資科系 110916015 林翡

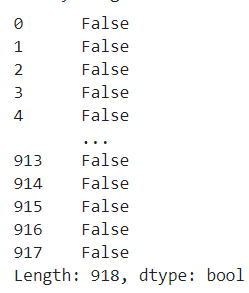
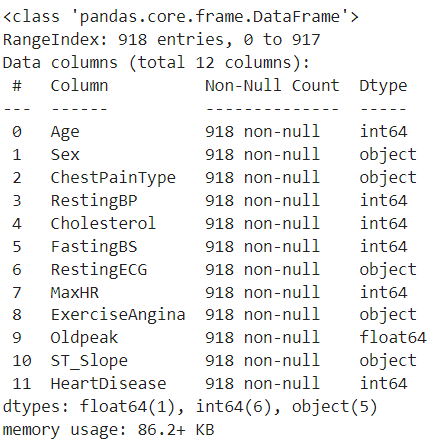
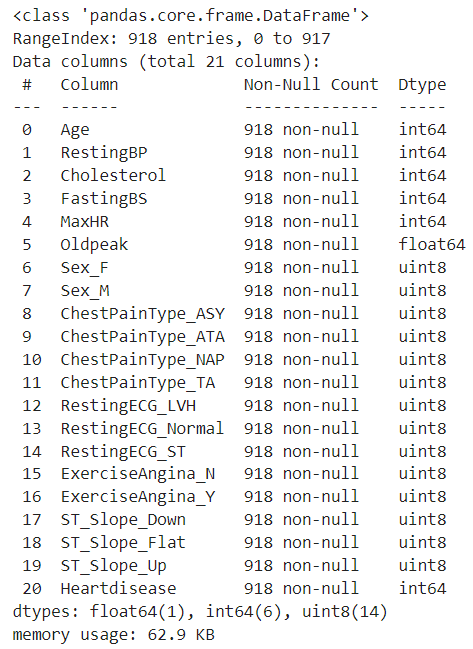
機器學習HW2\_Classification

程式碼語言:Python (在Jupyter Lab上寫)

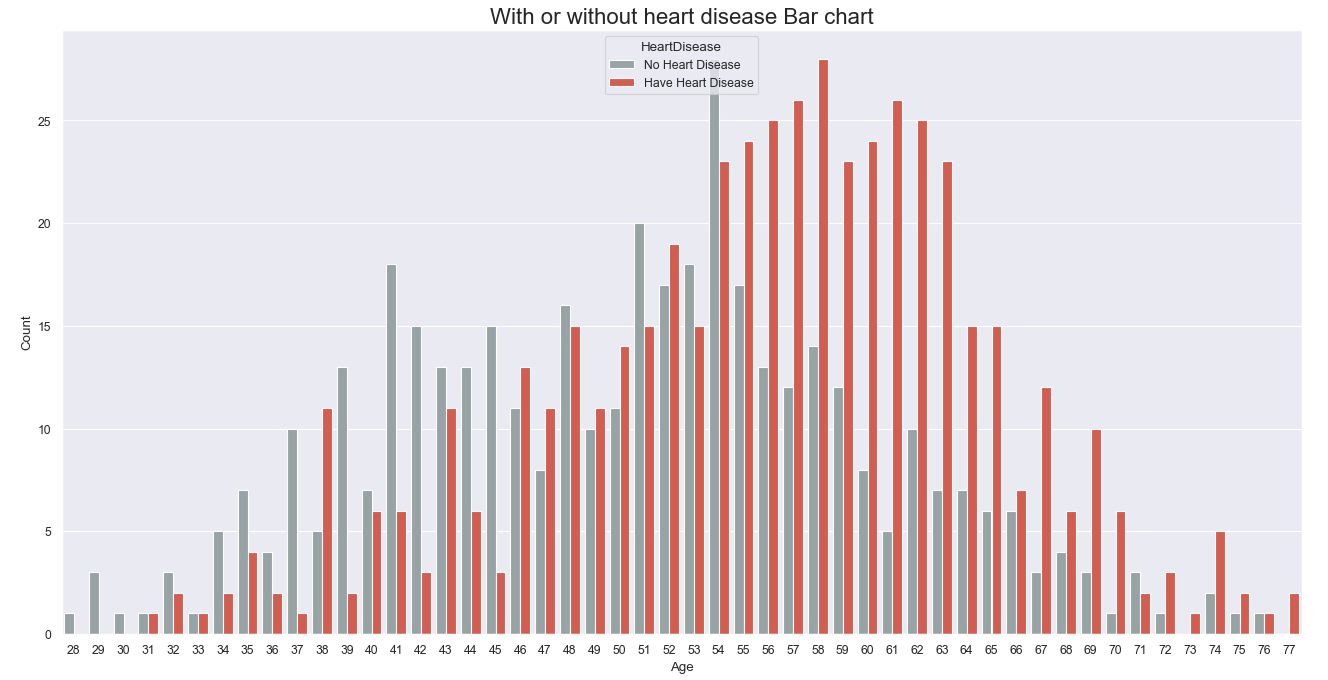
內容:

