearn.linear\_model的LinearRegression

前處理：

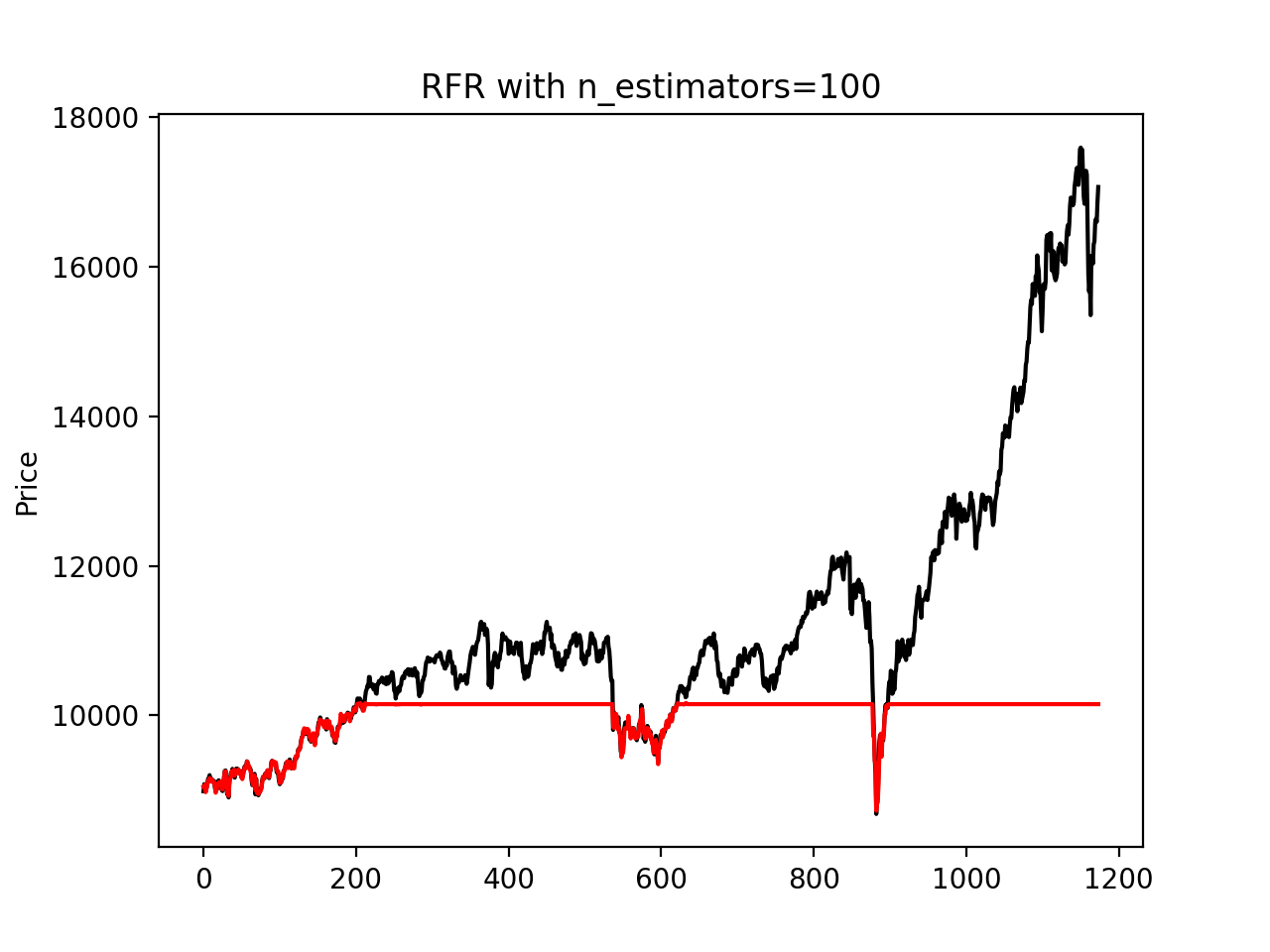
* 將csv檔裡有null值的資料移除
* 將標示每一個column的標籤移除
* 擷取High,Low欄位當作data
* 將Close當作target
* 將High,Low的所有值全部÷2

