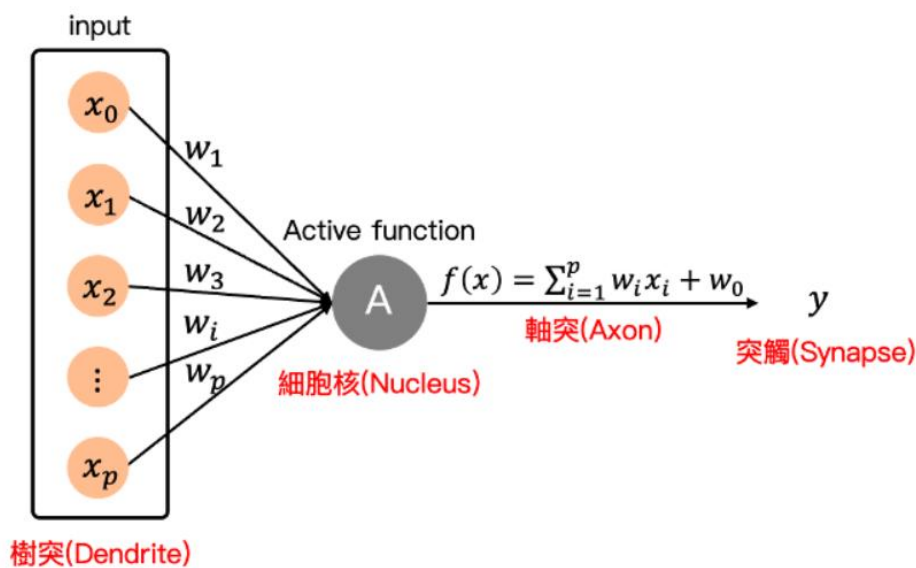


統計模型與 AI 模型混淆矩陣分析

統計模型基於統計學原理，通常較容易解釋，模型裡的參數和結果具直觀的統計意義。**AI 模型**涵蓋多種人工智慧方法及技術，應用層面廣泛，學習方式包括監督式學習、非監督式學習及強化學習...等，模型複雜性相對統計模型來的高，神經元、隱藏層...等參數設定皆關係到模型訓練的結果，解釋性較困難。

一、神經元種類

(一)、類神經網路(NN)



類神經網路(Neural Network, NN)，又稱為神經網路、人工神經網路(Artificial Neural Network, ANN)，屬於非線性的統計模型，是深度學習(Deep learning)的核心概念，而深度學習也是機器學習的一個分支領域，在影像辨識、語音識別與自然語言處理等方面有許多的應用。

- 機器學習
- 具有人腦基本的特性:學習、記憶和歸納
- 非線性統計性資料建模工具

參考網站: <https://ithelp.ithome.com.tw/articles/10305927>

(二)、循環神經網路(RNN)

語言通常要考慮前言後語，以免斷章取義，也就是說，建立語言的相關模型，如果能額外考慮上下文的關係，準確率就會顯著提高，因此，學者提出『循環神經網路』(Recurrent Neural Network, RNN)演算法，它是『自然語言處理』領域最常使用的 Neural Network 模型額外考慮前文的關係。也就是會把回歸模型的誤差項帶進去，把每一次的錯誤縮小修正。

- 神經網路中的一種
- 可以接受更加廣泛的時間序列結構輸入
- 處理次序資料
- 在隱藏層中會記錄每次處理的結果(權重)

二、 AI模型與統計模型

(一)、AI模型

1. LSTM

- 在 RNN 的基礎上加入了
- 遺忘閘門(forget gate)
- 輸入閘門(input gate)
- 輸出閘門(output gate)
- 防止梯度消失/爆炸的問題

2. BI-LSTM

- 能捕捉前後文
- 從兩個方向理解上下文的任務中特別有用
- 通常用於情緒分析、命名實體識別和機器翻譯等任務

3. GRU

- 在 LSTM 的基礎上將
- 遺忘閘門(forget gate)
- 輸入閘門(input gate)
- 以上兩種整合並命名為 update_gate

(二)、統計模型

1. Logistic -統計上的回歸
2. Bayesian-樹狀圖分類

(三)、演算法

- NSGA2(多目標非支配排序遺傳算法)
- 本次專題使用此模型參照專題

三、混淆矩陣

效力檢定(混淆矩陣)

