國立中山大學應用數學所

統計學習與資料探勘

期末報告

# 台灣各小時空氣指數狀態資料分析--分類分群

# Taiwan's Air Quality Data

M112040026 黃偉柏

M112040010 溫宏岳

民國 111 年 12 月 30 日

# 摘要

近年來受細懸浮微粒 PM2.5 增加的影響，導致台灣的空氣品質越來越差，加上冬季容易產生霧霾，這些因素都造成我們容易有身體的狀況產生。

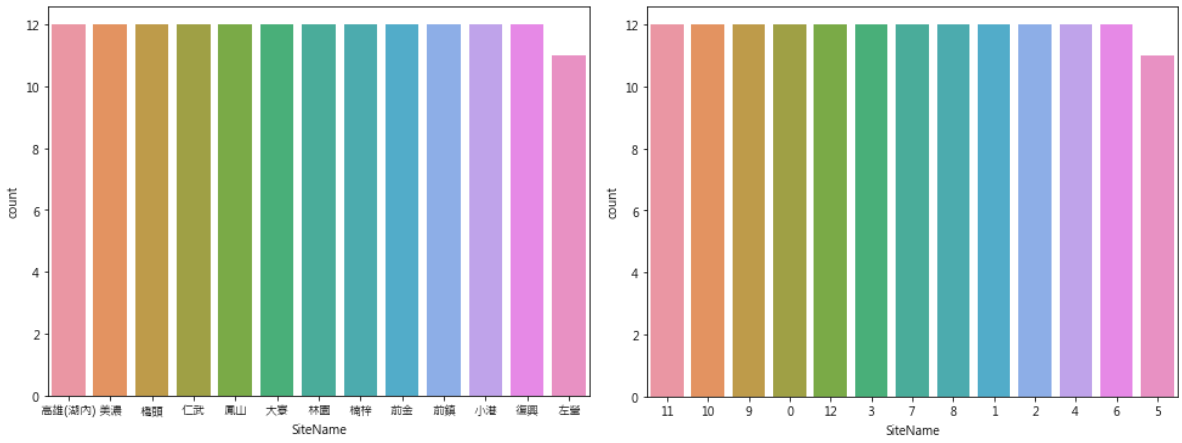
本次報告藉由多種分類的演算法，對空氣的品質做三元分類的預測，分為對敏感族群不健康、普通與良好，並著重在分類的準確率上。

