Intelligent Integration of Enterprise

Project 3

以抽血檢查特徵判定肝臟疾病

109034536 郭芸如

指導教授：邱銘傳 教授

目錄

摘要

醫師確立病患肝臟纖維化程度的檢查需使用肝臟切片方式(侵入檢查)，且嚴重肝硬化病患會有血小板低下問題，侵入式檢查風險高。本研究使用類神經網路與機器學習方法(如:決策樹、隨機森林、XGboost)，僅根據抽血特徵(例: 膽紅素、丙胺酸轉胺酶…)就判斷出病患是否有罹患肝臟疾病。模型使用 Gridsearch 進行超參數調整，模型驗證方面使用 Stratifiedkfold 法，模型準確率高達 84%。此疾病判斷模型可輔助醫生診察，提供決定是否有需要做侵入性檢查之依據。

一、研究動機與背景

1.研究動機

根據衛生福利部 108 年統計，慢性肝病及肝硬化位居國人死因第 10，108 年死亡人數達 4,240 人，及早篩檢出肝臟病症並進行治療變得愈發重要。常見的肝病的初步篩檢包含血液檢查、腹部超音波檢查，當病患初步篩檢結果有異常時，醫師為確立病癥、嚴重程度需使用肝臟切片方式(侵入檢查)，嚴重肝硬化病患會有血小板低下問題，侵入式檢查風險高。本研究旨在使用機器學習方法，僅根據血液檢查特徵(例: 膽紅素、丙胺酸轉胺酶…)就判斷出病患是否有罹患肝臟疾病，協助醫生、病患決定是否有需要做侵入性檢查與進一步治療。

2. 5W1H

- What(解什麼問題?)

解決醫生僅由血液檢查、腹部超音波檢查等特徵無法精準判斷是否患肝臟疾病的問題。

- Why?:(為什麼要進行改善此問題?)

切片檢查具腹腔、胸腔出血以及氣胸、膽道穿刺、抽搐等風險，應盡量避免對沒患肝病或為切片檢查高風險族群的病患做肝臟切片檢查。

- When?: (什麼時候辨別病徵?)

當病人至醫院做血液檢查(抽血)時

- Who?: (由誰來使用此判斷模型?)

肝臟病科醫生。

- Where?: (在何處可以進行問題改善?)

醫院的肝臟病科。

- How?: (如何解決此問題?)

