1. How did you preprocess this dataset ?

增加了幾個column:

return: high price – low price

close to open: |close price – open price|

close to high: |close price – high price|

close to low: |cloes price – low price|

result: close price跟前一天比是升還是跌(0: 下跌, 1:上升)

把result及date外的data進行normalize

1. Which classifier reaches the highest classification accuracy in this dataset ?

Logistic Regression accuracy: 0.7928286852589641

SVM accuracy: 0.8247011952191236

Neural network accuracy: 0.5219123505976095

* Why ?

這次我們要預測的是股價上升或下跌，只會return 0 or 1，可能因為這個原因，logistic regression和SVM這種比較簡單直觀的模式能發揮比較好的表現，像neural network這類複雜而且每筆資料都比較獨立處理的模型表現反而不好。不過在股票市場中股價會受基本分析面(像政策、外圍消息)等的影響，難以用模型作預測，也會減低準確率。

* Can this result remain if the dataset is different?

我使用sklearn中的 train\_test\_split以train:test=7:3作測試

Logistic Regression accuracy: 0.7879234167893961

SVM accuracy: 0.8085419734904271

Neural network accuracy: 0.5449189985272459

結果跟用test data的差不多

1. How did you improve your classifiers ?

除了一開始的preprocessing外，我還透過sklearn中的GridSearchCV協助取得 model parameter的最佳值。先用list儲存欲測試的值，然後借助GridSearchCV解出每一個組合的 score，再提取最佳score中的參數值。