執行方式：

直接用python指令執行lr.py, rfr.py

改變控制參數/技術說明：

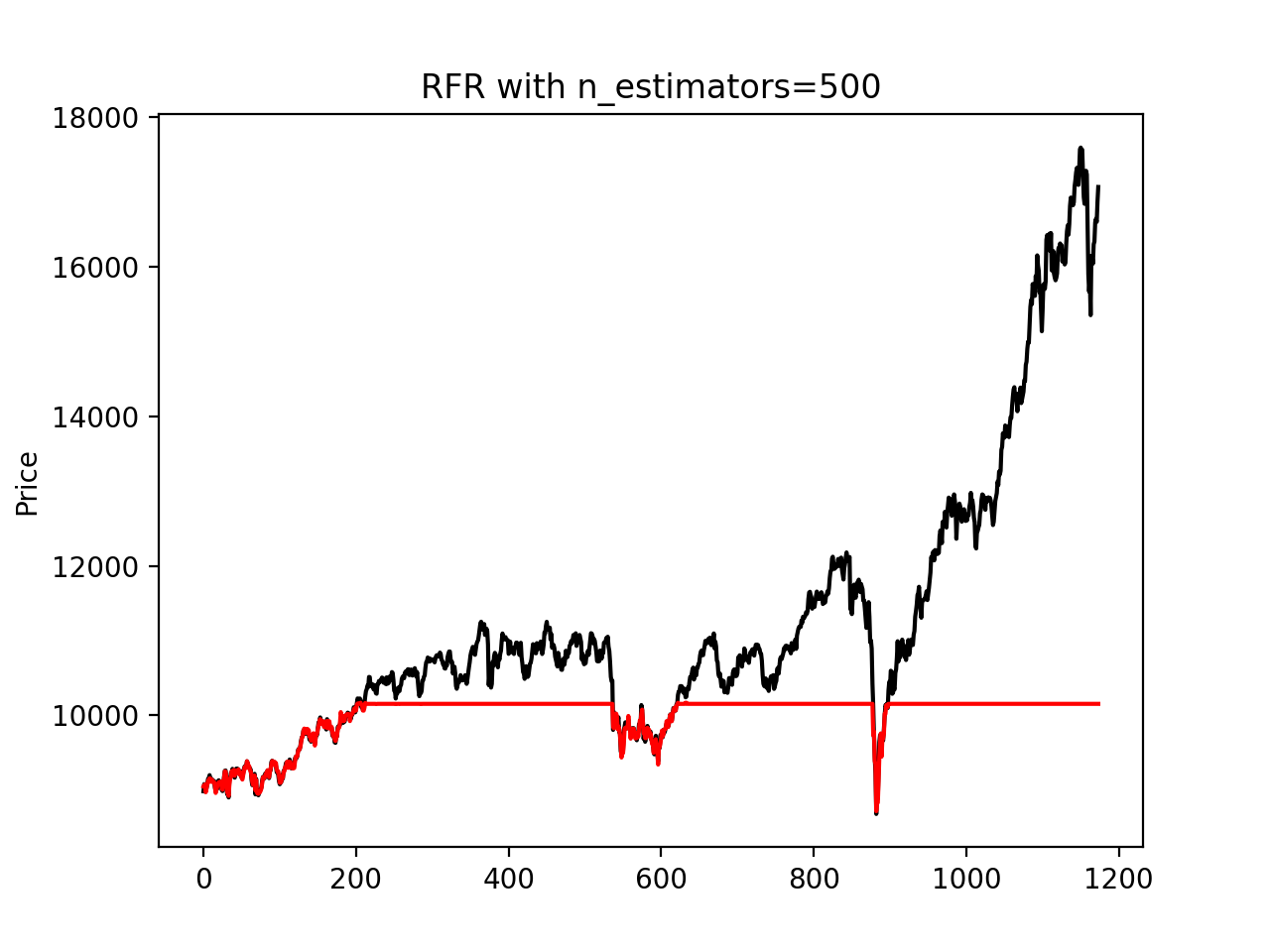
這是未進行調整的結果：（黑為實際資料、紅為模型的預測）



1.n\_estimators=500

因為RandomForestRegressor的預測表現比較不好，所以我想先從調整決策樹的數量下手，將數量調至500棵，但是效果仍然一樣。

這是經過第1項調整的結果：（黑為實際資料、紅為模型的預測）



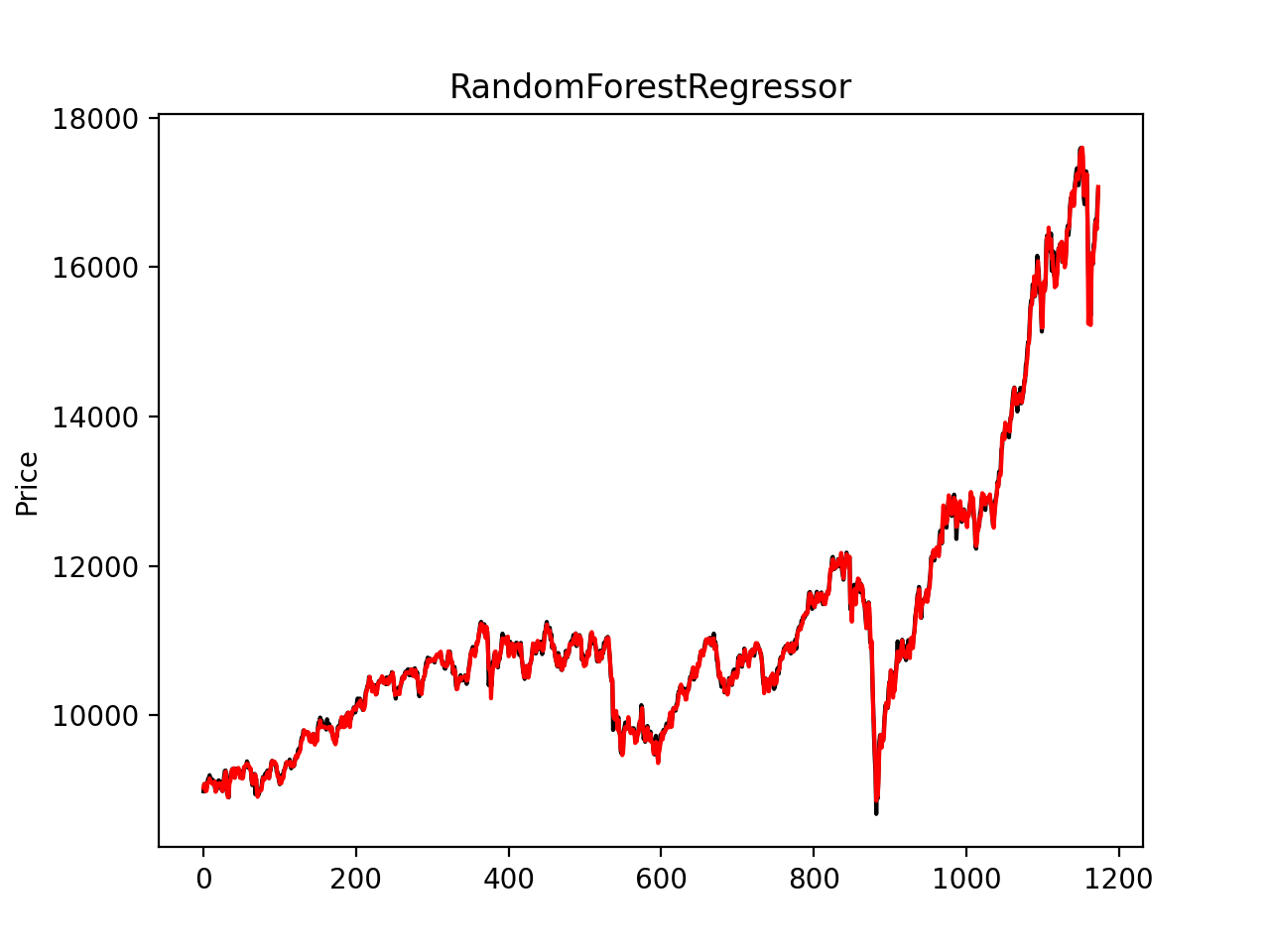
2.調整High,Low欄位的資料

觀察圖表後，發現RandomForestRegressor預測的值有一個上限，如果超過上限，則RandomForestRegressor就會當作那筆資料的預測值為上限。使用max()函式讀出該上限大約為10165。

所以我決定先將data，也就是High,Low欄位的所有數值全部÷4，然後待模型預測完之後，再將預測結果x4並與實際資料比較。

但是在測試的時候發現，RandomForestRegressor除了有上限值，還有下限值，而下限值大概是3500，而實際資料中最低的數據為9000，所以用4倍伸縮也不適合，後來決定使用2倍，這樣的話實際資料(9000~17500)就能完全的覆蓋在它的預測範圍之內(3500~10165)。

