國立屏東大學

111 學年度

機器學習期末報告

學生：吳俊易（CBB110112）

中　華　民　國　112　年　6　月

1. 資料集合一 (final_project_dataset_1.csv)

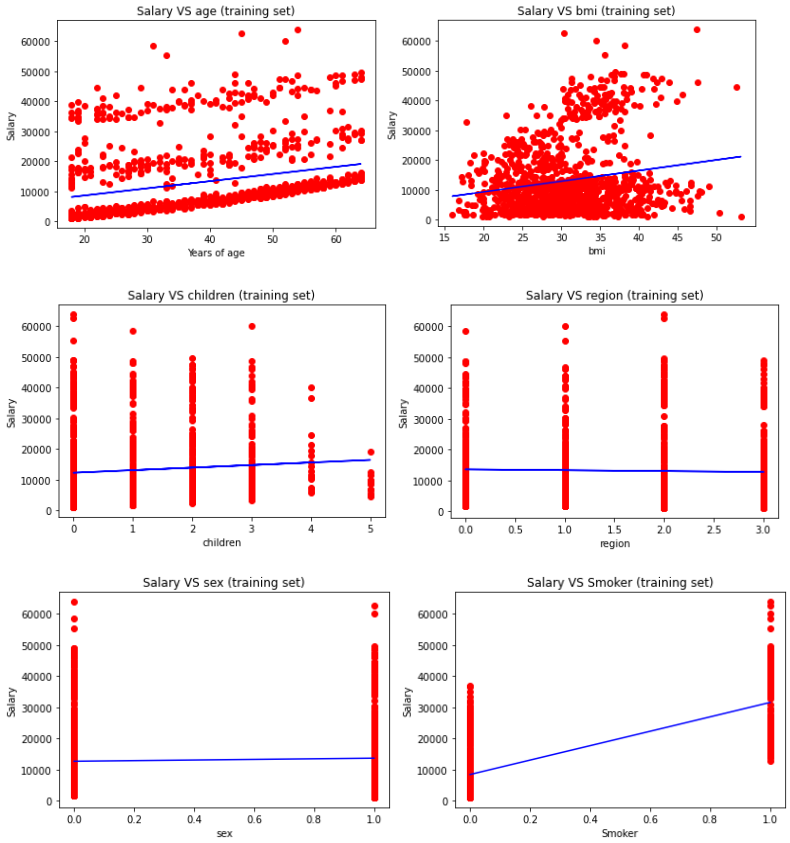
前情提要：因為應變量(y)指定用連續型變數，

所以用回歸模型來分析效果較佳。

- Simple Linear Regression(簡單回歸分析)

公式：$Y_i = \beta_0 + \beta_1 X_i$

以公式來判斷，此模型的分析效果，無法呈現出最完美的正確率及預測線。



分析結果：

charges vs age: 圖片顯示，當被保險人年紀越大，保險費越高。

charges vs bmi: 圖片顯示的情形無法解釋出線性回歸的狀態。

charges vs other: 剩下的自變量都是分類型欄位，只能看出在某些條件下的保險費是多少。

參數調整：測試集設定為 0.2。

模型建立：使用 sklearn 的套件產生，也可使用數學式來產生。

為何採用該模型：循序漸進，先由最簡單的迴歸分析開始分析資料集。

- Multiple Linear Regression(多元回歸分析)