使用機器學習方法、神經網路模型等，協助醫生對病患肝臟病癥有更精準的判斷。

3.研究流程

本研究流程如下

圖一、研究流程

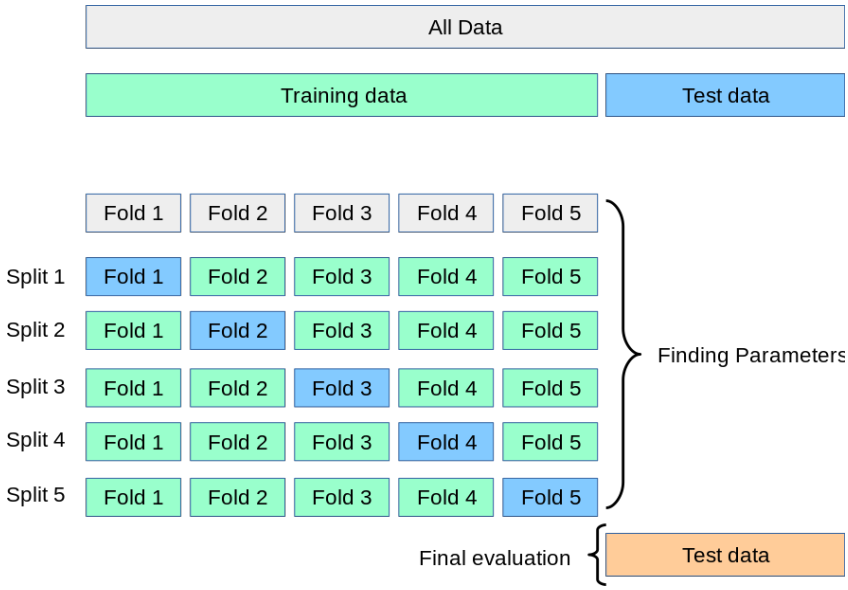
二、文獻回顧

1.決策樹、隨機森林、XGboost

決策樹透過在每一步驟中選擇一個適當的特徵分類，以達到資料分類的效果。隨機森林的原理是結合多個決策樹，預測結果是將各個決策樹的預測結果取眾數產生。其做法是從資料集中隨機抽取部分特徵、資料建立決策樹，即每個決策樹由不同的特徵與資料子集所建構。如此透過結合多個「弱學習器」來建構一個更強穩模型的方法稱為 Ensemble Method(或 Ensemble learning)，應用 Ensemble Method 模型較不會有偏差或是發生過擬合，有較高的分類準確率。XGboost(Extreme Gradient Boosting)與隨機森林相似，但透過序列的方式生成樹，後面生成的樹會與前一棵樹相關。

2.Kfold 交叉驗證

Kfold 交叉驗證將資料切分為 k 份(下圖 k=5)，每次選取其中一份作為驗證集進行模型訓練與驗證。每次驗證集與訓練集資料不同，重複試驗並計算平均測試績效與變異以評判模型好壞。如此驗證相較於只切分一次資料就進行訓練與驗證較不會偏頗。其中，本研究使用的是 Stratifiedkfold，Stratifiedkfold 再進行資料切分時會保證驗證集資料不同類別比例與原資料集相同。舉例: 資料集中有病沒病的比例為 5:3，則以 Stratifiedkfold 法切分之驗證集有病沒病比例亦會是 5:3。



圖二、Kfold

(圖源: Cross-validation: evaluating estimator performance)

三、資料描述與前處理

1.資料描述

本研究使用 UCI Machine Learning Repository 的 ILPD (Indian Liver Patient Dataset) Data Set 資料集，搜集 583 位印度安得拉邦 (Andhra Pradesh)東北部病患資訊。其中，患肝病者 416 人、無患病者 167 人，男性 441 人、女性 142 人。資料包含病患之血液檢查、患病資訊，各欄位詳細內容說明如下表。

	A	B	C	D	E	F	G	H	I	J	K
1	Age	Gender	Total_Bilirubin	Direct_Bilirubin	Alkaline Phosphotase	Alamine Aminotransferase	Aspartate Aminotransferase	Total Protiens	Albumin	Albumin and Globulin Ratio	Dataset
2	65	Female	0.7	0.1	187	16	18	6.8	3.3	0.9	1
3	62	Male	10.9	5.5	699	64	100	7.5	3.2	0.74	1
4	62	Male	7.3	4.1	490	60	68	7	3.3	0.89	1
5	58	Male	1	0.4	182	14	20	6.8	3.4	1	1
6	72	Male	3.9	2	195	27	59	7.3	2.4	0.4	1
7	46	Male	1.8	0.7	208	19	14	7.6	4.4	1.3	1

圖三、原始資料

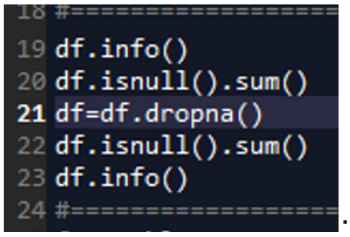
表一、資料集欄位資訊

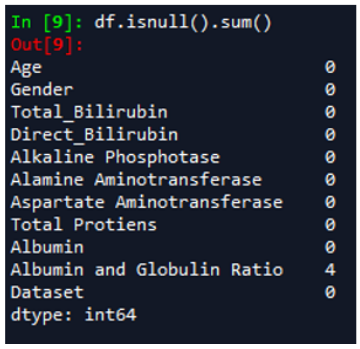
欄位	說明	統計資訊
Age	年齡(阿拉伯數字) 單位: 歲	Count: 583 Mean: 44.746141 Std: 16.189833 Min: 4；Max: 90
Gender	性別 Male, Female,	Count: 583 人 Male: 441 人；Female: 167 人
Total_Bilirubin	總膽紅素	Count: 583 Mean: 3.298799 Std: 6.209522 Min: 0.4；Max:75
Direct_Bilirubin	間接膽紅素	Count: 583 Mean: 1.486106 Std: 2.808498 Min: 0.1；Max: 19.7
Alkaline Phosphotase (ALP)	鹼性磷酸酶	Count: 583 Mean: 290.576329 Std: 242.937989 Min: 63；Max: 2110
Alamine Aminotransferase (ALT)	丙胺酸轉胺酶 (GPT)	Count: 583 Mean: 80.713551 Std: 182.620356 Min: 10； Max: 2000
Aspartate Aminotransferase (AST)	天門冬胺酸轉胺酶 (GOT)	Count: 583 Mean: 109.910806 Std: 288.918529 Min: 10；Max: 4929
Total Protiens (原資料集筆誤，應為”Proteins”)	總蛋白質	Count: 583 Mean: 6.48319 Std: 1.085451 Min: 2.7；Max: 9.6
Albumin	白蛋白	Count: 583 Mean: 3.141852 Std: 0.795519 Min: 0.9；Max: 5.5
Albumin and Globulin Ratio	白/球 蛋白比值	Count: 579 Mean: 0.947064 Std: 0.319592 Min:0.3；Max:2.8
Dataset	1:有肝病，2:沒肝病	Count:583 有肝病 416；沒肝病 167 人

