Machine Learning for Stock price prediction

王茗冠、王子軒、黃三騰 2019/06/13

研究目的

隨著年紀的增長，如何理財這個事情變得相當重要，一般人會做被動型投資，像是定存或是基金投資，但是如果想要讓財富顯著的增加，許多人會選擇股票作為投資標的。但是，如何選擇一隻會讓自己賺錢的股票，可以說是長久以來的難題，且市面上充斥著良莠不齊的分析師，有許多的不同的投資策略和技術指標，讓人無所適從。我們期望可以利用機器學習的方法，在眾多的技術指標中，預測未來股票漲或跌的可能性，提升投資的報酬率。

資料來源

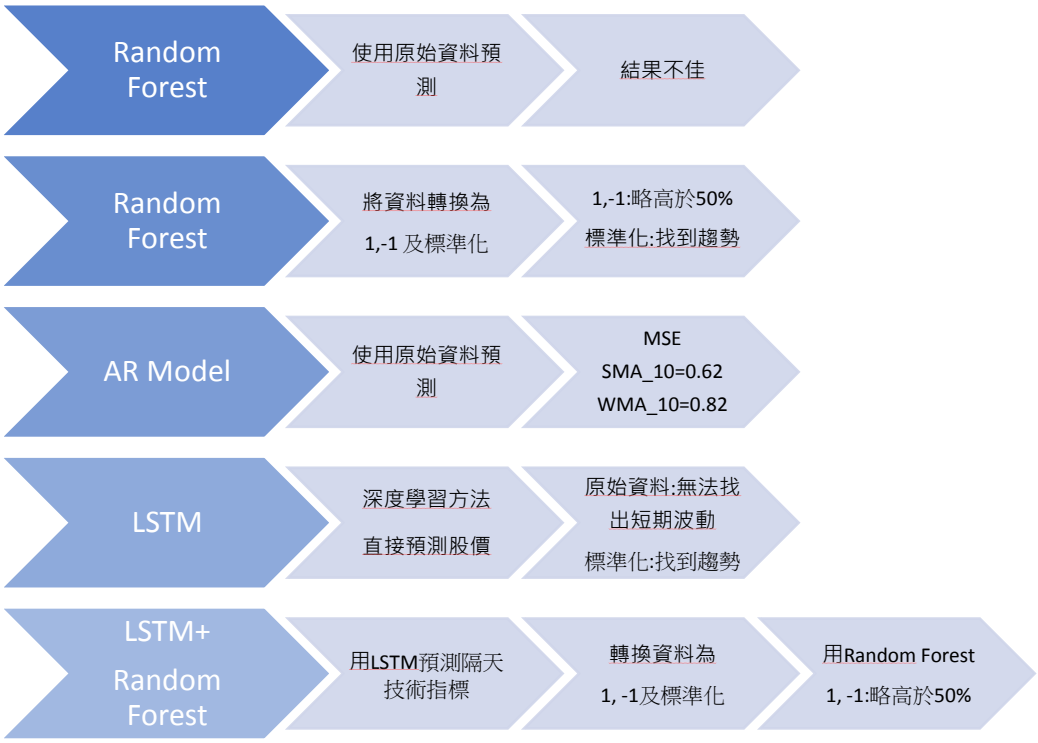
利用資料爬蟲的方式，從台灣證交所的網站上下載，也利用看盤軟體下載資料。

資料內容

1. 收盤價：為個股當天收盤的價格。
2. SMA：N 日收盤價總和/N，這邊我們採用了五日，十日及二十日，若股票向上突破 SMA 則代表股價走強，為買進訊號，反之亦然。
3. WMA：（十日前收盤價*1+九日前收盤價*2+…+今日收盤價*10）/(1+2+…+10)，我們採用十日的 WMA，為十日的股價加權平均。
4. RSV：(今日收盤價-九日內最低價)/(九日內最高價-九日內最低價)*100，該指標值介於 0 到 100 間。
5. K 值：前日 K 值*(2/3) + 當日 RSV*(1/3)，該指標值介於 0 到 100 間。
6. D 值：前日 D 值*(2/3) + 當日 K 值*(1/3)，該指標值介於 0 到 100 間。
   當 K 值由下而上穿越 D 值，為黃金交叉，行情看好；K 值由上而下跌破 D 值，為死亡交叉，行情看壞。
7. RSI：[九日股價上漲幅度的加總 /（九日股價上漲幅度的加總 + 九日股價下跌幅度的加總）]*100 ，越高時代表市場越熱絡，越低時越冷清，但在 RSI 大於 80 時，股價下跌的機率大，RSI 小於 20 時，上漲的機率高。
8. W%R：（十日內最高價 - 今日收盤價）/（十日內最高價 - 十日內最低價）* -100%。一般來說高於-20%為超買，反之，低於-80%以下為超賣，值為 0 時表示今日收盤價為十日內最高價。