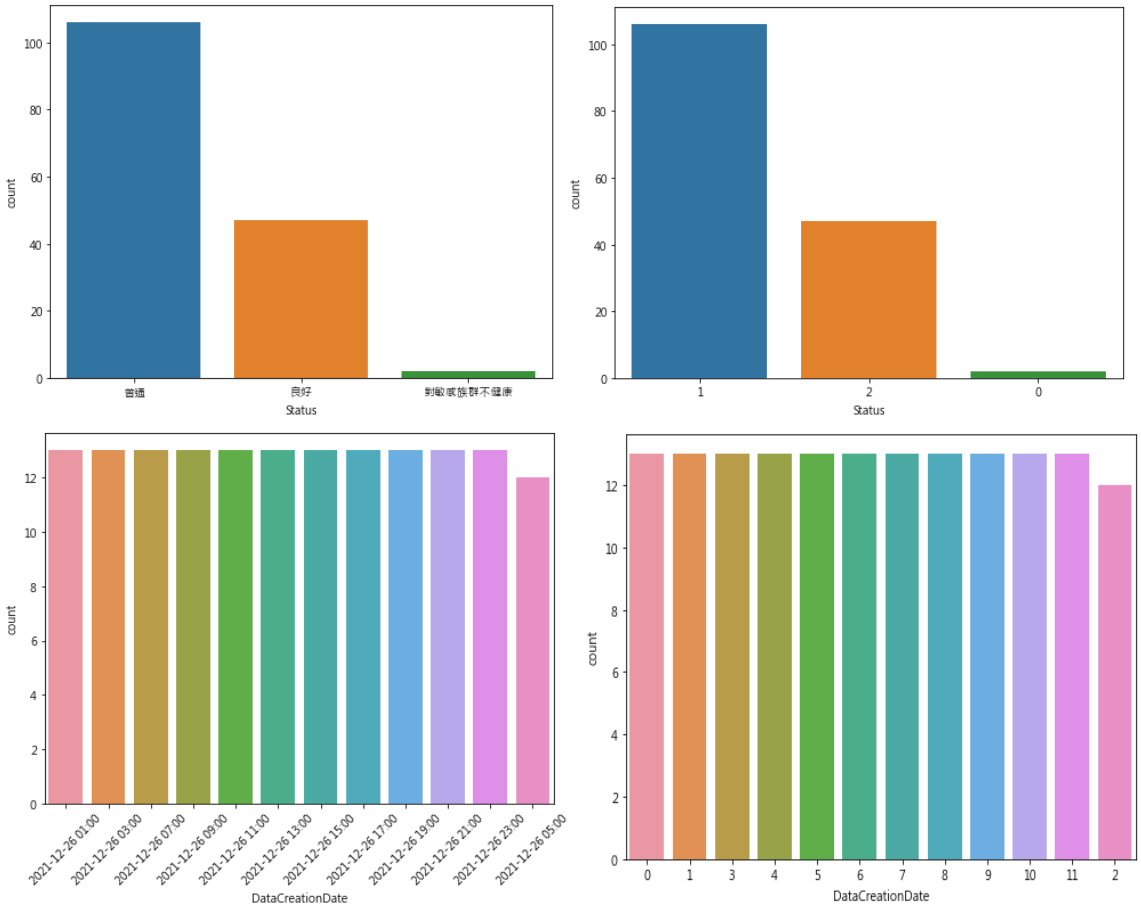
# 數據集

此份數據集，原有 3517803 筆，經過刪除、篩選剩下 155 筆。其中包含 18 個定量變數，分別為 SiteName、AQI、SO2、CO、O3、O38hr、PM10、PM25、NO2、NOx、NO、WindSpeed、WindDirec、DataCreationDate、CO_8hr、PM25AVG、PM10_AVG、SO2AVG，與目標變數 Status(空氣品質對敏感族群不健康:0，普通:1，良好:2)

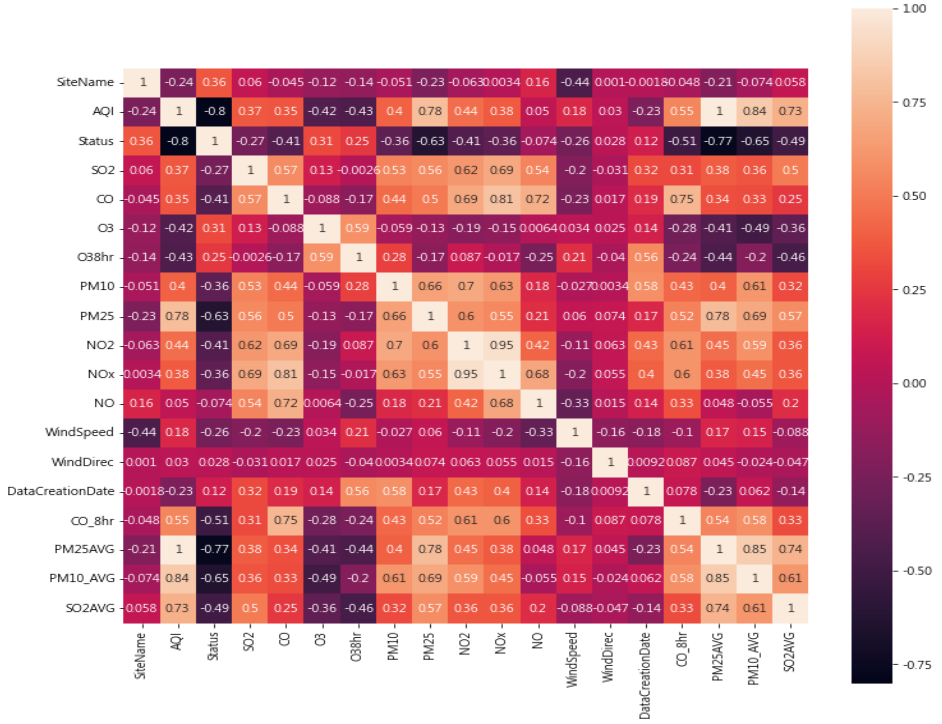
# EDA 與前處理

1. 首先對較多缺失值的 Pollutan、Unit、Longitude、Latitude、SiteId、County 刪除，再去 dropna。
2. 篩選高雄市、日期為 2021-12-26。
3. 下圖為轉換 SiteName、Status、DataCreationDate 編碼。