- 回召率(recall)：模型識別正類樣本的能力 $\frac{TP}{TP + FN}$
- 精確率(precision)：模型預測為正類的樣本中有多少是真正的正類樣 $\frac{TP}{TP + FP}$
- 特异性(specificity)：模型識別負類樣本的能力 $\frac{TN}{TN + FP}$
- 陰性預測值(NPV)：模型預測為負類的樣本中有多少是真正的負類樣本 = $\frac{TN}{TN + FN}$
- F1-score = $2 \times \frac{(\text{回召率} \times \text{精確度})}{(\text{回召率} + \text{精確度})}$
- 準確度(accuracy)： $(TP + TN) / (TP + TN + FP + FN)$

| 真實 \ 預測 | 上漲 | 下跌 |
|---------|----|----|
| 上漲 | TP | FP |
| 下跌 | FN | TN |

涵義

| 真實 \ 預測 | 上漲 | 下跌 |
|---------|--------------------------------------|-------------------------------------|
| 上漲 | 真實陽性 true positive(TP) | 虛偽陽性 false positive(FP) (型 I 錯誤) |
| 下跌 | 虛偽陰性 false negative(FN) (型 II 錯誤) | 真實陰性 true negative(TN) |

混淆矩陣常被應用於機器學習的分類領域中。會使用混淆矩陣內的元素加

以計算準確度(Accuracy)、召回率(Recall)、精確度(Precision)...等，用以判

斷該模型的表現。

- 召回率 >> 用以衡量模型辨別正類(TP)樣本的比例，表示模型能否正確預測股票上漲的比例。
- 準確度(Accuracy) >> 表示模型在預測股票上漲或下跌時與現實狀況是否相符，說明模型是否適合。看全部真實上漲和下跌比例
- 精確度(Precision)>> 衡量模型預測上漲時能否正確對應真實的漲跌。
只看那檔預測下來上漲的比例

四、 實驗說明

本實驗中將會使用 3 種 AI 模型(LSTM、Bi_LSTM、GRU)，以及 2 種統計模型(Logistic Regression、Bayesian Classifier)，針對五檔股票進行隔日股價上漲或下跌的預測，並判斷何種模型在準確度、執行時間...等的數據表現較為優秀。此五檔股票分別為 2330(台積電)、2379(瑞昱)、2412(中華電)、2912(統一超)、1101(台泥)。

五、 實驗結果

表 1-1 到 1-5 分別為五檔股票經過各模型訓練過後產生的數據，將各模型中的數據分別加總平均後的表請參考，表 2。

表 1-1 股票代號:2330 (台積電)

| 模型 | | 準確度 (Accuracy) | 回召率 (Recall) | 精確度 (Precision) | 執行時間 |
|----------|------------------------|-------------------|-----------------|--------------------|----------------|
| AI 模型 | LSTM | 0.67757 | 0.721649 | 0.625 | 0:08:49.572593 |
| | Bi-LSTM | 0.623932 | 0.84 | 0.538462 | 0:06:56.892865 |
| | GRU | 0.714953 | 0.907216 | 0.628571 | 0:06:51.194847 |
| 統計 模型 | Logistic Regression | 0.62963 | 0.542857 | 0.904762 | 0:00:00.029919 |
| | Bayesian Classifier | 0.432099 | 0.432099 | 1.0 | 0:00:00.007979 |

表 1-2 股票代號:2379(瑞昱)

| 模型 | | 準確度 (Accuracy) | 回召率 (Recall) | 精確度 (Precision) | 執行時間 |
|----------|------------------------|-------------------|-----------------|--------------------|----------------|
| AI 模型 | LSTM | 0.67757 | 0.632812 | 0.786408 | 0:09:18.994388 |
| | Bi-LSTM | 0.58547 | 0.848485 | 0.592593 | 0:07:16.259023 |
| | GRU | 0.67757 | 0.71875 | 0.736 | 0:06:53.957752 |
| 統計 模型 | Logistic Regression | 0.658436 | 0.642487 | 0.898551 | 0:00:00.023935 |
| | Bayesian Classifier | 0.567901 | 0.567901 | 1.0 | 0:00:00.006990 |

表 1-3 股票:2412(中華電)

| 模型 | | 準確度 (Accuracy) | 回召率 (Recall) | 精確度 (Precision) | 執行時間 |
|----------|------------------------|-------------------|-----------------|--------------------|----------------|
| AI 模型 | LSTM | 0.654206 | 0.377551 | 0.74 | 0:08:45.722578 |
| | Bi-LSTM | 0.641026 | 0.625 | 0.590909 | 0:07:16.590632 |
| | GRU | 0.714953 | 0.602041 | 0.728395 | 0:06:41.731589 |
| | Logistic | 0.670782 | 0.813953 | 0.327103 | 0:00:00.033868 |
| 統計 模型 | Regression | | | | |
| | Bayesian Classifier | 0.559671 | NaN | 0.0 | 0:00:00.007978 |