  公式：$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} + \epsilon$

  以公式來看最能表現出此資料集的回歸分析。

  圖例：此模型所呈現的是 3 維圖形，必須有專業參數。

  模型建立：同樣使用 sklearn 來幫助模型建立，也能使用數學式；
  使用反向淘汰法，移除多餘的項防止干擾數學式分析。

  反向淘汰法：P|t| > 0.05 時，就淘汰此項(挑最大的項)，淘汰到沒有項的 P|t| > 0.05 就停止。

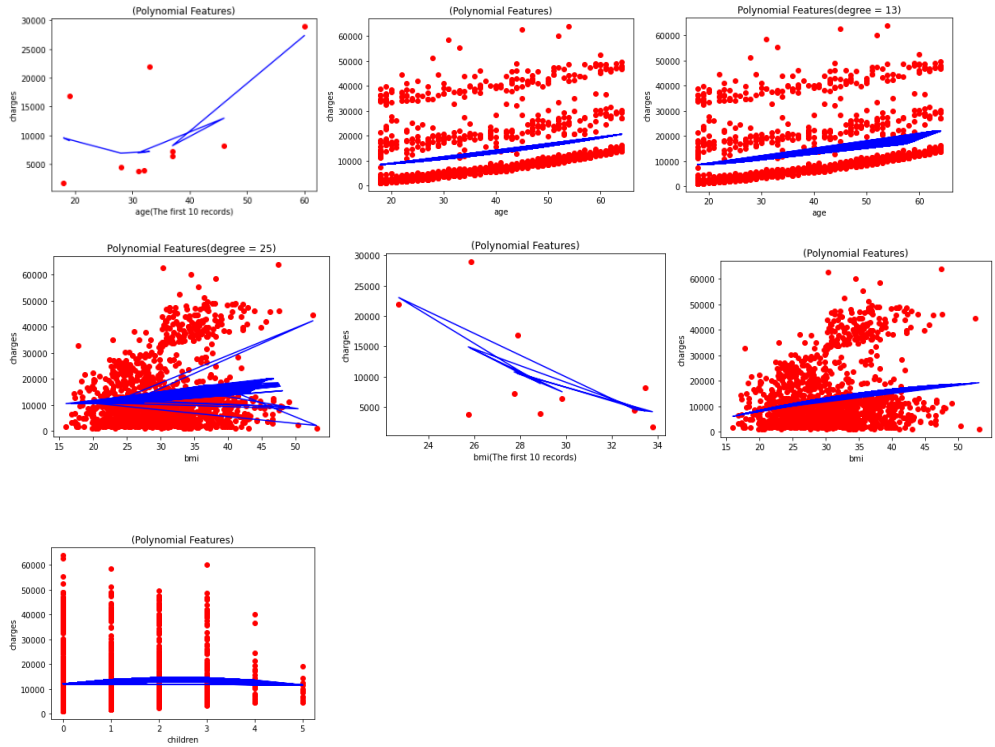
  參數調整：測試集設定為 0.2。

  最佳分析結果：sex、children、charges

  為何採用該模型：由數學式和資料集判斷，覺得可以藉由誤差項調整回歸線。

- Polynomial Regression(多項式回歸分析)

公式：$y = b_0 + b_1x_1 + b_2x_1^2 + ... + b_nx_1^n$



分析結果：

charges vs age: 圖片顯示，當被保險人年紀越大，保險費越高；比較前 10 筆資料跟全部資料所算出來的回歸線，會出現此情況，估計是要算 b0~bn 時，梯度下降法無法到達真正的絕對低點所導致而成的，不建議用此模型。