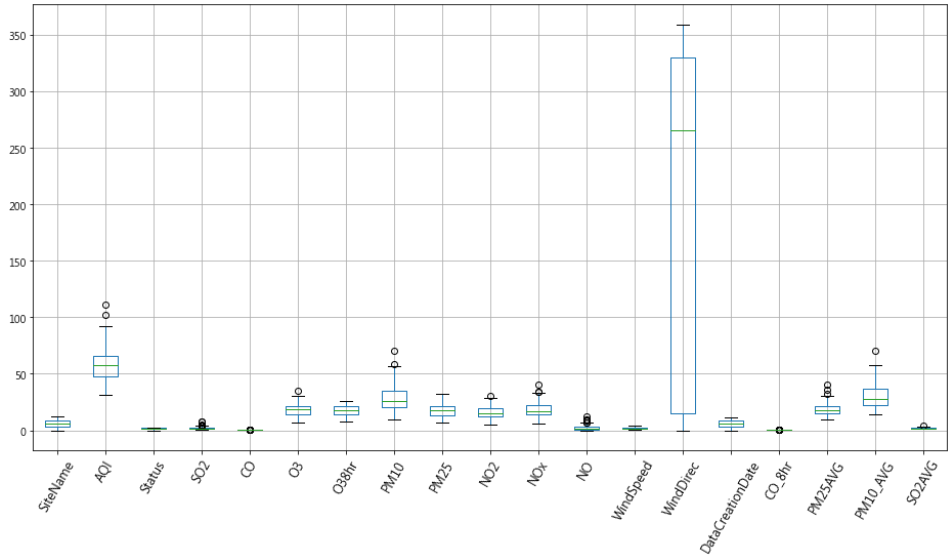





4. 下圖 heatmap 我們採|相關係數| ≥ 0.5 的欄位，所以 feature(X) = AQI PM25、CO_8hr、PM25AVG、PM10_AVG、SO2AVG，target(y) 採取 Status。



5. 下圖 Boxplot 是來判斷 outliers，雖然有 outliers 但沒有很多，所以我們仍採用真實資料。




# 方法(使用的模型)

## Supervised Learning

- Linear Classification

1.LDA:希望投影後的資料，組內分散量(within-class scatter)越小越好，組間分散量(between-class scatter)越大越好。

- Nonlinear Classification

1.Decision Tree:可以同時處理連續型與類別型變數，不需要進行太多的資料預處理（Preprocessing）。

2.Xgboost:除了可以做分類也能進行迴歸連續性數值的預測，而且效果通常都不差。並透過 Boosting 技巧將許多弱決策樹集成在一起形成一個強的預測模型。利用了二階梯度來對節點進行劃分 利用局部近似算法對分裂節點進行優化 在損失函數中加入了 L1/L2 項，控制模型的複雜度 提供 GPU 平行化運算。

## Unsupervised Learning

- Clustering:

1.Kmeans: 可以非常快速地完成分群任務，但是如果觀測值具有雜訊（Noise）或者極端值，其分群結果容易被這些雜訊與極端值影響，適合處理分布集中的大型樣本資料。K-means 運作的流程：

1. 先設定要分幾群（K 群）
2. 隨機找 K 個樣本點做出使得群的“重心”
3. 對每個資料計算距離這 K 個重心的距離，最後該筆資料分給距離最短的那群
4. 每筆資料都被分到 k 群中，對每一群的資料計算重心
5. 重複直到收斂

➢ Dimension Reduction

1.PCA:將座標軸中心移至數據集的中心，利用旋轉座標軸，使數據在 C1 軸的變異數最大，以保留更多信息，C1 即為第一主成分。 找一個與 C1 主成分的共變異數為 0 的 C2 主成分，以避免信息重疊。

主成分分析經常用於減少數據集的維數，同時保留數據集當中對變異數貢獻最大的特徵。優點：以變異數為衡量信息量的指標，不受數據集以外的因素影響。 用正交轉換方式，可消除數據成分間相互影響的因素。