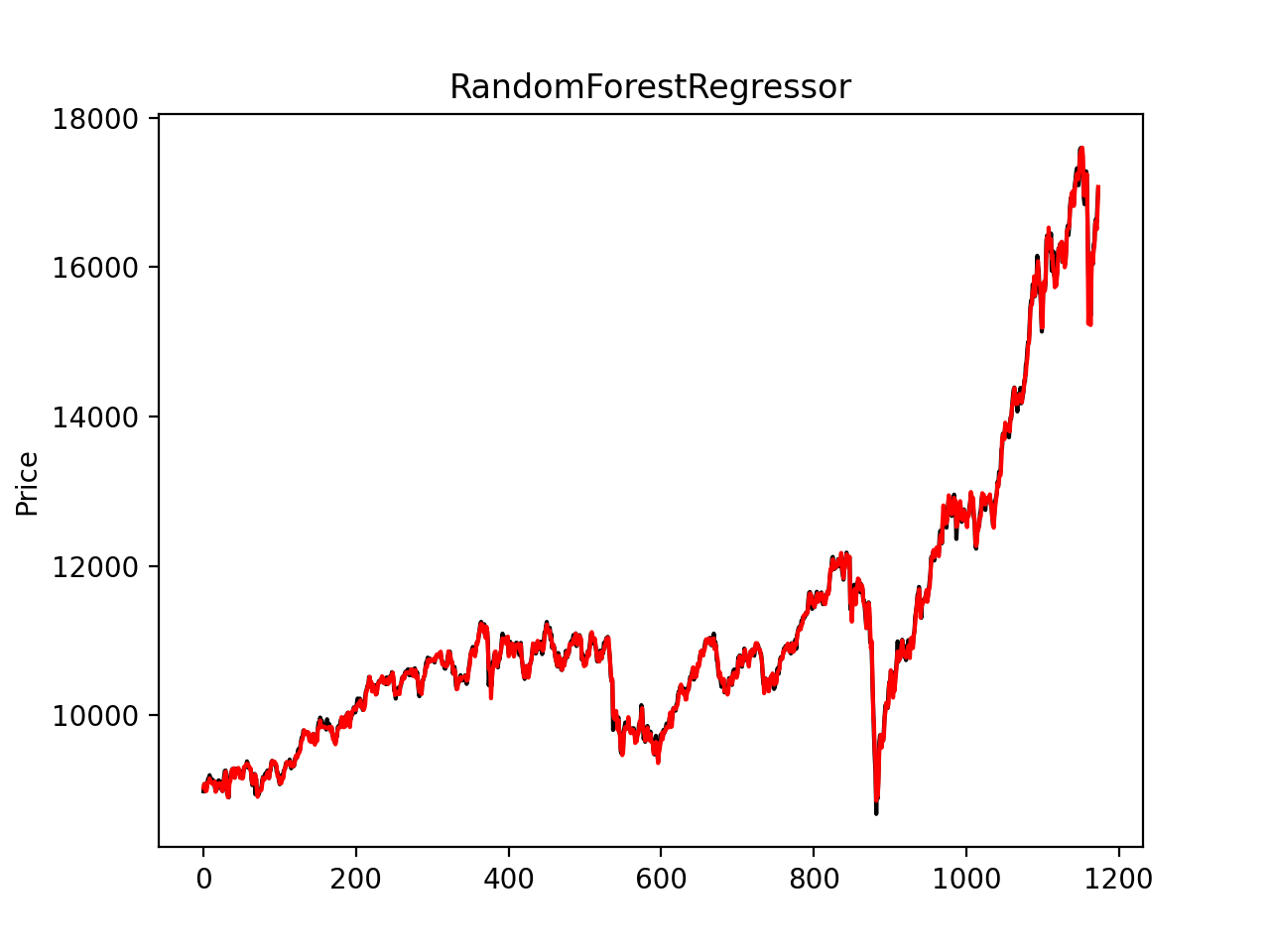