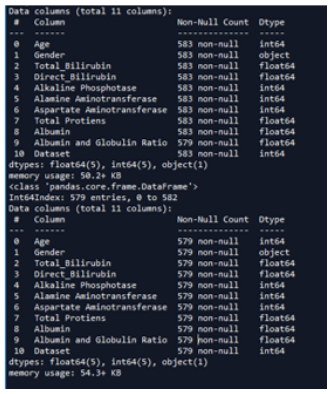
2.資料前處理

- 檢查並刪除缺失值

使用 pandas 套件中.info()、.isnull().sum()觀察每欄資料型態與數量，發現 Albumin and Globulin Ratio 有 4 個缺失值，因缺失值數量不多，直接使用.dropna()將含有缺失值的 4 位病患資料刪除，最終使用 579 位病患資料。







圖四、檢查並刪除缺失值

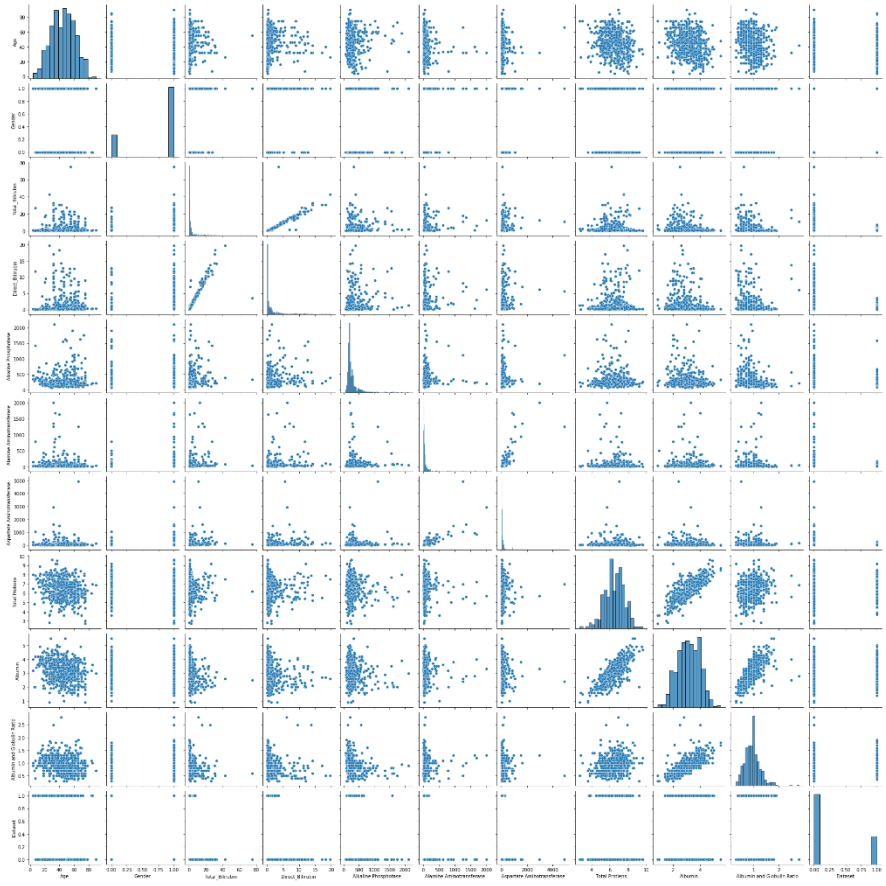
- 對類別欄位進行 encode

針對類別欄位“Gender”、”Dataset”使用 sklearn 套件 LabelEncoder()功能進行資料轉換，” Gender”中 Female 和 Male 分別改標籤為 0、1，” Dataset”中 1:有肝病和 2:沒肝病分別改標籤為 0、1。