charges vs bmi: 梯度下降法造成的回歸線混亂的情況更為明顯，看不出多項式回歸的回歸線，不建議用此模型。

charges vs other: 剩下的自變量都是分類型欄位，只能看出在某些條件下的保險費是多少；同上不建議用此模型。

模型建立：同樣使用 sklearn 來幫助模型建立，也能使用數學式。

參數調整：測試集設定為 0.2；degree 基本值為 2(有較大值有標記在圖片上)；degree 設定到一定的大小時，會到極限(即不會在變化)。

最佳分析結果：無

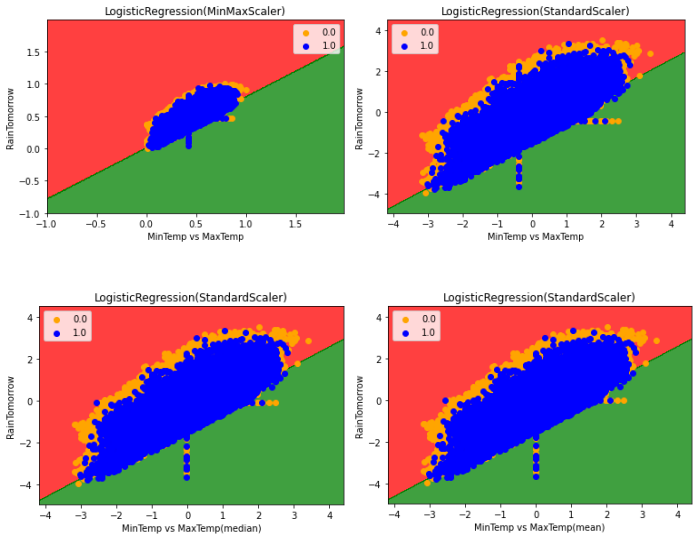
為何採用該模型：觀察梯度下降法，並記錄可能發生的問題。

2. 資料集合二 (final_project_dataset_2.csv)

- Logistic Regression

公式： $\frac{e^{(\beta_0+\beta_1 x)}}{1+e^{(\beta_0+\beta_1 x)}}$

1. MinTemp and MaxTemp



分析結果：

從圖片中的分類能得出在甚麼溫度下(分類效果不好)，

隔天是有可能會下雨；也有沒有下雨的機會，不會下雨的範圍大於會下雨；不管使用甚麼方式填補缺失值，結果大致相同。

模型建立：同樣使用 sklearn 來幫助模型建立，也能使用數學式。

參數調整：測試集設定為 0.2，缺失值(mean, median, most frequent)。

特徵縮放(StandardScaler,MinMaxScaler)之間的差異：

$$z = \frac{x - \mu}{\sigma}$$

StandardScaler

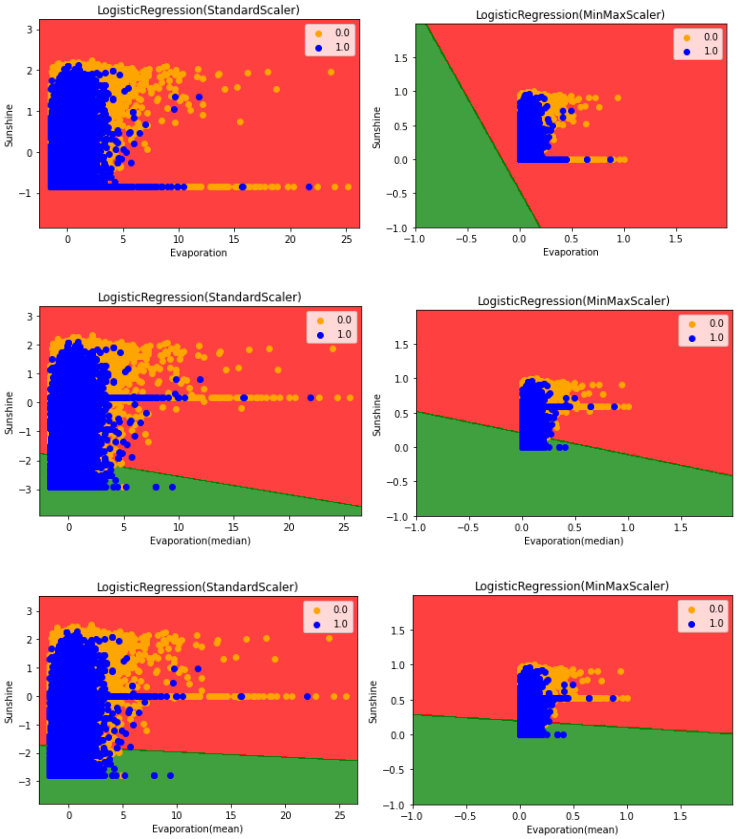
$$X_{new} = \frac{X_i - \min(X)}{\max(x) - \min(X)}$$