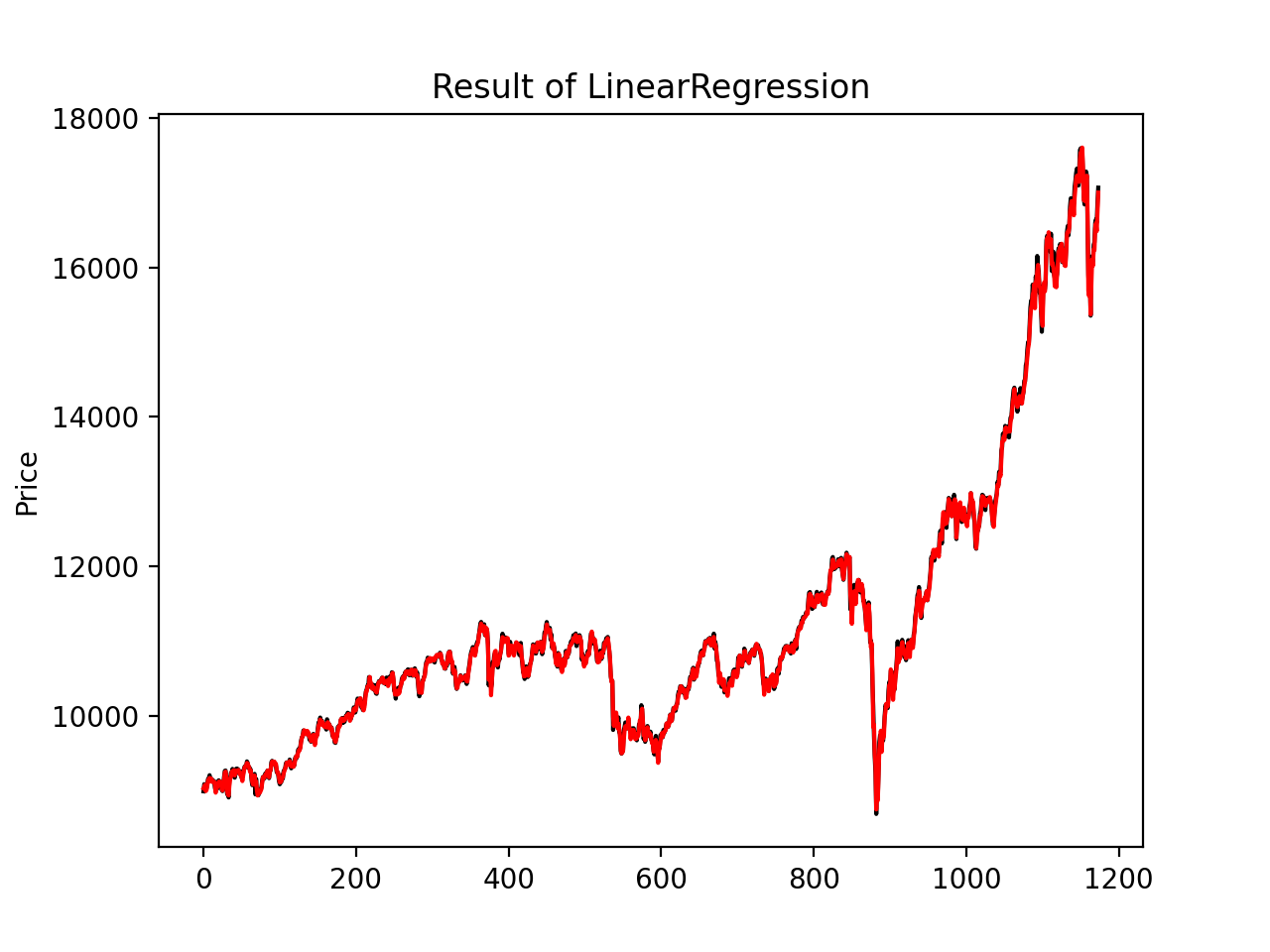
這是經過第2項調整的結果：（黑為實際資料、紅為模型的預測）