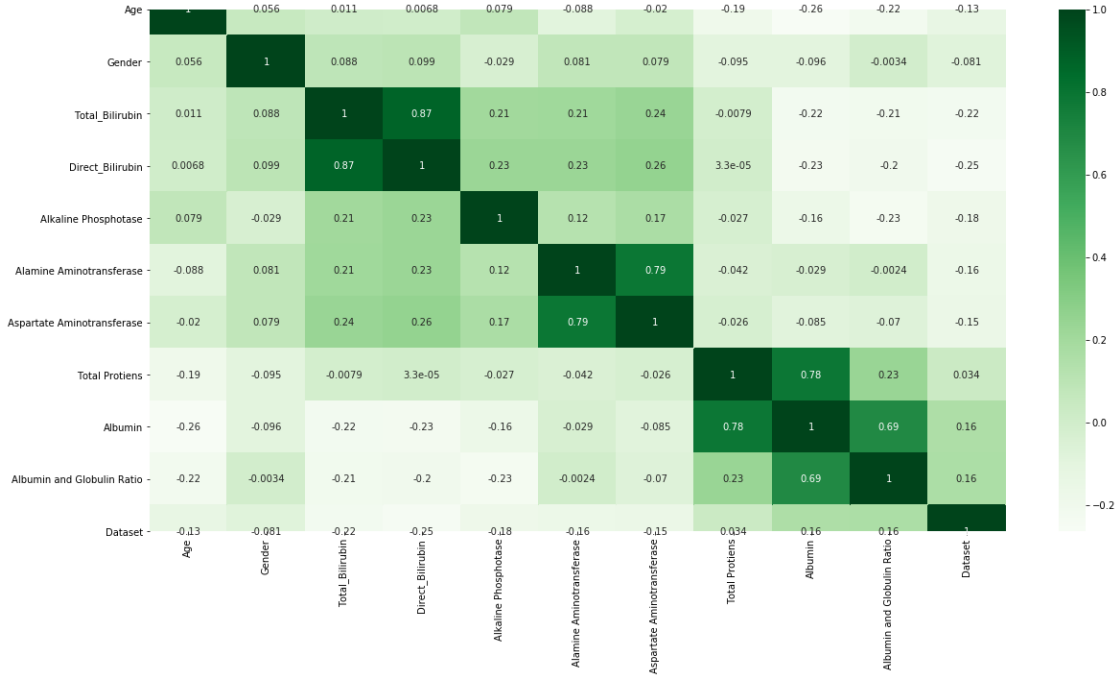
- 觀察資料分布，資料擴增(平衡有肝病、沒肝病資料筆數)

使用 seaborn 套件繪製多個圖表如下。從熱力圖可觀察到所有特徵項中，”Gender”、” Total Proteins”與”Dataset(有無肝病)”的相關性較其他特徵小許多，分別是-0.081、0.034。後續模型將分兩部分 1.使用全部特徵 2.使用部分特徵(刪除”Gender”、” Total Proteins”特徵)。

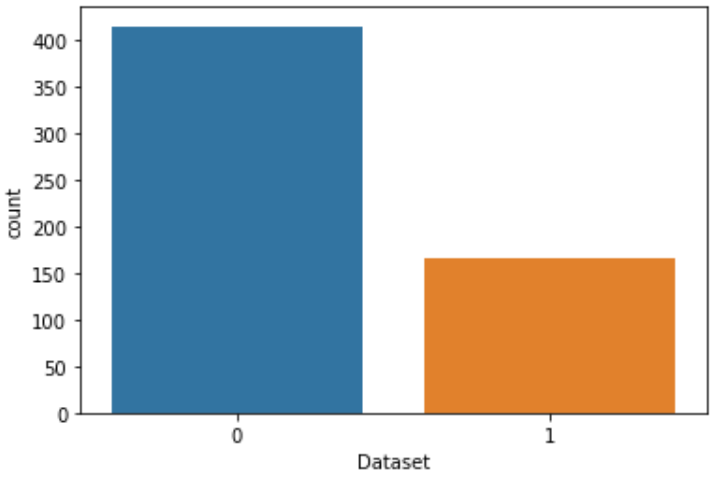
此外，從”Dataset(有無肝病)”分布圖可看出有病與沒病的資料比例懸殊(刪除缺失值後有肝病人數為 414 人，無肝病 165 人)，為使後續建構模型時能有效學習，使用 sklearn 套件 resample()功能擴增資料，使有肝病、沒肝病資料筆數相等(有病沒病均為 414 筆，總共 828 筆)。