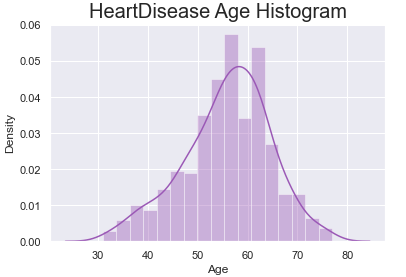
1. 資料清理與視覺化圖表 (程式碼[1]~程式碼[14])

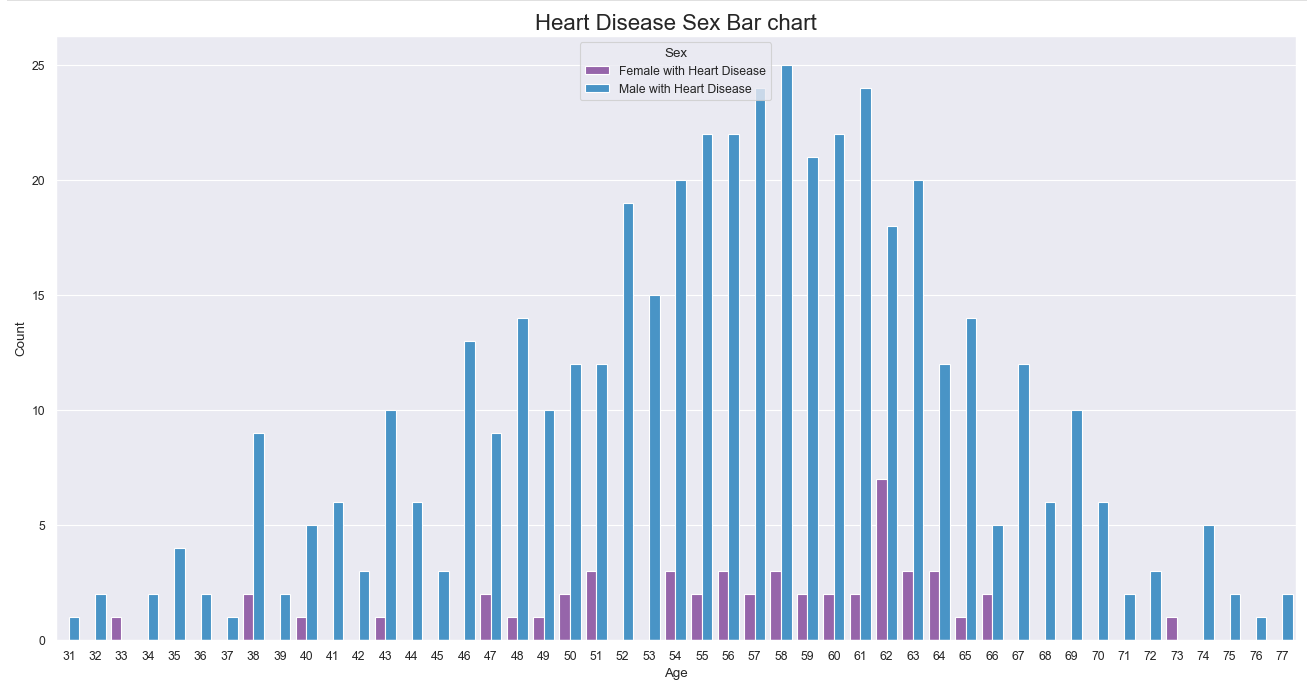
* 資料清理

1. 檢查資料集有無遺漏值及重複值，由圖中可看出此資料集都沒有遺漏值及重複值。(程式碼[1])
2. 把型態是object的欄位(Sex、ChestPainType、RestingECG、ExerciseAngina、ST\_Slope)轉換成數值表示，再把這些欄位還原它的真實含意，並擴展特徵；最後把HeartDisease移到後面改名為Heartdisease。(程式碼[7]~程式碼[11]、程式碼[14]~程式碼[15]))