缺點：主成分間的特徵維度較難解釋。 容易捨棄一些也帶有信息量的特徵，分析結果可能會受影響。

# 實驗與結果

首先將資料拆分為 train data 80%、test data 20%。

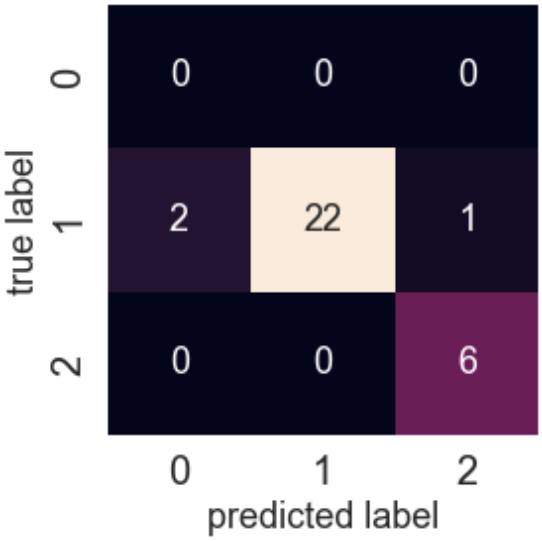
並選擇 random_state = 4 的情況下分別對監督式與非監督式學習的模型做三元分類的討論。

## 監督式學習

### ➢ 線性分類:

LDA:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.00 | 0.00 | 0.00 | 2 |
| 1 | 0.88 | 1.00 | 0.94 | 22 |
| 2 | 1.00 | 0.86 | 0.92 | 7 |
| accuracy | | | 0.90 | 31 |
| macro avg | 0.63 | 0.62 | 0.62 | 31 |
| weighted avg | 0.85 | 0.90 | 0.87 | 31 |



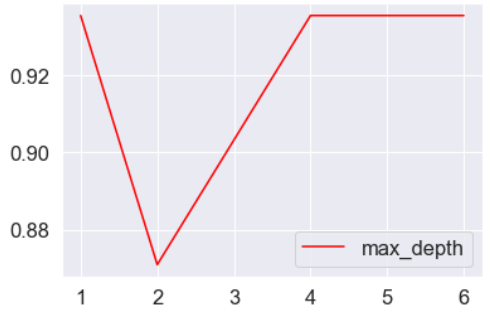

我們第一個選擇的模型是線性判別分析，會選擇的原因是想試圖找到三類的特徵所組的一個線性組合，以能夠特徵化或區分它們。所得的組合可用來作為一個線性分類器。

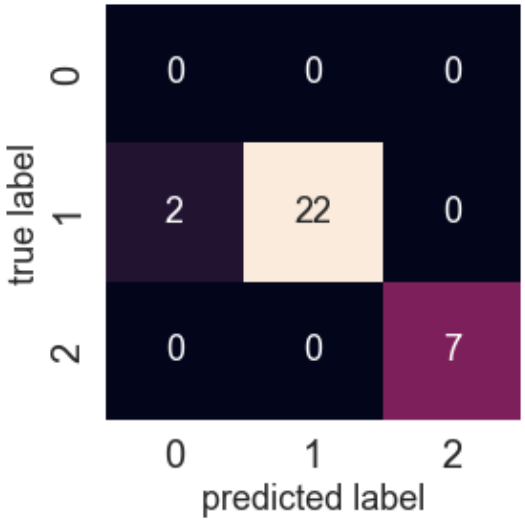
經過模型運算後得到 Train Accuracy = 96.774 %，而 Test Accuracy = 90.322 % 是相當不錯且沒有 overfitting 的。

## ➢ 非線性分類

### Decision tree:

| | Max_Depth | Accuracy |
|---|---|---|
| 0 | 2.0 | 0.870968 |
| 1 | 3.0 | 0.903226 |
| 2 | 4.0 | 0.935484 |
| 3 | 5.0 | 0.935484 |
| 4 | 6.0 | 0.935484 |




| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.00 | 0.00 | 0.00 | 2 |
| 1 | 0.92 | 1.00 | 0.96 | 22 |
| 2 | 1.00 | 1.00 | 1.00 | 7 |
| accuracy | | | 0.94 | 31 |
| macro avg | 0.64 | 0.67 | 0.65 | 31 |
| weighted avg | 0.88 | 0.94 | 0.90 | 31 |