圖五、資料集 pairplot



圖六、資料集相關性熱力圖



圖七、資料擴增前”Dataset(有無肝病)”分布圖(0 有肝病、1 無肝病)

• 資料標準化(針對使用 ANN、SVM 之資料)

使用 Sklearn 套件的 StandardScaler()將”Total_Bilirubin”、”Albumin and Globulin Ratio”…等特徵之資料值轉換為平均值會為 0，標準差為 1。

四、模型架構

本節將先針對 6 種常見分類模型進行初步測試，並挑選測試集準確率較佳之模型進行進一步模型設計與參數優化。

1. 使用預測模型

• 初步測試

初步測試(未使用交叉驗證)決策樹、SVM、ANN 等以下 6 種分類模型，ANN 自行架構，其餘模型僅針對分類必要參數進行設定，其他參數使用 default 值。發現決策數、隨機森林、XGBoost、ANN 之測試集準確率皆大於 80%，下一階段將根據篩選出的 4 種模型進行超參數調整。下表呈現個模型測試準確率，以及使用所有資料進行交叉驗證之結果(mean test accuracy、std)

表二、初步模型測試結果

分類器	test accuracy	mean test accuracy	std
LogisticRegression	73%	69%	0.08
DecisionTree	82%	81%	0.05
RandomForest	86%	83%	0.07
XGBoost	81%	76%	0.07
KNN	71%	71%	0.06
SVM	73%	66%	0.05
ANN	81%	35%	0.31

2. 模型分析與改善

2.1 ANN 模型深度

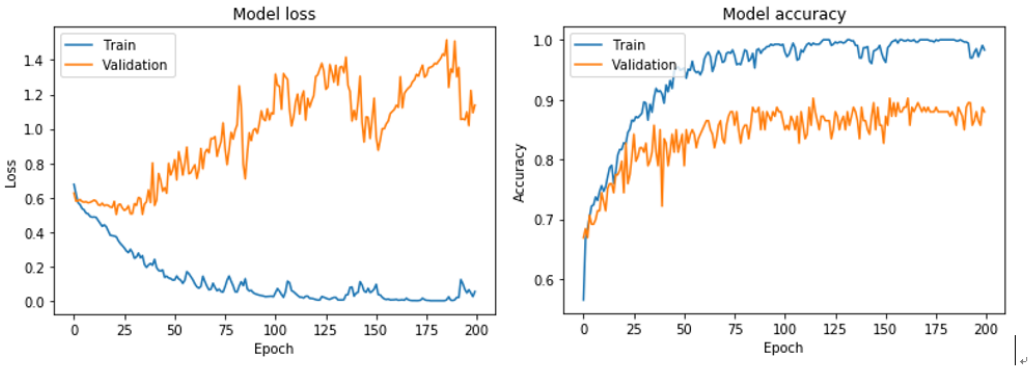
依據初步測試之 ANN 模型架構增減層數，設計出深度由淺至深的 3 種模型如下表，並進一步測試並比較 3 種模型績效。其中，模型 2 為初步測試中使用之 ANN 架構。比較後模型 2、模型 3 之 test accuracy 幾乎相同且模型 1 績

效最差。考量運算負荷，後續準確率達 80%但使用參數較少的模型 2 架構進行進一步參數優化。

表三、ANN 比較

模型編號	模型 1	**模型 2**	模型 3
輸入層	Dense(32, input_dim = 10, activation = 'relu'))		
隱藏層 (結使用 relu 激活函數)	Dense(64) Dropout(0.4)) Dense(32)	Dense(64) Dense(128) Dense(64) Dropout(0.4)) Dense(32)	Dense(64) Dense(128) Dense(64) Dropout(0.4)) Dense(32) Dense(64) Dense(128) Dense(64) Dropout(0.4)) Dense(32)
輸出層	Dense(1, activation = 'sigmoid'))		
test accuracy	73%	80%	80%