MinMaxScaler

StandardScaler: 標準化，根據資料內容進行特徵縮放。

MinMaxScaler: 將資料內容特徵縮放成 0~1 的數值。

2. Evaporation and Sunshine



分析結果：

以日蒸發量與照射時長來進行預測隔天是否下雨(分類效果不好)，

這 2 個自變量的缺失值較多，所以會隨著填補資料的方式而改變結果；

特徵縮放的方式也會改變分析結果，所以不太能當作參考依據

模型建立：同樣使用 sklearn 來幫助模型建立，也能使用數學式。

參數調整：測試集設定為 0.2，缺失值(mean, median, most frequent)。

特徵縮放(StandardScaler,MinMaxScaler)之間的差異：

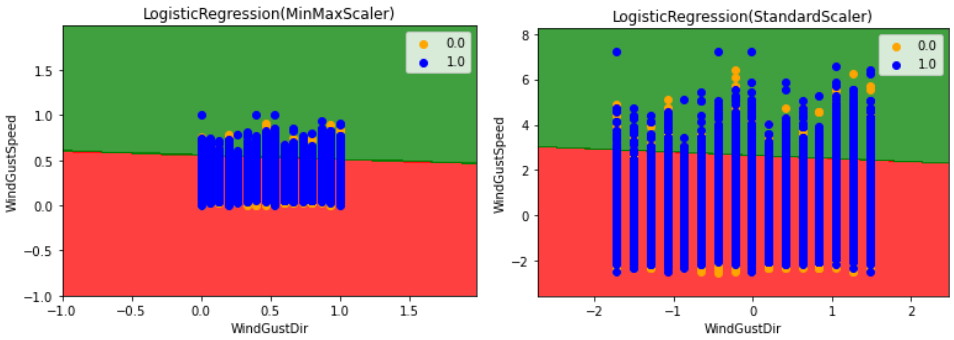
$$z = \frac{x - \mu}{\sigma} \qquad X_{new} = \frac{X_i - \min(X)}{\max(x) - \min(X)}$$

StandardScaler　　MinMaxScaler

StandardScaler: 標準化，根據資料內容進行特徵縮放。

MinMaxScaler: 將資料內容特徵縮放成 0~1 的數值。

3. WindGustDir and WindGustSpeed



分析結果：

以最強陣風的方向和速度（km / h）預測明天會不會下雨(分類效果不好)；因為有一個自變量是分類型欄位，所以能看出在甚麼風向及風速下，明天是否下雨。

模型建立：同樣使用 sklearn 來幫助模型建立，也能使用數學式。

參數調整：測試集設定為 0.2，缺失值(most frequent)，特徵縮放。

特徵縮放(StandardScaler,MinMaxScaler)之間的差異：

$$z = \frac{x - \mu}{\sigma}$$

StandardScaler

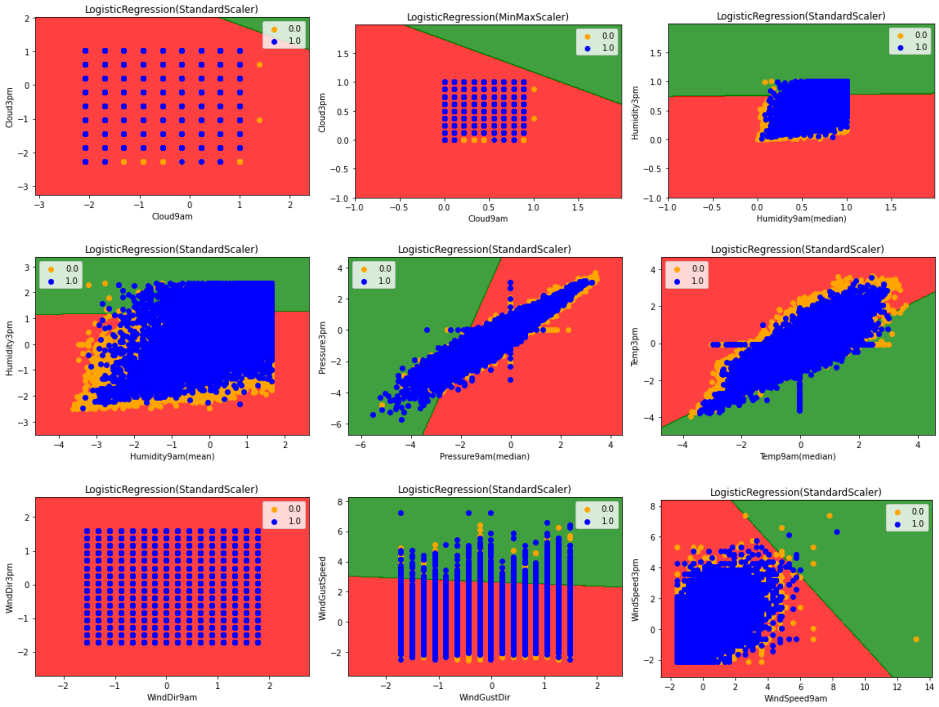
$$X_{new} = \frac{X_i - \min(X)}{\max(x) - \min(X)}$$