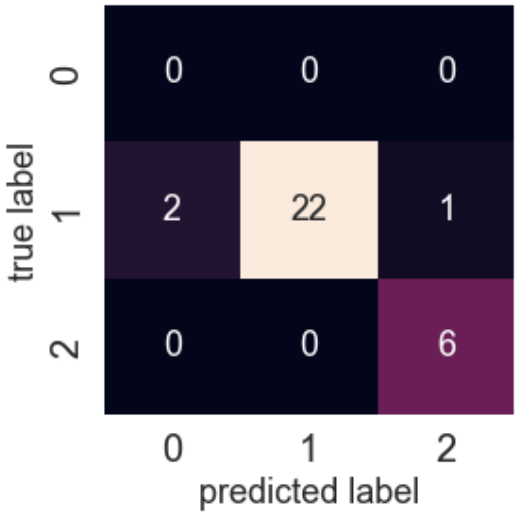
接下來選擇非線性模型的原因是大多數問題還是非線性的。而選擇 decision tree 是因為它同時處理連續型與類別型變數，不需要進行太多的資料預處理，但為了解決超參數可能造成的 overfitting 問題，用 loop 來選擇最適合的 Max depth 來做限制，也就是上圖在 Max_depth = 4 的時候就可以達到不錯的效果。

接著建一顆新的樹後將所得超參數代入模型，可以獲得 Train Accuracy = 100 %，而 Test Accuracy = 93.548 % 也是滿不錯的，同時也比 LDA 來得高。

### XGBoost:

```
Fitting 5 folds for each of 54 candidates, totalling 270 fits
Best parameters: {'colsample_bytree': 0.3, 'learning_rate': 0.01, 'max_depth': 3, 'n_estimators': 100}
```

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.00 | 0.00 | 0.00 | 2 |
| 1 | 0.88 | 1.00 | 0.94 | 22 |
| 2 | 1.00 | 0.86 | 0.92 | 7 |
| accuracy | | | 0.90 | 31 |
| macro avg | 0.63 | 0.62 | 0.62 | 31 |
| weighted avg | 0.85 | 0.90 | 0.87 | 31 |



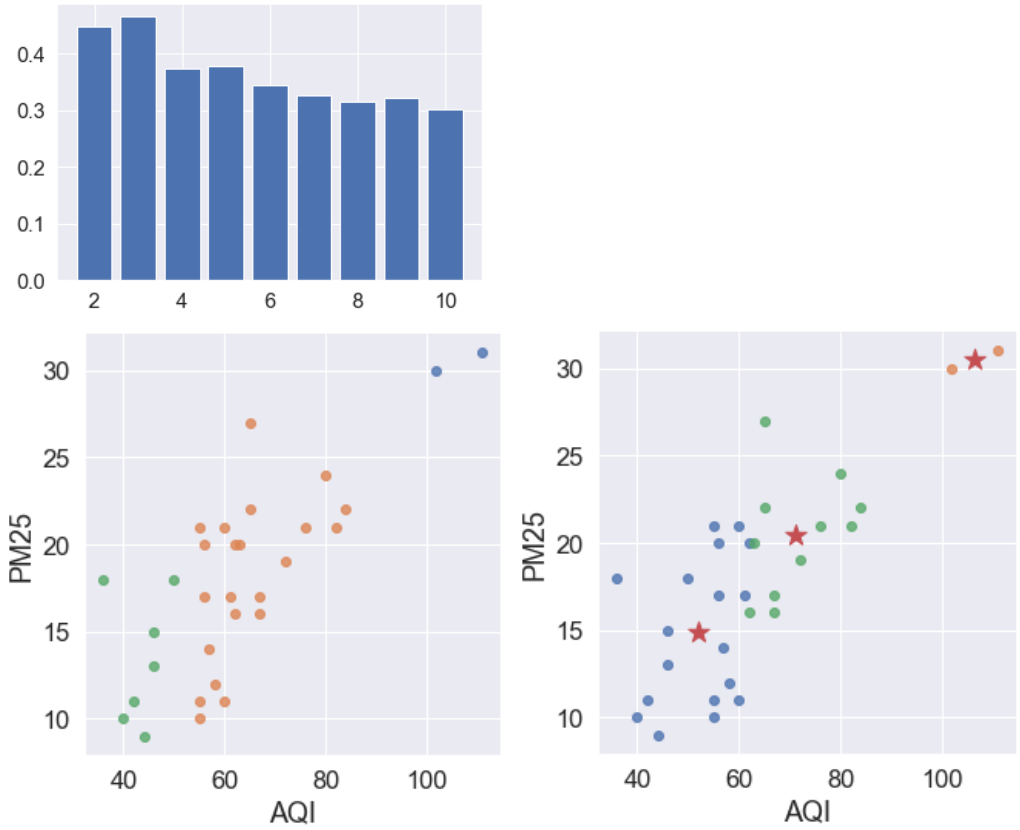
選擇 XGBoost 是因為它除了可以做分類也能進行迴歸連續性數值的預測，而且效果通常都不差。並透過 Boosting 技巧將許多弱決策樹集成在一起形成一個強的預測模型。接著我們利用 GridSearchCV 找需要的超參數(如上方) 避免造成 overfitting 的問題，再套入模型。

