Assignment 1: Classification Report

Student ID: 113552016, Name: 陳帝嘉

1. (5%) **Explain the meaning of Precision, Recall, F1 score** and discuss how it helps in understanding the performance of the model.

精確率(**Precision**)是指模型預測為正類的樣本中，實際是正類的比例。換句話說，精確率衡量的是在所有被模型預測為正的樣本中，有多少是真正的正樣本。

$$\text{精确率} = \frac{\text{真正类（True Positives, TP）}}{\text{真正类（TP）} + \text{假正类（False Positives, FP）}}$$

召回率(**Recall**)是指实际为正类的样本中，模型成功预测为正类的比例。也就是说，它反映了模型在所有正类样本中能够识别出多少。

$$\text{召回率} = \frac{\text{真正类（True Positives, TP）}}{\text{真正类（TP）} + \text{假负类（False Negatives, FN）}}$$

F1 分数是精确率和召回率的调和平均数，用来平衡二者的影响。当精确率和召回率的差距较大时，F1 分数提供了一个综合的评价指标。

$$F1 = 2 \times \frac{\text{精确率} \times \text{召回率}}{\text{精确率} + \text{召回率}}$$

2. (5%) **Preprocessing & Feature Engineering**: What preprocessing (e.g. normalization) and feature engineering exhibit the better performance.

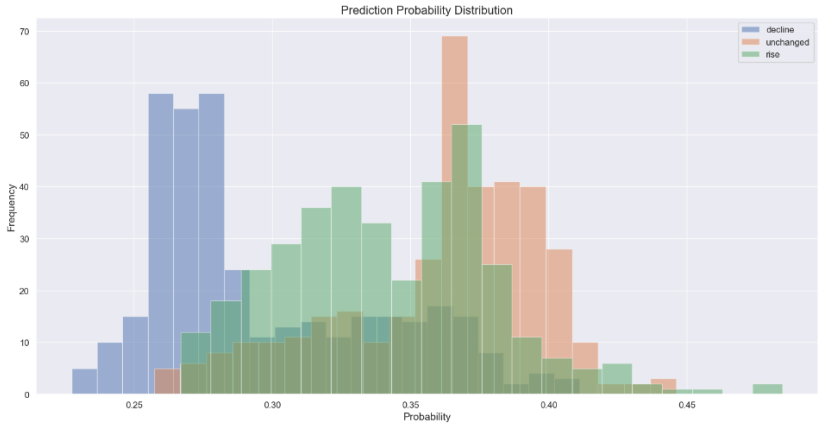
在 SVM 裡面需要對資料做標準化，但是在樹類模型不需要，這裡使用 StandardScaler 是用平均值和標準差作為標準化的過程。

3. (5%) **What models have you implemented?** How do you modify these models to make them perform well on this task? What is the difference between these models? Please answer these questions. Example Table

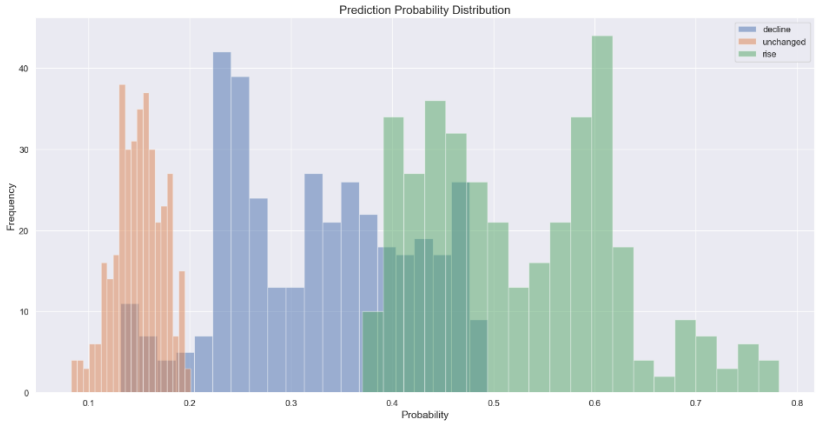
	1_trend	5_trend	10_trend
SVM	改變 kernal rbf , poly, linear	改變 kernal rbf , poly, linear	改變 kernal rbf , poly, linear
Catboosting	改變 n_estimators, max_depth, learning_rate	改變 n_estimators, max_depth, learning_rate	改變 n_estimators, max_depth, learning_rate

4. (5%) **Does longer history data present better performance?** Say, using data from the previous 60 days versus the previous 30 days. Explain the possible reasons for the differences.
我嘗試 catboosting 60 天在 validation data 表現較 30 天資料差，使用過長的歷史資料反倒成反效果，越長的資料提供越多的訊息，但是過長的資料，可能會造成有許多的雜訊，導致效果變差。

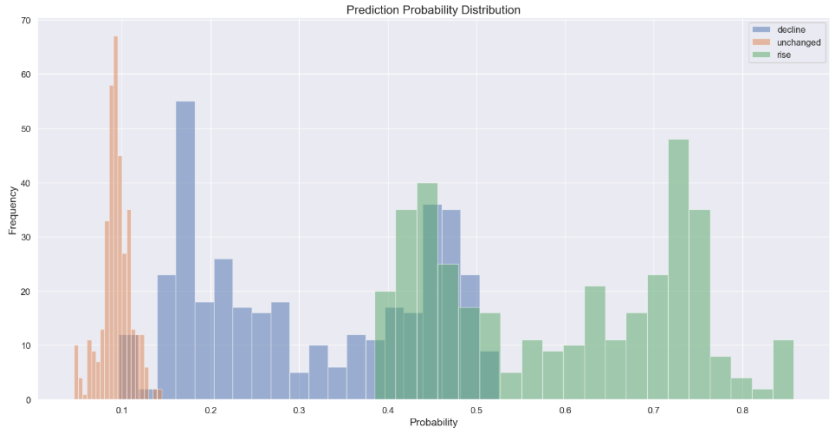
5. (5%) **Compare performance of the 3 tasks for the next 1, 5, and 10 days**.
Example Charts
1_trend:



5_trend:



10_trend:



6.(5%) **Can we train three models together, e.g. multitasking?** Compare the performance for independent training and parallel training

可以同時訓練模型做 multitasking，但是就會有 3*3*3 種可能性，也就是說
(1,1,1),(1,1,0), (1,1,-1), (1,0,1),(1,0,0), (1,0,-1), (1,-1,1),(1,-1,0), (1,-1,-1)
(0,1,1),(0,1,0), (0,1,-1), (0,0,1),(0,0,0), (0,0,-1), (0,-1,1),(0,-1,0), (0,-1,-1)
(-1,1,1),(-1,1,0), (-1,1,-1), (-1,0,1),(-1,0,0), (-1,0,-1), (-1,-1,1),(-1,-1,0), (-1,-1,-1)
我想單獨訓練模型的效果應該會比較好，因為，對於模型來說要分別較多種可能性的效果可能不如專注於預測一種結果的模型。