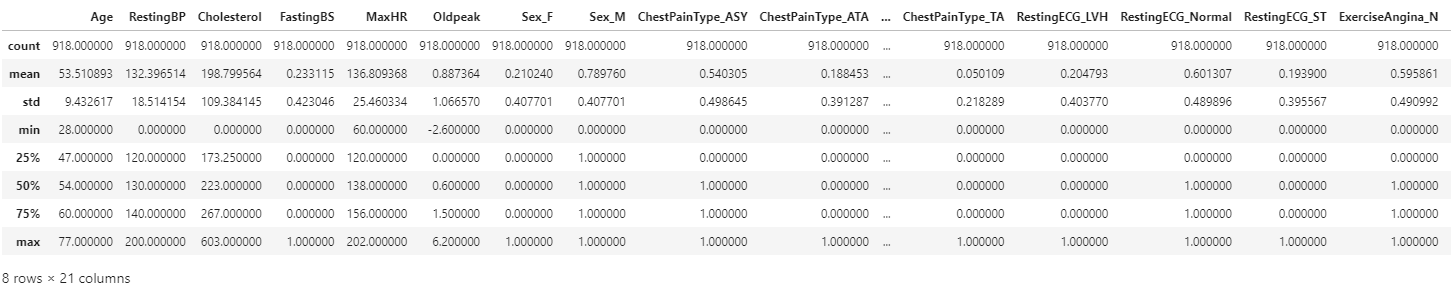
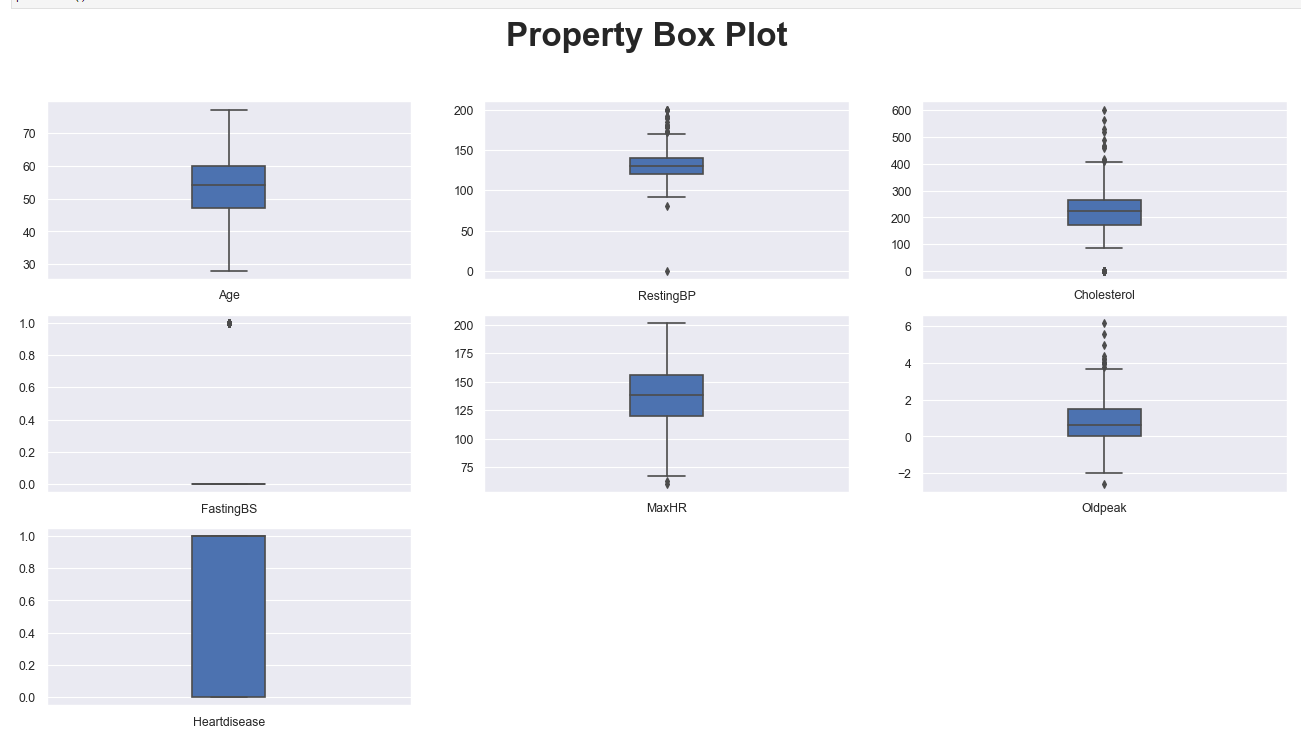
* 視覺畫圖表 (程式碼[3]~程式碼[6])