在進行 ANN 訓練時，由下圖 loss 趨勢發現模型過擬合問題，曾嘗試使用 early stopping 技巧提前終止訓練，但發現模型在約 6~10 epoch 就會終止訓練，且測試準確率極差只有約 70%。後續參數優化將討論如何設定最佳 epoch 數。



圖八、ANN 訓練集、驗證集 loss、accuracy

2.2 應用 Ensemble learning(集成學習)

Ensemble learning 概念如前所述，是結合多個「弱學習器」來建構一個更強穩模型。本研究在進行各個模型參數優化後，透過 sklearn 的 VotingClassifier 組合決策樹、隨機森林、XGBoost 模型以下稱 Ensemble，透過 3 個模型判斷結果以多數決產出 Ensemble 模型的判斷結果。參數優化方法與績效比較請見下兩節。

```
clf1 = XGBClassifier(max_depth=10, n_estimators=150,learning_rate= 0.1,objective='binary:logistic')
clf2 = RandomForestClassifier(n_estimators=10, criterion = 'entropy',max_features = 'sqrt')
clf3 = DecisionTreeClassifier(criterion='entropy', max_features='sqrt')

# 硬投票
eclf = VotingClassifier(estimators=[('xgb', clf1), ('rf', clf2), ('DTree', clf3)], voting='hard')
for clf, label in zip([clf1, clf2, clf3, eclf], ['XGBoosting', 'Random Forest', 'Decision Tree', 'Ensemble']):
    scores = cross_val_score(clf, X_train, y_train, cv=5, scoring='accuracy')
    print("Accuracy: %0.2f (+/- %0.2f) [%s]" % (scores.mean(), scores.std(), label))
```

圖九、使用 VotingClassifier 建構 Ensemble 模型

2.3 超參數調整

本節針對前述 5 種模型設計參數水準，並使用 sklearn 套件的 GridSearchCV 功能找出最佳參數水準組合。即窮舉出所有可能參數水準組合，並以 Stratifiedkfold 法驗證績效(5 個 fold)，選出個模型 mean_test_score(即平均驗證集準確率)最高之參數組合。各參數水準見下表。

```
#=================RF GridSearch=============================================
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
import numpy as np
from sklearn.model_selection import GridSearchCV

model_RF=RandomForestClassifier(n_estimators=10, criterion = 'gini',max_features =
# define the grid search parameters
n_estimators = [5, 10, 50, 100, 150, 200]
criterion = ['gini', 'entropy']
max_features = ['auto', 'sqrt', 'log2']
param_grid_RF = dict(n_estimators = n_estimators, criterion = criterion, max_featur

grid = GridSearchCV(estimator=model_RF, param_grid=param_grid_RF, n_jobs=-1, cv=5)
grid_result = grid.fit(X_train, y_train)
# summarize results
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print("%f (%f) with: %r" % (mean, stdev, param))
```

圖十、應用 GridSearchCV 進行超參數優化(以隨機林為例)

```
...: for mean, stdev, param in zip(means, stds, params):
...:     print("%f (%f) with: %r" % (mean, stdev, param))
Best: 0.832388 using {'criterion': 'entropy', 'max_features': 'auto', 'n_estimators': 10}
0.759843 (0.016394) with: {'criterion': 'gini', 'max_features': 'auto', 'n_estimators': 5}
0.826373 (0.026849) with: {'criterion': 'gini', 'max_features': 'auto', 'n_estimators': 10}
0.814240 (0.019029) with: {'criterion': 'gini', 'max_features': 'auto', 'n_estimators': 50}
0.818831 (0.031443) with: {'criterion': 'gini', 'max_features': 'auto', 'n_estimators': 100}
0.811267 (0.028048) with: {'criterion': 'gini', 'max_features': 'auto', 'n_estimators': 150}
0.817270 (0.021600) with: {'criterion': 'gini', 'max_features': 'auto', 'n_estimators': 200}
0.796024 (0.034095) with: {'criterion': 'gini', 'max_features': 'sqrt', 'n_estimators': 5}
0.821793 (0.031313) with: {'criterion': 'gini', 'max_features': 'sqrt', 'n_estimators': 10}
0.817327 (0.039299) with: {'criterion': 'gini', 'max_features': 'sqrt', 'n_estimators': 50}
0.820324 (0.023405) with: {'criterion': 'gini', 'max_features': 'sqrt', 'n_estimators': 100}
0.821827 (0.028792) with: {'criterion': 'gini', 'max_features': 'sqrt', 'n_estimators': 150}
```