9. 動量指標 (Mom)：當日股價 - 前九天的股價，由此可看出股價在其中波段漲跌幅度。
10. CCI：當典型價格 (TP) 等於其平均值時，CCI 值會等於零，只有當最後股價在極短期內作劇烈的向上或向下運動時，CCI 值才 會出現突然向上或向下大幅擺盪的極端值。

11. 三大法人：分別為外資、投信、自營商，所佔的持股比例。 一般而言，若外資持股比例在股票市場中佔有較大，則股價的漲跌容易受到外資的買賣而影響。

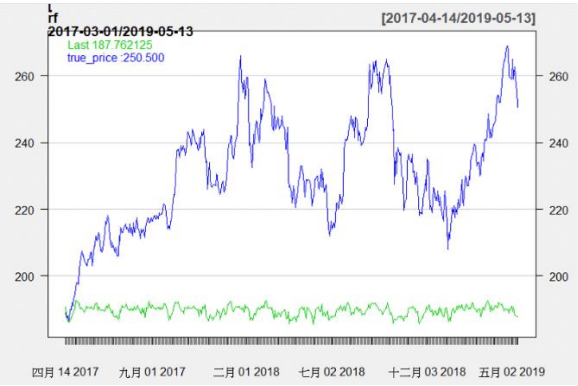
流程圖



Random Forest

一開始我們將所有原始的資料放入 Random Forest 做訓練，訓練資料和測試資料比為八比二，圖一是用所有的技術指標，預測出明天收盤價。可以發現到預測的結果並不好，比實際值低估了許多。

接著將資料中的X標準化，Y不做標準化，且訓練資料和測試資料分別做標準化，得到的結果如圖二所示，可以發現到可以逐漸預測到股價的震盪，但估計出來的值仍比實際值低了不少。



圖一: 藍色: 真實值;　綠色: 預測值

圖二: 藍色: 真實值;　綠色: 預測值

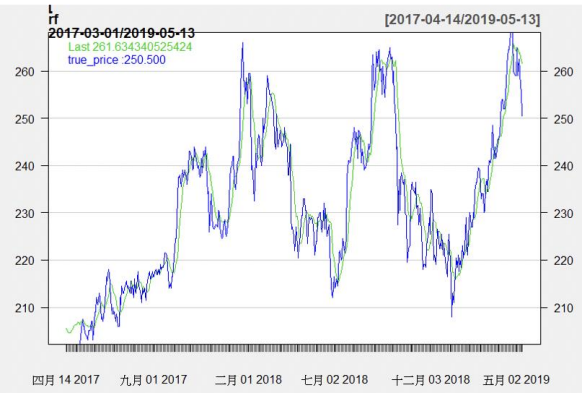
- train error (MSE): 0.5560 ;
- test error (MSE): 1983.96

- train error (MSE): 0.5664 ;
- test error (MSE): 16383.32

我們也將資料中的X和Y都做標準化，且訓練資料和測試資料分別做，結果如圖三所示，可以發現到估計的結果更接近實際的值了。

另一個作法參考了 Patel, Jigar, et al 的論文，調整 X 和 Y 的資料，舉例來說，用前日的收盤價做比較，若比前日高則調整為 1，若比前日低則為-1;若技術指標對比前日來說，得到股價看漲的趨勢，則調整為 1，反之則調整為-1。結果如圖四所示，發現預估的準確率並不高，只略高於 50%而已。