MinMaxScaler

StandardScaler: 標準化，根據資料內容進行特徵縮放。

MinMaxScaler: 將資料內容特徵縮放成 0~1 的數值。

4. 9AM 跟 3PM 當自變量所出來的分析圖

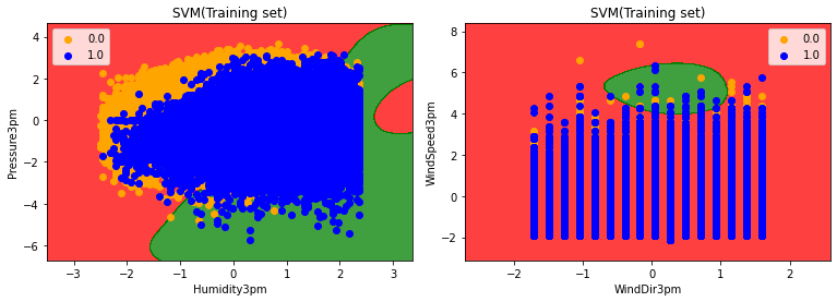


根據上述的圖片來看，用 9AM 跟 3PM 來分析並不是很恰當，分類的結果也是差強人意，也看出來邏輯回歸不太適合分類此類型的資料集。

- SVM (支援向量機)

  公式:

$$min_w \lambda \parallel w \parallel^2 + \sum_{i=1}^{n} (1 - y_i \langle x_i, w \rangle)_+$$



從這 2 張圖來看， SVM 的分析結果稍好於 Logistic Regression ，曲線描繪效果較佳，哪個區間會不會降雨雖然還是較模糊，但還是能大致看出圖帶來的意義。

比較 MinTemp and MaxTemp (svm vs Logistic Regression):

	0	1
0	21826	241
1	5750	622

	0	1
0	21648	419
1	5503	869

藉由混淆矩陣來觀察可以看出 TP(00) 的位置 (SVM 佳)、TN (11)的位置 (SVM 較少)，但在 FN 上較少的則是 (Logistic Regression)，呈現型 2 錯誤；型 1 錯誤較多者則是(Logistic Regression)，所以我們必須考慮此變數的分析適合可以原諒哪種錯誤，還有一種可能是缺失資料太多導至而成。

比較 WindGustDir and WindGustSpeed (svm vs Logistic Regression):

	0	1
0	21812	255
1	6062	310

	0	1
0	21835	232
1	6079	293

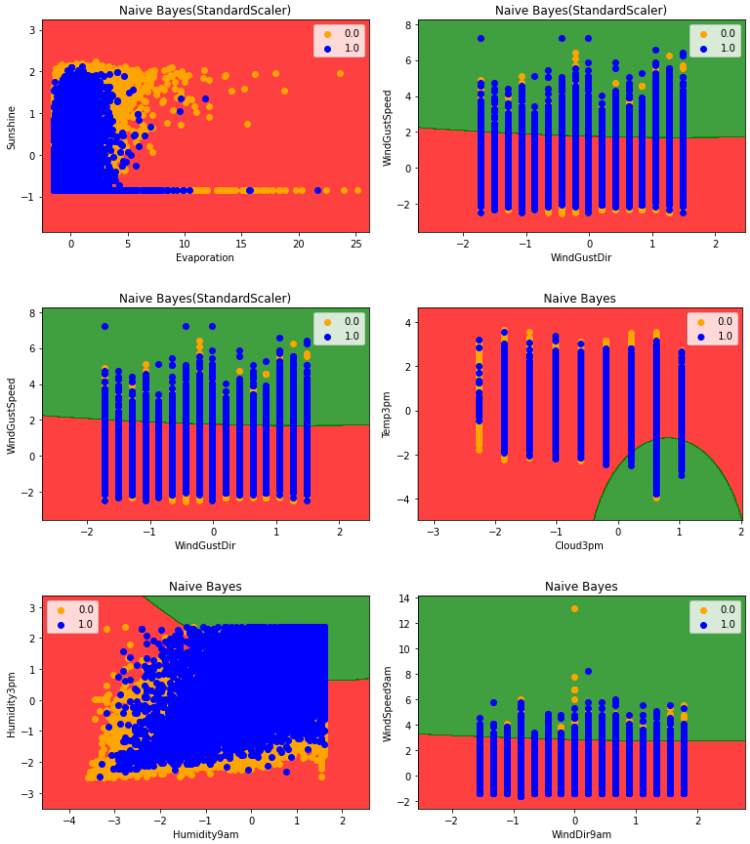
這組變數的型 1 錯誤與型 2 錯誤都是(SVM 較高)，但其實差異不大，無法太明確的判斷說哪個模型較好，因為還是有缺失值過多進而影響整體的問題存在！