 用Age分布來看有無心臟疾病的人數，可以發現在年齡54~63是有心臟疾病人數較多的。

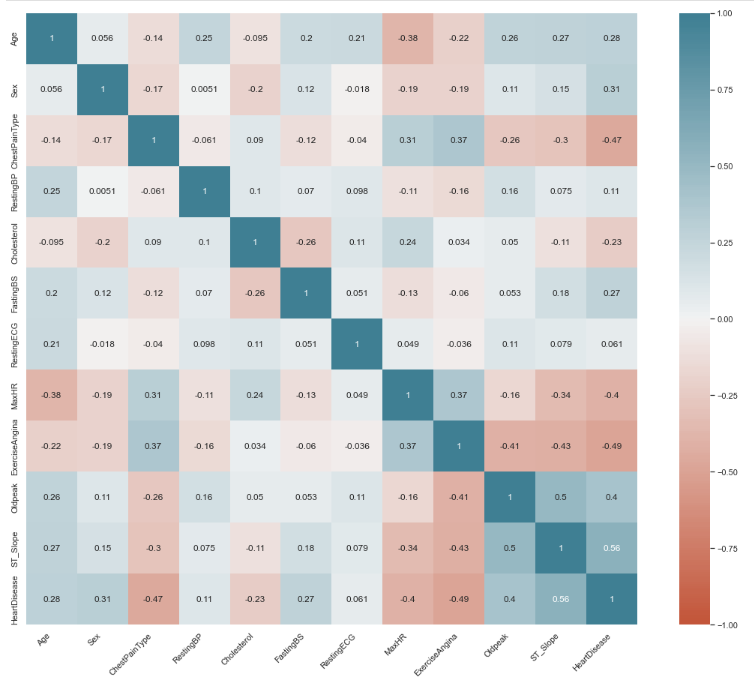
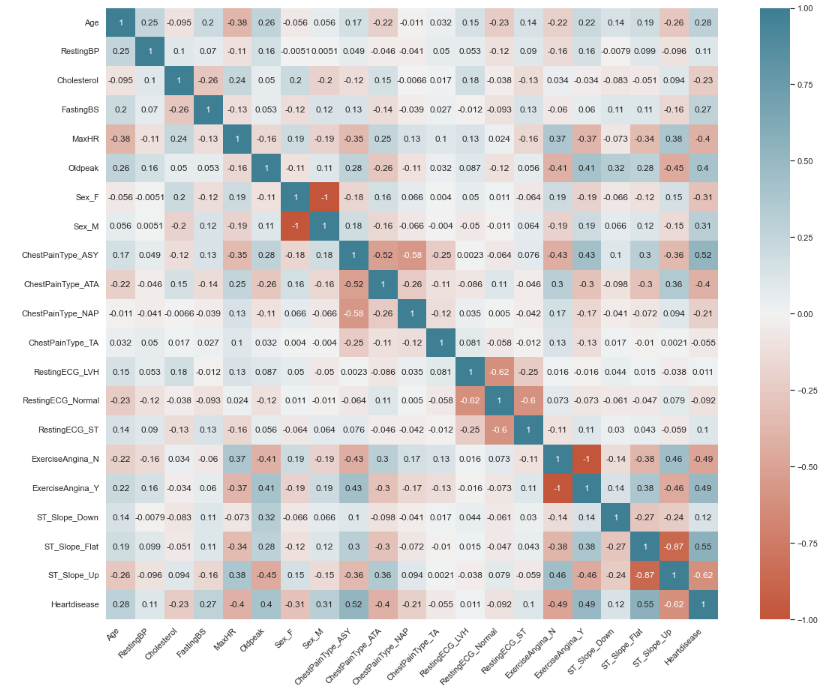
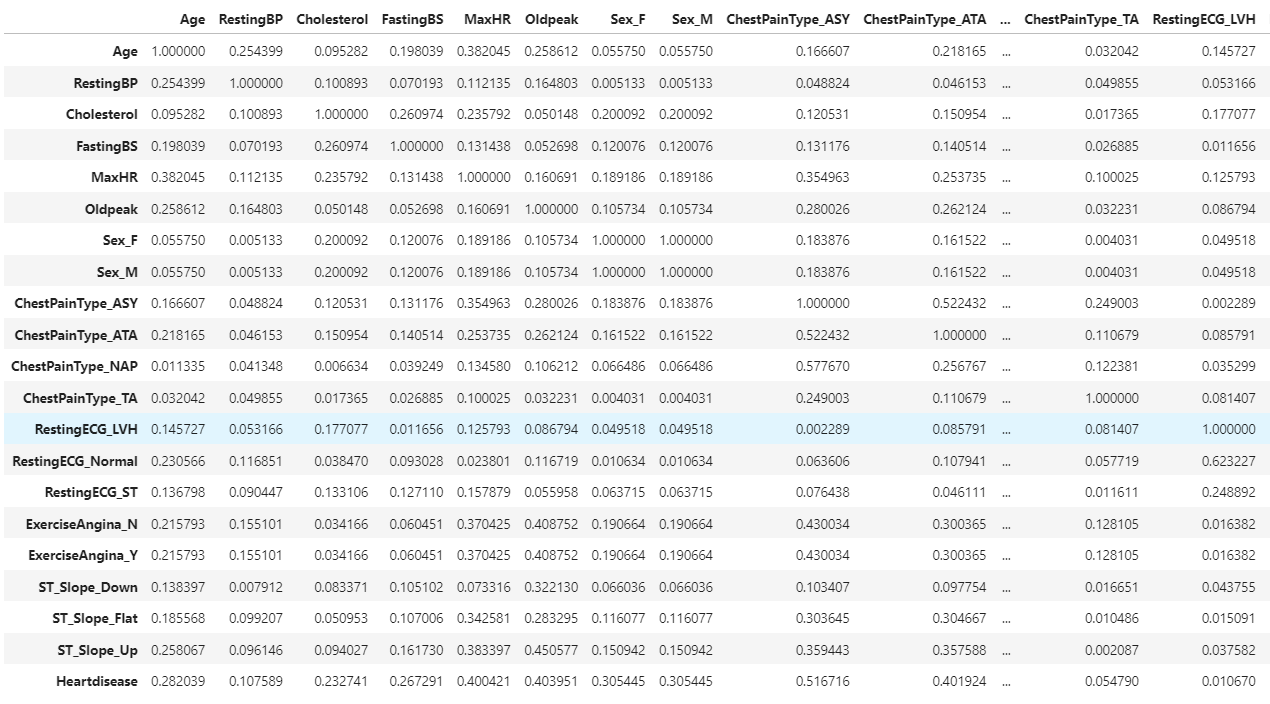
有心臟疾病的年齡直方圖，可看出在50~60歲的區間有最高峰。

