**Fundamentals of Data Analytics HW3-1 Report**

**F74062028 歐子杰**

**Classifier Select**

* **Logistic Regression**

使用sklearn.linear\_model的LogisticRegression

* **Neural Network**

用pytorch 實現RNN(Recurrent Neural Network)的LSTM(Long Short-Term Memory) model

* **One other classifier**

使用sklearn.ensemble的RandomForestClassifier

**Discussion**

* **How did you preprocess this dataset ?**

我將當天的所有資訊(Open Price, Close Price, High Price, Low Price, Volume)作為train\_x，用以預測隔天的Close Price，再比較Real Close Price和Predict Close Price的漲跌(相較於前一天的Close Price)

Training Logistic Regression Model的過程發現，會有全部結果都是上升或跌落的撞況，推測是不同的feature本身數值的量級差異，所以對所有的train\_x features做normalize

由於這次的資料沒有缺失值，也不需做數值轉換，故preprocess就只做normalize

* **Which classifier reaches the highest classification accuracy in this dataset ?**

RNN LSTM Reach highest in this dataset

* + **Why ?**

RNN EPOCH我設成300，Learning Rate 0.0001，參考前30筆資料，因為訓練的過程會滾動式的修正結果，得到的train accuracy和test accuracy相較於其他兩種Classifier更不會因為過於專注在實際值得預測而忽略了走勢，換句話說Logistic Regression和Random Forest預測出來的close price對於train data來說較高，但是在ups and downs部分test data誤差會比較大且實際數值會偏低

認為是因為RNN LSTM會一直參考前30筆資料做修正的結果，必較能掌握變化的趨勢

* + **Can this result remain if the dataset is different ?**

當然accuracy不能重現，但RNN LSTM的ups and downs預測趨勢依然會比另外兩者高

* **How did you improve your classifiers ?**
  + Logistic Regression

對資料做normalize，依據sklearn官方doc上的說明，調整penalty、solver、max\_iter參數，比較有影響的是penalty和solver

* + RNN LSTM

RNN LSTM參照前面多少筆資料，以及learning rate對結果影響最深

* + Random Forest

對資料做normalize，調整n\_estimators和random\_state，主要n\_estimators決定the number of trees in the forest，對結果較有影響

因為使用模型的經驗不多，所以我沒有概念要怎麼調整，都是看完範例default value在一次調一些參數試出來的