分析結果：svm 在某些情況下會好於邏輯回歸，但缺失資料太多，所以無法那麼明確的得知 svm 的分析結果是否好於 邏輯回歸。
模型建立：使用 sklearn 來幫助模型建立。
參數調整：測試集設定為 0.2

- Naive Bayes (貝式分類器)

  公式:

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$



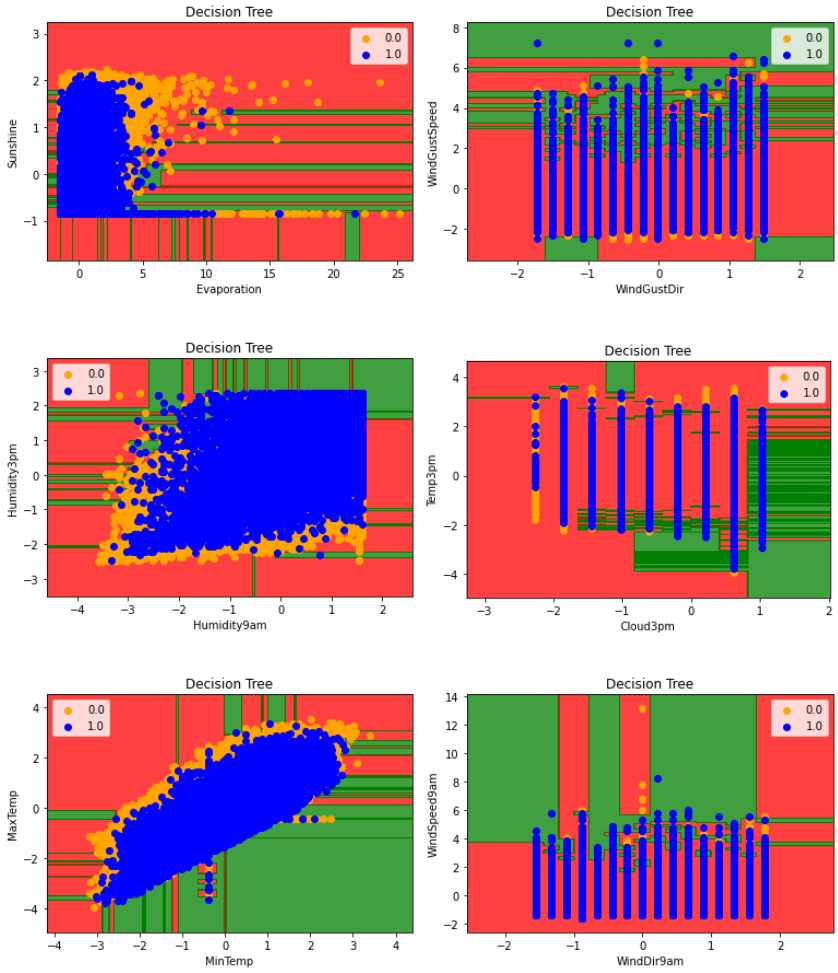
分析結果：貝氏分類器有些分析結果趨近於邏輯回歸，但不能說分類效果沒有邏輯回歸的效果好，數學式的不同會導致分類情況不相同，會導致跟邏輯回歸差不多的原因，可以解釋成，缺失資料多導致計算事前及事後機率時無法導向正確的區域。