也是用年齡來看有心臟疾病的男女人數，可看出男性有心臟疾病的人數大於女性。

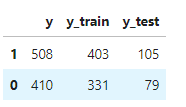
1. 敘述性統計分析 (程式碼[17]~程式碼[18])

* 敘述性統計分析表 (程式碼[17]):計算各個欄位的平均值、標準差、最小值、四分位數、中位數、最大值。
* 敘述性統計分析圖(程式碼[18]):選型態是int和float的。

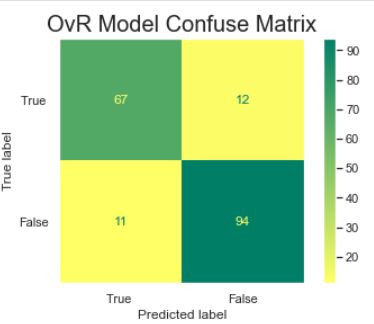
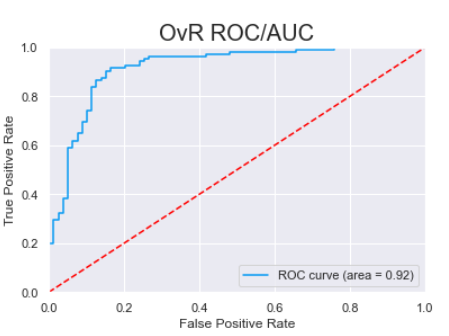
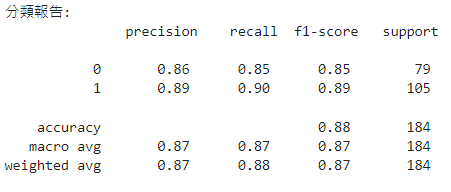
1. 特徵相關性分析 (程式碼[13]、程式碼[19]~程式碼[20])

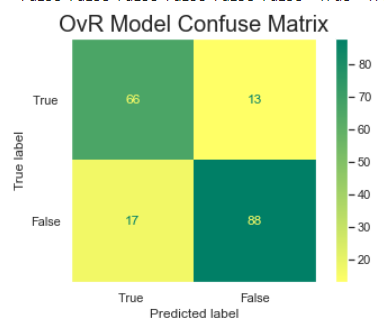
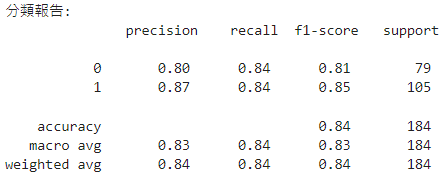
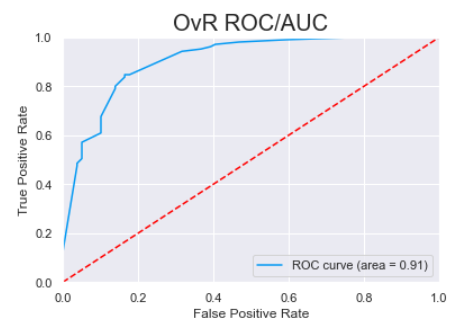
* 把型態是object的轉換成數值後的相關係數熱度圖 (程式碼[13])
* 資料清理後所有欄位的相關係數熱度圖 (程式碼[19]):
* 分析:罹患心臟病與ChestPainType、ExerciseAngina、ST\_Slope有較強的相關性，相關係數都超過0.4:而與RestingECG之間的相關性較弱。
* 皮爾森相關係數 (程式碼[20]):計算各個欄位之間是否有線性相關。

1. 資料分割與建置4個分類模型(1. Logistic regression、2. SVM、3. Random forest、4. KNN)

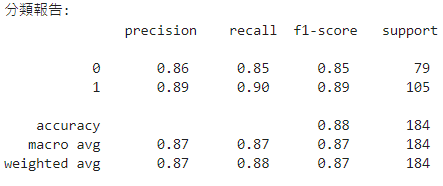
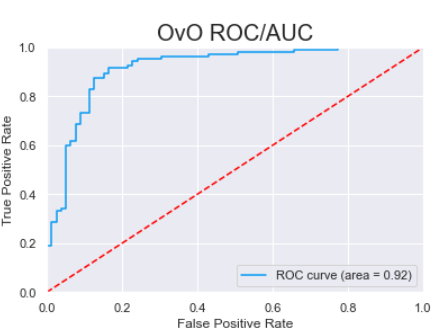
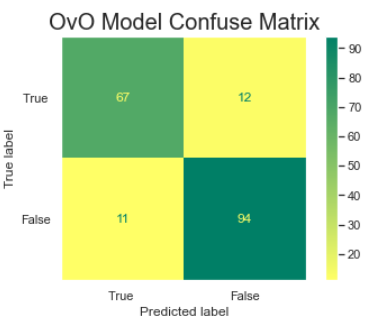
* 資料分割 (程式碼[23]):將原始資料的80%分為訓練集(X\_train、y\_train)，20%分為測試集(X\_test、y\_test)，特徵選擇除Heartdisease以外的全部特徵，目標類別Heartdisease，最後印出資料集、訓練和測試集的類別比例。
* 建置4個分類模型(有外加一個決策樹模型)