Train Accuracy = 100%，Test Accuracy = 90.322 % 與前兩個相去不遠，但因為梯度提升，修正前一棵樹的誤差較有隨機性，所以相對 decision tree 有一些不確定性。

## 非監督式學習

- Clustering

KMeans




由於教授上課有提到一些，但課程後面並沒有太多練習的機會，因此想試著做看看分群的成果。

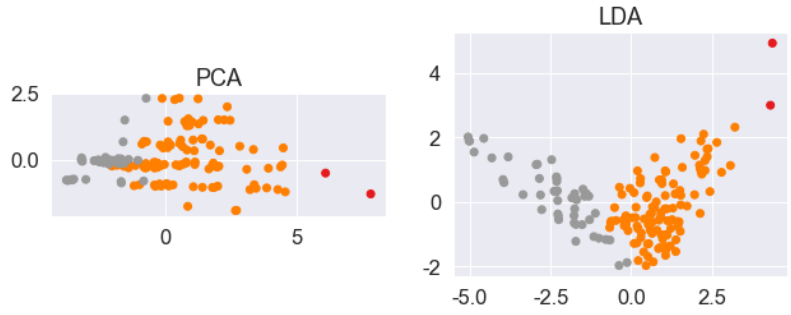
首先，將資料 label 標前移除後分群，但演算法有個最大的缺點，就是要選定 K，所以接著做 Silhouette 輪廓分析，輪廓係數法的概念是「找出同群資料點內最近/不同群越分散」的值，也就是滿足 Cluster 的定義，分析完後，就用 for 迴圈產生不同的 n_clusters 去

看看何者輪廓係數較高，透過上圖可以發現 在 n_clusters =3 的時候，分的效果最好，所以可以選定 3 當作 K。

接著將 n_clusters=3 代入模型會得到上圖的兩個對比圖。左圖為測試集原始的 AQI 與 PM2.5 迴歸圖，會選這 2 個當 x、y 軸是因為他們比較相關，拿來作圖比較好判斷。

右圖為預測出來的，紅色星型點為各自分群的中心點，我們可以發現右上角距離其他 2 群分的最好，而藍、綠色點之間有些預測的不太好。但由於是非監督式學習的分群問題，無法用準確率來衡量，因此如同上述只是做來練習的成果，或許對未來需要做分群會有些幫助。