表 1-4 股票:2912(統一超)

| 模型 | | 準確度 (Accuracy) | 回召率 (Recall) | 精確度 (Precision) | 執行時間 |
|----------|------------------------|-------------------|-----------------|--------------------|----------------|
| AI 模型 | LSTM | 0.619718 | 0.410526 | 0.609375 | 0:08:49.081475 |
| | Bi-LSTM | 0.635193 | 0.524272 | 0.6 | 0:07:12.440295 |
| | GRU | 0.643192 | 0.494737 | 0.626667 | 0:06:40.902667 |
| 統計 | Logistic Regression | 0.665289 | 0.694915 | 0.394231 | 0:00:00.024932 |
| 模型 | Bayesian Classifier | 0.61157 | 0.727273 | 0.153846 | 0:00:00.007979 |

表 1-5 股票:1101 (台泥)

| 模型 | | 準確度 (Accuracy) | 回召率 (Recall) | 精確度 (Precision) | 執行時間 |
|----------|------------------------|-------------------|-----------------|--------------------|----------------|
| AI 模型 | LSTM | 0.577465 | 0.669643 | 0.585938 | 0:10:30.610998 |
| | Bi-LSTM | 0.60515 | 0.559322 | 0.622642 | 0:07:27.346519 |
| | GRU | 0.70892 | 0.848214 | 0.678571 | 0:08:05.848098 |
| 統計 | Logistic Regression | 0.640496 | 0.588235 | 0.97561 | 0:00:00.051862 |

| | | | | | |
|----|---------------------|-----|----------|---------|----------------|
| 模型 | Bayesian Classifier | 0.5 | 0.504167 | 0.98374 | 0:00:00.012965 |
|----|---------------------|-----|----------|---------|----------------|

透過圖 1、表 2，能看到準確度最高的為 GRU 模型(69%)，最差的為 Bayesian(僅有 53%)，此外 LSTM 及 Logistic 模型也有不錯的準確度(介於 65%上下)。另外透過圖 2、表 2，可以看出兩個統計模型花費的時間少上許多，而在 AI 模型中 GRU 所需的時間相較於另外兩者來的短，準確度也較高。

- 準確度：GRU > Logistic > LSTM > Bi-LSTM > Bayesian
- 執行速度：Bayesian > Logistic > GRU > Bi-LSTM > LSTM

雖然執行出的結果準確度最高只到 69%，但由於 AI 模型準確度會受到各項參數所影響，因此後續會再嘗試各種參數設定，以期望達到更高的準確度。

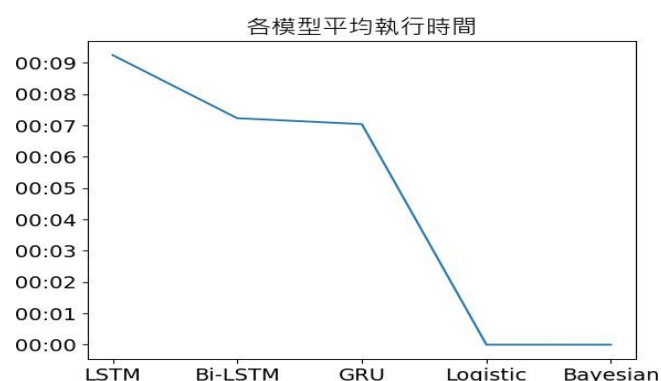
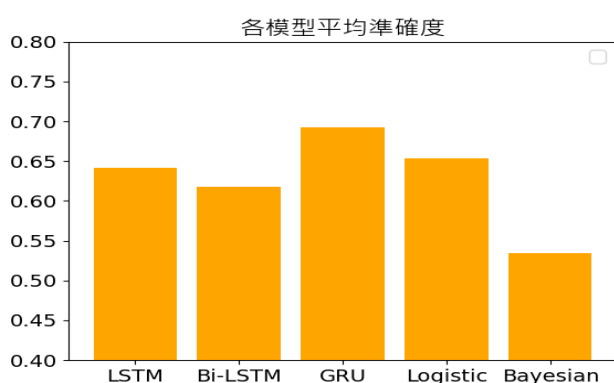


圖 1 各模型平均準確度

表 2 各模型平均數據

| 模型 | | 準確度 (Accuracy) | 回召率 (Recall) | 精確度 (Precision) | 執行時間 |
|----------|---------------------|-------------------|-----------------|--------------------|------------------------|
| AI 模型 | LSTM | 0.641306 | 0.562436 | 0.669344 | 00:09:14.796406 |
| | Bi-LSTM | 0.618154 | 0.679416 | 0.596073 | 00:07:13.905867 |
| | GRU | 0.691918 | 0.714192 | 0.679641 | 00:07:02.726990 |
| 統計 模型 | Logistic Regression | 0.652927 | 0.656489 | 0.700051 | 00:00:00.032903 |
| | Bayesian Classifier | 0.534248 | 0.446288 | 0.627517 | 00:00:00.008778 |