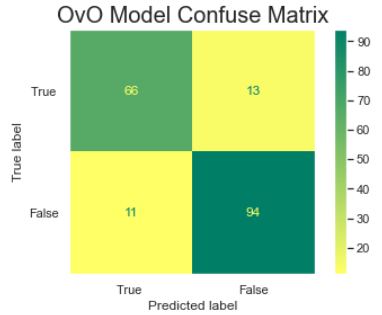
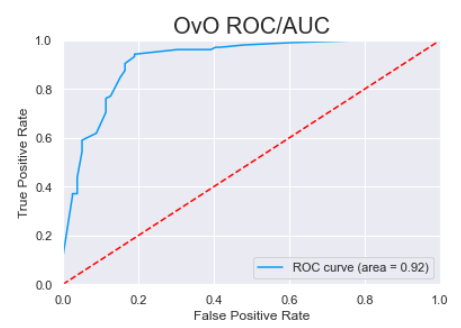
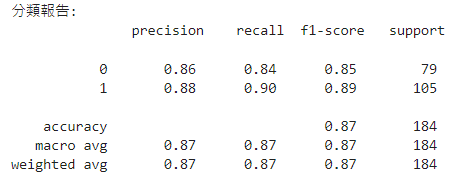
1. OvR- Logistic regression (程式碼[26]~程式碼[30])

混淆矩陣的評估結果 ROC/AUC曲線 分類報告

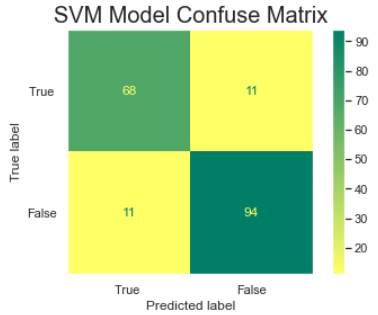
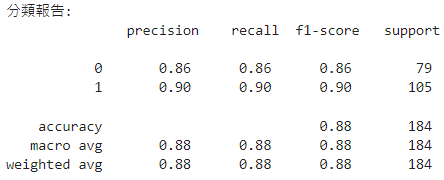
 篩選特徵後的混淆矩陣、ROC/AUC曲線、分類報告

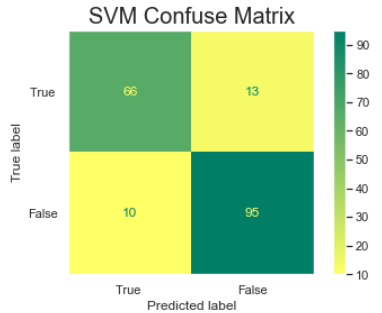
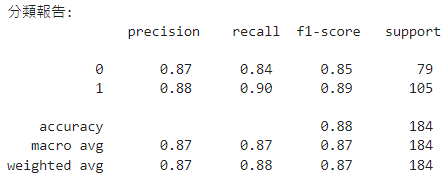
1. OvO- Logistic regression (程式碼[31]~程式碼[35])

 混淆矩陣的評估結果 ROC/AUC曲線 分類報告

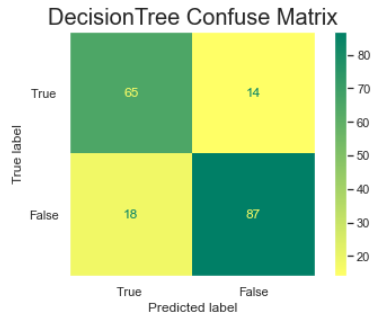
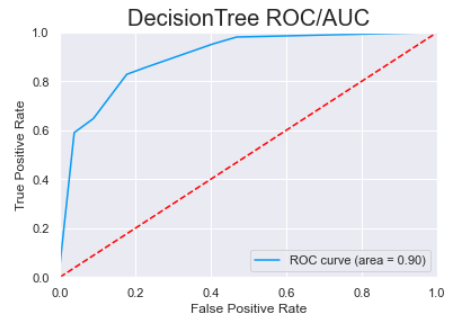
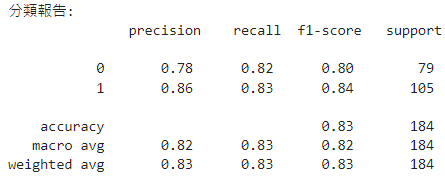
篩選特徵後的混淆矩陣、ROC/AUC曲線、分類報告