模型建立：使用 sklearn 來幫助模型建立。

參數調整：測試集設定為 0.2

- Decision Tree (決策樹):

$$Entropy = -\sum_{j} p_j \log_2 p_j$$



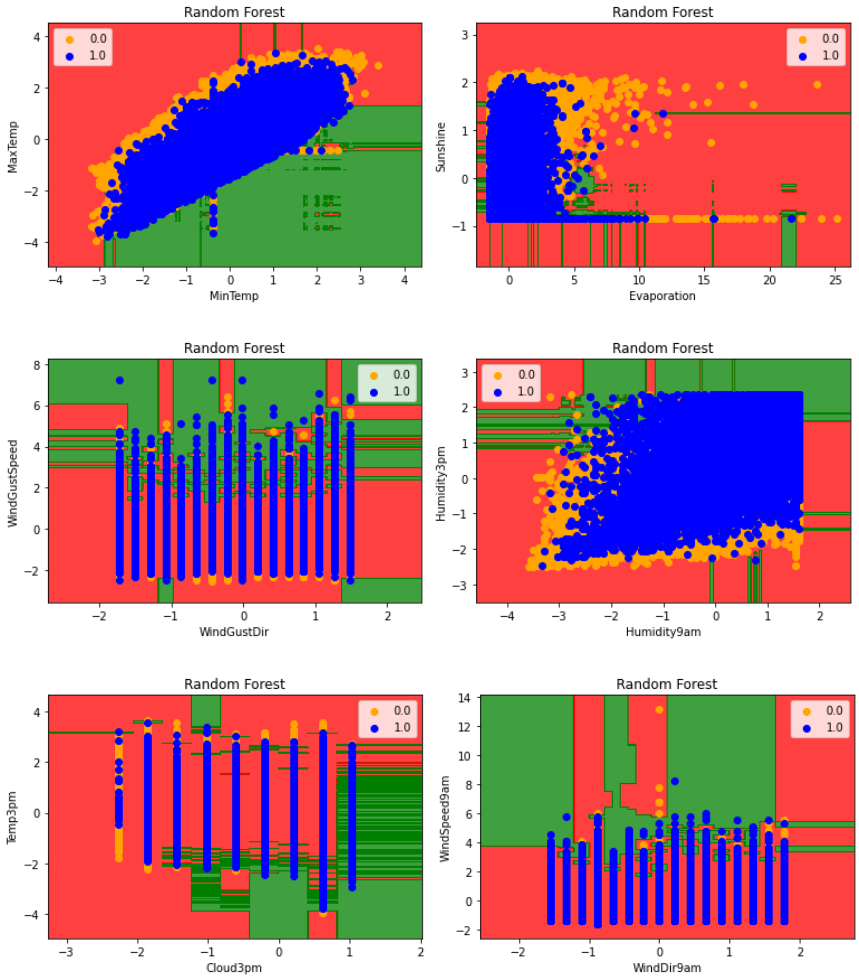
分析結果：決策樹是目前分類效果最好的模型，可以畫出某些較精細的區塊，並透過圖片較能準確看出哪個狀況下是否降雨，雖然還是有不正確的地方，但決策樹的分類效果是較好的

模型建立：使用 sklearn 來幫助模型建立。

參數調整：測試集設定為 0.2

- Random Forest (隨機森林)

$$\left(\hat{\theta}(x),\, \hat{\nu}(x)\right) \in \operatorname{argmin}_{\theta,\nu} \left\{ \left\| \sum_{i=1}^{n} \alpha_i(x)\, \psi_{\theta,\nu}\left(O_i\right) \right\|_2 \right\}$$



分析結果：與決策樹的效果差不多。

模型建立：使用 sklearn 來幫助模型建立。

參數調整：測試集設定為 0.2

最佳分析結果：Decision Tree

Random Forest