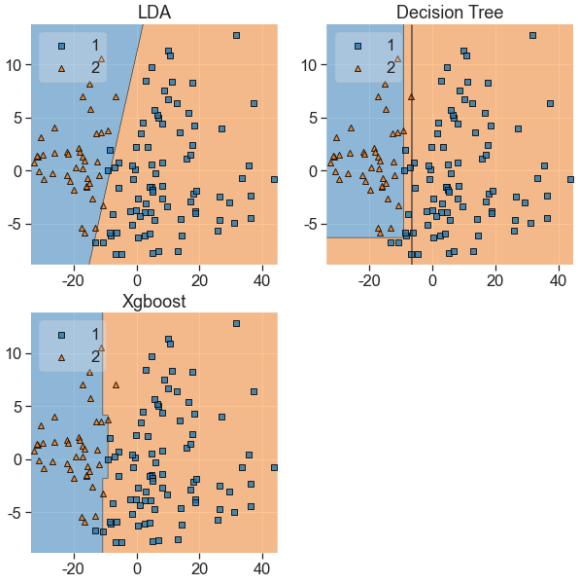
## ➢ Dimension Reduction




這裡簡單的對 PCA 和 LDA 取 n_components = 2 降維，並由上圖可發先 LDA 區分的比較好一些。

## ➢ 利用 PCA 降至 2 維後分別做 LDA、Decision Tree、XGBoost 模型

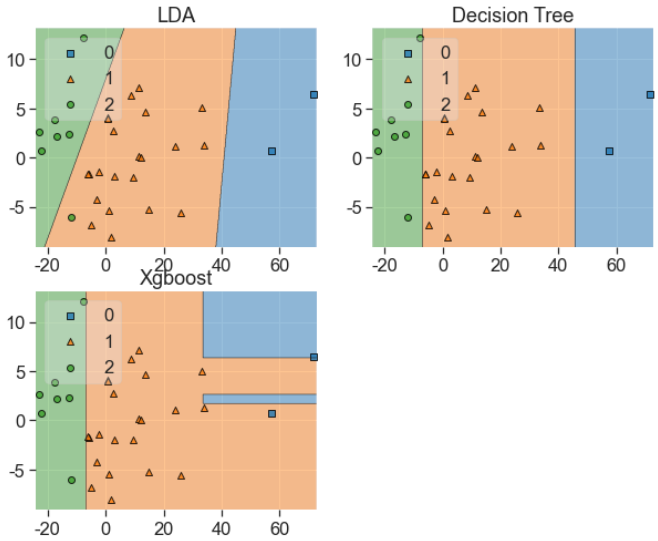
```
LinearDiscriminantAnalysis() Train Accuracy: 0.9758064516129032
DecisionTreeClassifier(max_depth=4, random_state=4) Train Accuracy: 1.0
XGBClassifier(colsample_bytree=0.3, learning_rate=0.01) Train Accuracy: 0.9838709677419355
```




(Train Data)

由上圖可以發現 Train Accuracy 的大小為 Decision Tree > XGBoost > LDA，說明降維後在訓練集用非線性的模型會有比較好的正確率。其中 Decision Tree 可以完全正確的判斷與分類。

```
LinearDiscriminantAnalysis() Test Accuracy: 0.967741935483871
DecisionTreeClassifier(max_depth=4, random_state=4) Test Accuracy: 1.0
XGBClassifier(colsample_bytree=0.3, learning_rate=0.01,
              objective='multi:softprob') Test Accuracy: 0.967741935483871
```




(Test Data)

由上圖可以發現 Test Accuracy 的大小為 Decision Tree > XGBoost = LDA，說明降維後在測試集用非線性的模型會有好一點點的正確率。其中 Decision Tree 甚至可以完全正確的判斷與分類。

## 比較

由以上 6 種模型中選擇 1 個最好的，如果我們著重在 Test Accuracy 上，無論是原始或使用 PCA 後的模型皆是 Decision tree 為最好、最準確的預測模型。雖然幾乎經過 PCA 後的模型都得到了準確率的提升，但如果考慮我們所選的主要影響的 6 個變數的話，依然還是 Decision tree 為最佳的模型。

| | Origin | | | After PCA | | |
|---|---|---|---|---|---|---|
| Model/ Accuracy | LDA | Decision tree | XGBoost | LDA | Decision tree | XGBoost |
| Train Accuracy | 96.774 % | 100 % | 100 % | 97.581 % | 100 % | 98.387 % |
| Test Accuracy | 90.322 % | 93.548 % | 90.322 % | 96.774 % | 100 % | 96.774 % |

# 結論與未來工作

我們發現以下幾個問題:

1. 部分資料集過少 。
2. Outliers 沒有多做處理。
3. 當資料集過少可能導致 Random state 的選擇造成資料不平衡。

結論且在未來可改善的:

1. 盡量選擇分類目標較平均的資料集，這樣在處理上比較不花時間。但這種不平衡的資料或許也比較貼近現實狀況，因此也是值得去做一些處理與討論的。
2. Outliers 的部分可試著多處理提升準確度，但相對的就比較不貼近現實狀況。
3. 若 Random state 讓訓練與測試的資料集較為平衡 ，後續可以多做處理，這次剛好討論的是不平衡資料都跑到測試集去的情況，以後可以多加注意這部分。
4. Decision tree 是此分類問題最佳的模型。
5. 可嘗試將此三元分類整理合併為二元分類問題，再加上可用 ROC 曲線、AUC 分數(面積)來做更加一步的解釋。

# 貢獻

| | |
|---|---|
| 黃偉柏 | 資料處理、PPT 製作 |
| 温宏岳 | 模型預測、程式碼編寫 |

# 參考文獻


1. Kaggle: https://www.kaggle.com/datasets/yenruchen/taiwans-air-quality-data-by-hours
2. 行政院環保署空氣品質監測網: https://airtw.epa.gov.tw/cht/Information/Standard/AirQualityIndicator.aspx