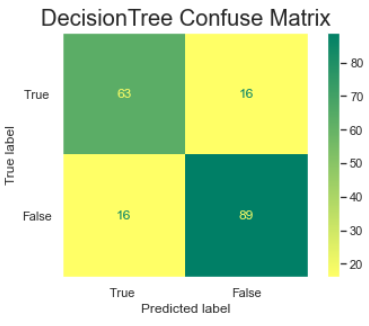
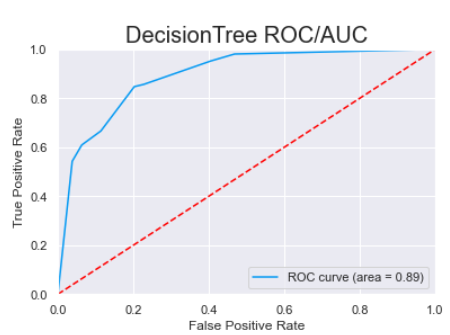
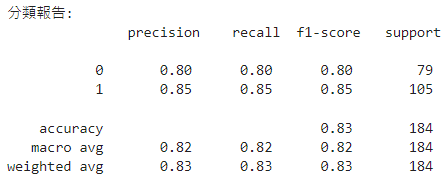
1. SVM (程式碼[36]~程式碼[40])):最大回合數設定為1500以增加收斂機會，因為LinearSVC沒有predict\_proba屬性所以就沒有畫ROC/AUC曲線了。

 混淆矩陣的評估結果 分類報告

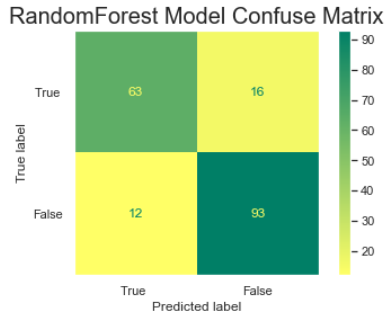
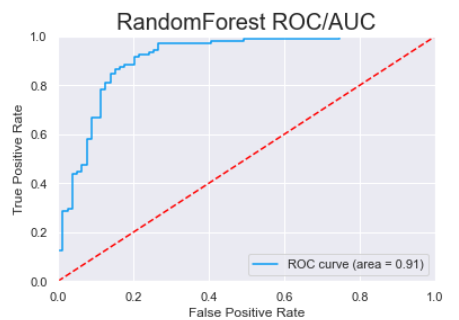
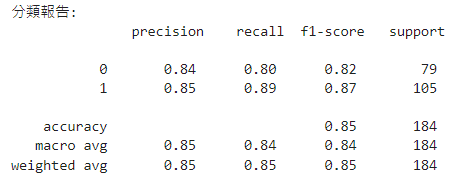
 篩選特徵後的混淆矩陣、分類報告

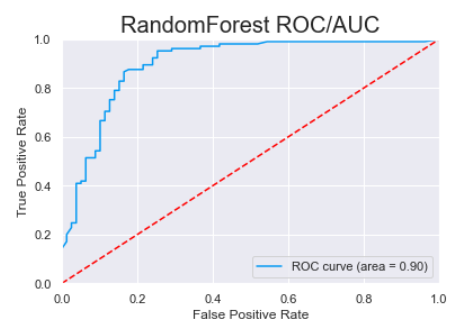
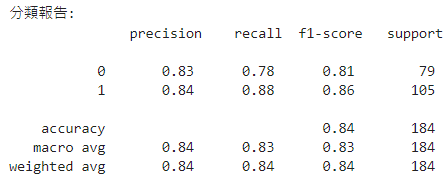
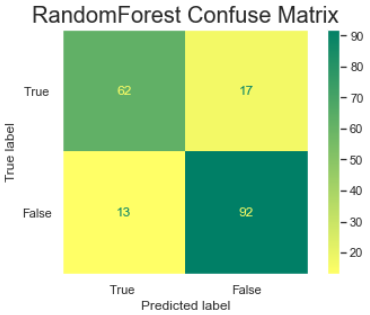
1. Decision Tree (程式碼[41]~程式碼[45]):決策樹的最大高度設為3以防止過度擬合的問題。

 混淆矩陣的評估結果 ROC/AUC曲線 分類報告

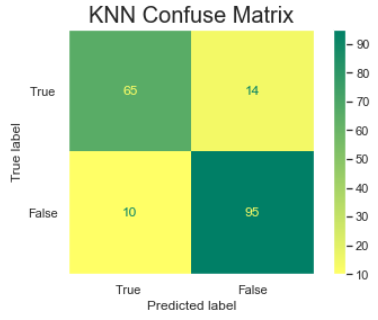
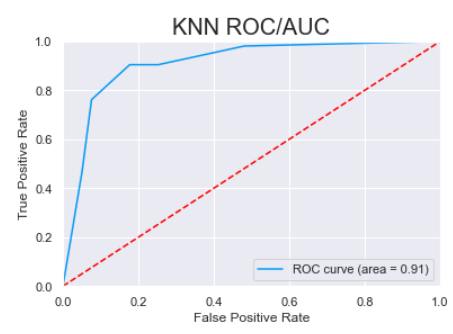
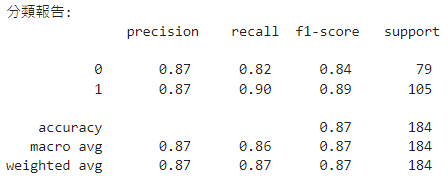
篩選特徵後的混淆矩陣、ROC/AUC曲線、分類報告