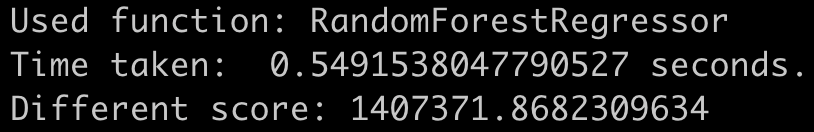
評估方法：

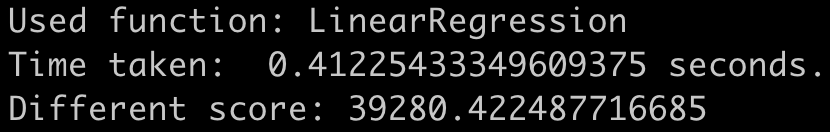
使用差異值，也就是先算出「每筆預測資料」與「每筆實際資料」差的值，然後再加總，數值越小，代表預測資料越相近實際資料。

結果及討論：

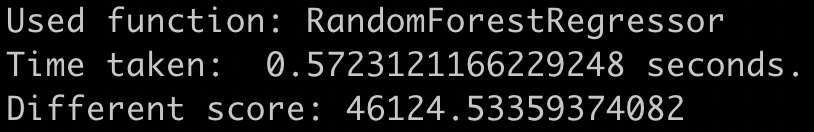


我們可以看到，若是讓這兩個模型使用一模一樣的資料進行學習並預測，因為RFR (RandomForestRegressor)的預測值有一個上限，當實際資料超過上限越多，差距就會越多，導致RFR最後的差異值來到140萬左右。而LR(LinearRegression)幾乎貼合實際資料，表示模型訓練的成果相當不錯，差異值只有3.9萬左右。





不過，若針對RFR將資料調整成RFR可正確預測的範圍，則RFR的表現也很優異，差異值也下降到4.6萬，也和實際資料非常相近。



若是從學習加辨識的時間來看，LR只需要0.4秒，而RFR則花了0.55秒左右，LR的表現依然比較好。

綜合不用調整資料、學習速度極快、預測精準這些優點，LinearRegression無疑是最適合用來預測價錢走勢的演算法。