圖、GridSearchCV 搜尋結果(僅擷取部分，以隨機林為例)

表四、模型參數水準

模型	參數	參數水準
DecisionTree	criterion	gini, entropy
	max_features	None, auto, sqrt, log2
RandomForest	n_estimators	5, 10, 50, 100, 150, 200

XGBoost	criterion	gini, entropy
	max_features	auto, sqrt, log2
	n_estimators	5, 10, 50, 100, 150, 200
	max_depth	3, 6, 10
	learning_rate	0.01, 0.1, 0.2
ANN	batch_size	16, 32, 64
	optimizer	Adadelta, Adam, Nadam
	epochs	10, 50, 100, 200

表五、GridSearchCV 求出以模型最佳參數水準

模型	最佳參數水準	mean_test_score (平均驗證集準確率)
DecisionTree	criterion: 'entropy' max_features: 'sqrt'	81%
RandomForest	criterion: 'entropy' max_features: 'sqrt' n_estimators: 10	84%
XGBoost	learning_rate: 0.1, max_depth: 10 n_estimators: 150	83%
ANN	batch_size :32 optimizer : adam epochs: 200	82%

3.模型績效比較

下表分別為使用所有抽血特徵，以及使用刪除”Gender”、” Total Proteins”特徵訓練之模型結果，模型參數皆使用上階段求得最佳參數水準。左方欄位為以訓練集訓練模型(內含 20%驗證集)並以測試集測試之結果(accuracy、precision、recall、f1-score)。右方欄為使用所有資料，以 Stratifiedkfold 驗證之結果(mean accuracy、std)。取所有特徵訓練之模型中，決策樹分類效果最為突出，其測試準確率高達 87%，交叉驗證之平均準確率雖與隨機森林、XGBoost、Esemble 相同，但標準差是最小的。相較於初步測試，使用最佳參數使決策數測試準確率提升 5%，以平均準確率相比則提升 1%。取部分特徵訓練之模型中，決策數有最高的測試準確率，但交叉驗證平均準確率最高的為隨機森林。此外，取部分特徵訓練 ANN、隨機森林、Esemble 的平均準確率都有所提升，故本研究建議使用部分特徵訓練模型，並以隨機森林方法作為肝病判定模型，建議參數設計如前節所述。

表六、以所有特徵進行訓練之模型績效

模型	test	Stratifiedkfold

	accuracy	precision	recall	f1-score	mean test accuracy	std
DecisionTree	87%	97%	77%	86%	83%	0.03
RandomForest	85%	92%	81%	86%	83%	0.06
XGBoost	87%	87%	81%	84%	83%	0.05
Esemble	84%	91%	75%	82%	83%	0.05
ANN	81%	82%	78%	80%	35%	0.31

表七、刪除”Gender”、” Total Proteins”特徵訓練之模型績效

模型	test				Stratifiedkfold	
	accuracy	precision	recall	f1-score	mean test accuracy	std
DecisionTree	86%	94%	77%	84%	83%	0.05
RandomForest	84%	88%	79%	83%	84%	0.05
XGBoost	85%	89%	79%	84%	83%	0.07
Esemble	85%	89%	79%	84%	84%	0.06
ANN	73%	69%	81%	75%	54%	0.41

五、結果與討論

根據本研究改善後所提出的模型，測試集準確率最高可達 0.87%(決策樹)，比初步測試的最高準確率 86%(隨機森林)提升了 1%，平均測試準確率則由 83%提升到 84%，提升了 1%，有小幅度改善。此外，目前 kaggle 挑戰者中無人使用 Gridsearch 尋找最佳參數，並以交叉驗證模型績效。本研究之結果以交叉驗證結果更為具公信力，能輔助醫生進行肝病診斷。

六、未來展望

目前本研究模型平均測試準確率 84%，尚無法達到醫療高準確率之要求。若能取得更多抽血檢查數據，或加入超音波照片綜合評估，或許能提升模型準確率。此外肝纖維化、硬化等有嚴重程度之分，若能在資料標籤上區分目前肝病嚴重程度，未來可以發展判斷項目更詳細之模型。