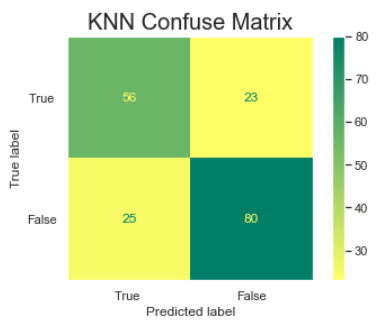
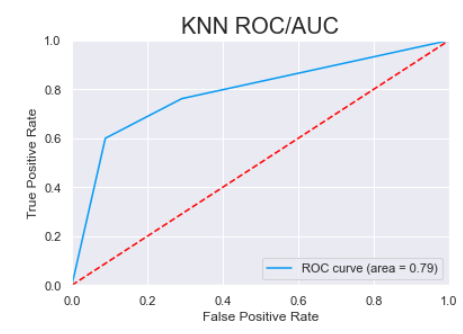
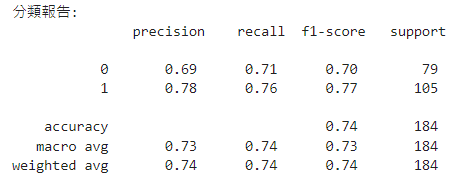
1. Random Forest (程式碼[46]~程式碼[50]):樹的最大高度設為3，因為隨機森林可能會增加計算成本所以加上n\_jobs=-1設定使用多核心進行平行運算。

 混淆矩陣的評估結果 ROC/AUC曲線 分類報告

 篩選特徵後的混淆矩陣、ROC/AUC曲線、分類報告

1. K-NN (程式碼[51]~程式碼[55]):同樣用n\_jobs=-1增加運算效能；因為KNeighborsClassifier沒有coef\_和feature\_importances\_屬性，所以改用SelectKBest類，以f\_classif為評估指標對特徵進行篩選。

 混淆矩陣的評估結果 ROC/AUC 分類報告

篩選特徵後的混淆矩陣、ROC/AUC曲線、分類報告

1. 綜合比較4個模型的分類結果與分析討論:評估模型效能我多用了交叉驗證的結果，純粹看課本上有做做看而已。紅色數值是評估結果最高分的，橘色數值是交叉驗證結果最高分的，而紫色框是經特徵篩選後有變高的值。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 分類模型 | | OvR- LR | OvO- LR | SVM | | Decision Tree | Random Forest | K-NN |
| 交叉驗證結果 | Accurary | 87.48% | 88.02% | 85.86% | | 77.18% | 84.76% | 86.95% |
| Precision | 86.98% | 87.71% | 84.01% | | 78.97% | 85.0% | 88.17% |
| Recall | 92.38% | 92.38% | 93.33% | | 86.67% | 93.33% | 89.52% |
| F1-score | 0.89 | 0.90 | 0.88 | | 0.81 | 0.88 | 0.89 |
| 評估結果 | Accurary | 87.5% | 87.5% | 88.04% | | 82.61% | 86.96% | 86.96% |
| Precision | 88.68% | 88.68% | 89.52% | | 86.14% | 87.156% | 87.16% |
| Recall | 89.52% | 89.52% | 89.52% | | 82.86% | 90.48% | 90.48% |
| F1-score | 0.89 | 0.89 | 0.90 | | 0.85 | 0.89 | 0.89 |
| AUC | 0.92 | 0.92 |  | | 0.90 | 0.91 | 0.91 |
| 篩選特徵後 | | | | | | | | |
| 交叉驗證結果 | Accurary | 80.44% | 85.87% | | 85.87% | 76.08% | 85.33% | 76.02% |
| Precision | 81.36% | 85.13% | | 85.13% | 80.95% | 81.38% | 73.47% |
| Recall | 86.67% | 91.43% | | 91.43% | 77.14% | 94.29% | 91.43% |
| F1-score | 0.83 | 0.88 | | 0.88 | 0.79 | 0.88 | 0.81 |
| 評估結果 | Accurary | 83.7% | 86.96% | | 87.5% | 82.61% | 83.7% | 73.91% |
| Precision | 87.13% | 87.85% | | 87.96% | 84.76% | 82.61% | 77.67% |
| Recall | 83.81% | 89.52% | | 90.48% | 84.76% | 90.48% | 76.19% |
| F1-score | 0.85 | 0.89 | | 0.89 | 0.85 | 0.86 | 0.77 |
| AUC | 0.91 | 0.92 | |  | 0.89 | 0.89 | 0.79 |

* 模型效能分析:由以上表格可以發現OvO-LR在交叉驗證結果是表現最好的，Recall則是SVM和Random Forest表現最好；而在一般的評估結果中是SVM表現最佳，Recall是Random Forest和KNN表現最佳，但其實每個模型的Accurary都沒有差很多，除了Decision Tree是六個模型中表現明顯最差的。在篩選特徵後，發現效能沒有比篩選前好，可能是受到有些相關性低的特徵影響，然後可以發現KNN在篩選特徵後模型效能明顯低於先前的，表示它可能都選到無關緊要特徵。以綜觀來看， SVM是整體表現最好的。