- train error (MSE): 0.5623 ; test error (MSE): 33.40811



圖三: 藍色: 真實值; 綠色: 預測值

predict \ true	-1	1
-1	163	149
1	95	104

Train_error(錯誤率): 27.25%
Test_error(錯誤率): 47.74%

Accurancy(正確率): 52.26%

圖四

AR model

我們使用的另一個模型為 Autoregressive model (AR)，是時間序列的模型，利用一個或數個過去期間的資料，找出模型預測未來，這邊我們利用今天的收盤價，預測明天的收盤價，較特別的部分在於這個模型只使用了一個變數，為收盤價，就可以預測隔天的資料，也有使用當日的技術指標預測隔天的技術指標，圖五圖六為預測的結果。

圖五: 藍色: 真實值;　綠色: 預測值

- train error (MSE): 2.4788 ; test error (MSE): 10.9944

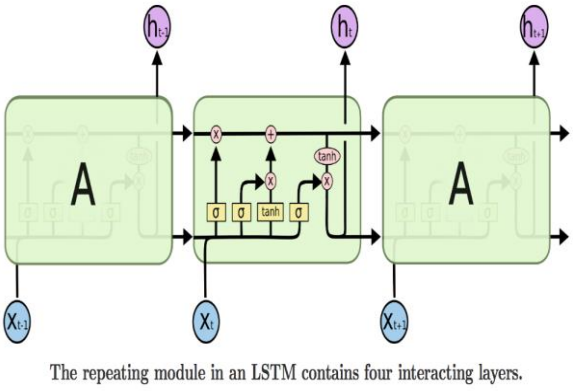
MSE	AR(1)	AR(2)	AR(3)	AR(4)	AR(5)
Y(股價)	10.9944	10.9740	11.1530	11.1813	11.2540
SMA_10	0.9659	0.2002	0.2007	0.2013	0.2010
WMA_10	1.2684	0.3748	0.3758	0.3759	0.3752
RSI_9	48.5738	48.7603	49.1203	49.2162	49.3017
K_value	72.2447	55.7346	55.2089	55.5410	55.4482
D_value	24.4506	6.3513	6.0907	6.1434	6.1638
CCI_14	1993.6600	1941.0450	1944.4980	1945.9440	1949.0350

圖六

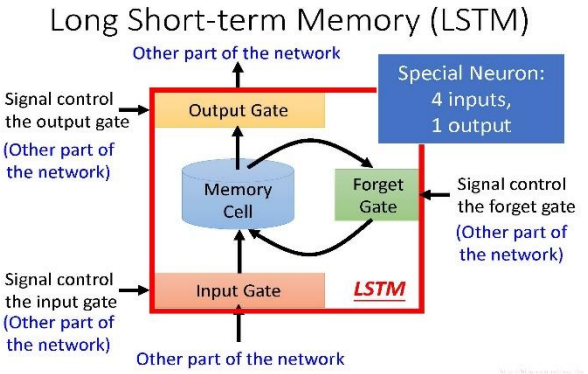
LSTM (Long short-term memory)

我們最後一個要使用的模型為 Long short-term memory，這個模型是屬於 deep learning 架構裡面的 recurrent neural network (RNN)其中之一，整體的流程大致是許多的「模塊」串聯在一起的，而每個「模塊」裡面通常會有一個存資料的「單元」及三個門:輸入門、輸出門、忘記門，這三個門的開或關是經由這套系統自行調整的，所以當我們進入「模塊」，是有 four inputs:新的資訊、跟個別控制三個門的開關；one output:產生出來的預測資訊。

以下圖七與圖八是 LSTM 的架構圖:



圖七

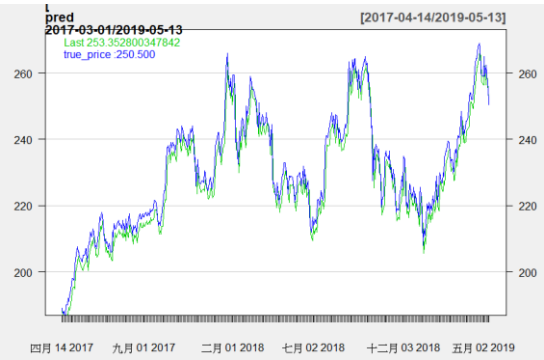


圖八

使用 LSTM 對股價直接進行預測

這邊使用的資料是把每一天的收盤價換成前後兩天的差值，例如:第一筆資料為第一天的收盤價與第二天的收盤價的差值，第二筆資料為第二天的收盤價與第三天的收盤價的差值，…以此類推。之後利用第一筆差值去預測第二筆差值這樣的預測模式進行，再把差值加上原本的真實股價，就可以得出我們想要估計的隔一天股價。

第一種想法是把轉換後的解釋變數(X)及應變數(Y)都進行標準化，放入 LSTM 模型直接進行預測股價，我們可以看到在測試資料的預測出來跟真實的值對比圖(圖九)



- train error (MSE): 2.5295 ;
- test error (MSE): 19.0717

圖九: 藍色: 真實值；綠色: 預測值

可以從(圖九)發現估計的狀況還不錯，但是這款模型的預測都有慢一天的趨勢，所以這個模型比較像是學到前一天的股價當作隔天的股價來估計。

LSTM + Random forest

會使用這個方法是因為想說讓深度學習的方法跟機器學習的方法結合再一起使用，看看這樣做會不會提升正確率。

第一種方法是使用上述的 Random forest 並把資料轉成正負 1 的形式稍加改變，我們把那些$X_1, \dots, X_p$的未來值使用預測的方法來估計，再轉換回正負 1，代入 Random forest 模型預測結果。

我們在圖十一可以看出正確率是有些許提升，但跟原本使用 Random forest 的結果差不多，對於實質上並不會有太大的幫助。

LSTM	SMA10	WMA10	RSI	Mom	k_value	d_value	CCI	%R
MSE	0.2053	0.0863	7.0251	0.2827	1.0709	1.671	458.5	53.77

圖十: 為使用 LSTM 估計技術指標$X_1, \dots, X_p$的誤差

true / predict	-1	1
-1	175	150
1	83	103

Test_error(錯誤率): 45.59%
Train_error(錯誤率): 27.25%

Accurancy(正確率): 54.41%

圖十一: Random Forest 預測明日的股價漲跌 Y

第二種方法是使用第一種方法預估出來的未來值放入 Random forest 的模型(資料中的 X 標準化，Y 不做標準化，且訓練資料和測試資料分別做標準化)下計算，放棄轉換成正負 1 的方式，是因為這個方法是經由大家常常界定的資訊來判定了，但是我們應該讓機器去學習到這樣的規則，說不定跟我們一般界定的範圍是不一樣的。

我們做出來的結果(圖十二)還是不如預期是有可能是因為在 LSTM 下的估計就會出現誤差，而這樣的誤差放入 Random forest 的模型時，還是會出現問題的。

- train error (MSE): 0.4942 ;
- test error (MSE): 38.2950

圖十二: 藍色: 真實值；綠色: 預測值

模型比較

可以發現到 AR 模型訓練速度較快，因為只有股價單一變數，但是準確率比起另外兩個模型較高。相比之下，LSTM 訓練時間最長，但是準確率卻沒有比較高。

	Random Forest	AR	LSTM+ Random Forest
變數資料	多	少	多
訓練速度	快	快	最慢
準確率	普通	高	普通

問題討論

- 不論是做哪個模型，先將資料標準化對預測的結果差異相當大。
- 其中我們有將技術指標轉換為 1,-1，再用 Random Forest 做預測，發現到這個方法並不可行，準確率不高。也有嘗試用過 LSTM + Random Forest 的方法，但是預測結果和單純使用 Random Forest 差不多。
- AR 模型預測結果最佳，研判可能原因在於資料包含時間性，因此機器學習方法預測誤差較大，利用時間序列模型會得到較好結果，未來也可以試著利用時間序列模型結合機器學習方法，或許會得到更好的結果。 由於我們目前只預測隔天的股價，期望未來可以預測長期趨勢，或是預測出最佳的投資策略。

參考文獻

- Patel, Jigar, et al. ” Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques.” Expert Systems with Applications 42.1 (2015): 259-268.
- LSTM 作法: http://rwanjohi.rbind.io/2018/04/05/ time-series-forecasting-using-lstm-in-r/
- Random Forest: https://www.finlab.tw/Machine-learning
- Ar model: http://homepage.ntu.edu.tw/ sschen/Book/Slides/ Ch3AR.pdf