Intelligent Integration of Enterprise
Final Project

以抽血檢查特徵判定肝臟疾病

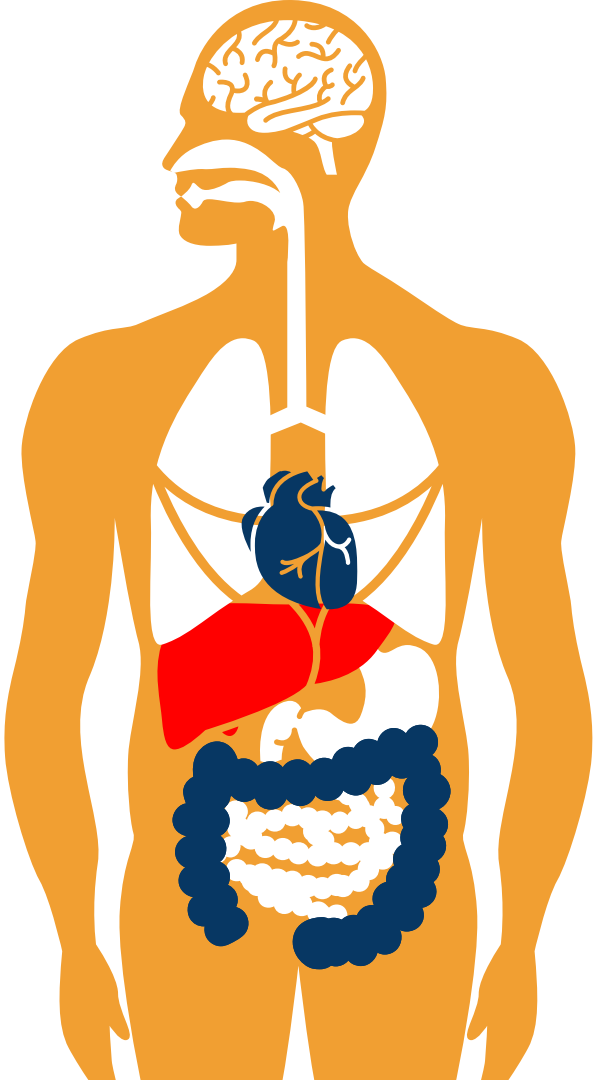
109034536 郭芸如

TABLE OF CONTENTS

背景介紹

慢性肝病及肝硬化位居國人死因第10

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



01 問題背景與定義

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。

資料來源

UCI Machine Learning Repository的ILPD (Indian Liver Patient Dataset) Data Set資料集，583位病患資訊。其中，患肝病者416

	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

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

欄位	說明	統計資訊
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

資料前處理

檢查並刪除缺失值
(4筆)

```
#==================
df.info()
df.isnull().sum()
df=df.dropna()
df.isnull().sum()
df.info()
#==================
```

```
In [9]: df.isnull().sum()
Out[9]:
Age                          0
Gender                       0
Total_Bilirubin              0
Direct_Bilirubin             0
Alkaline Phosphotase         0
Alamine Aminotransferase     0
Aspartate Aminotransferase   0
Total Protiens               0
Albumin                      0
Albumin and Globulin Ratio   4
Dataset                      0
dtype: int64
```

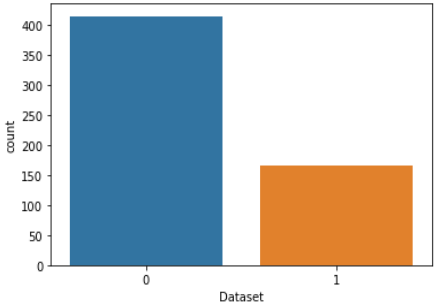
```
Data columns (total 11 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   Age                         583 non-null    int64
 1   Gender                      583 non-null    object
 2   Total_Bilirubin             583 non-null    float64
 3   Direct_Bilirubin            583 non-null    float64
 4   Alkaline Phosphotase        583 non-null    int64
 5   Alamine Aminotransferase    583 non-null    int64
 6   Aspartate Aminotransferase  583 non-null    int64
 7   Total Protiens              583 non-null    float64
 8   Albumin                     583 non-null    float64
 9   Albumin and Globulin Ratio  579 non-null    float64
 10  Dataset                     583 non-null    int64
dtypes: float64(5), int64(5), object(1)
memory usage: 50.2+ KB
<class 'pandas.core.frame.DataFrame'>
Int64Index: 579 entries, 0 to 582
Data columns (total 11 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   Age                         579 non-null    int64
 1   Gender                      579 non-null    object
 2   Total_Bilirubin             579 non-null    float64
 3   Direct_Bilirubin            579 non-null    float64
 4   Alkaline Phosphotase        579 non-null    int64
 5   Alamine Aminotransferase    579 non-null    int64
 6   Aspartate Aminotransferase  579 non-null    int64
 7   Total Protiens              579 non-null    float64
 8   Albumin                     579 non-null    float64
 9   Albumin and Globulin Ratio  579 non-null    float64
 10  Dataset                     579 non-null    int64
dtypes: float64(5), int64(5), object(1)
memory usage: 54.3+ KB
```

Encode
(Gender=、=Dataset=)

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**。

資料擴增
(Gender=、=Dataset=)

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**)**。



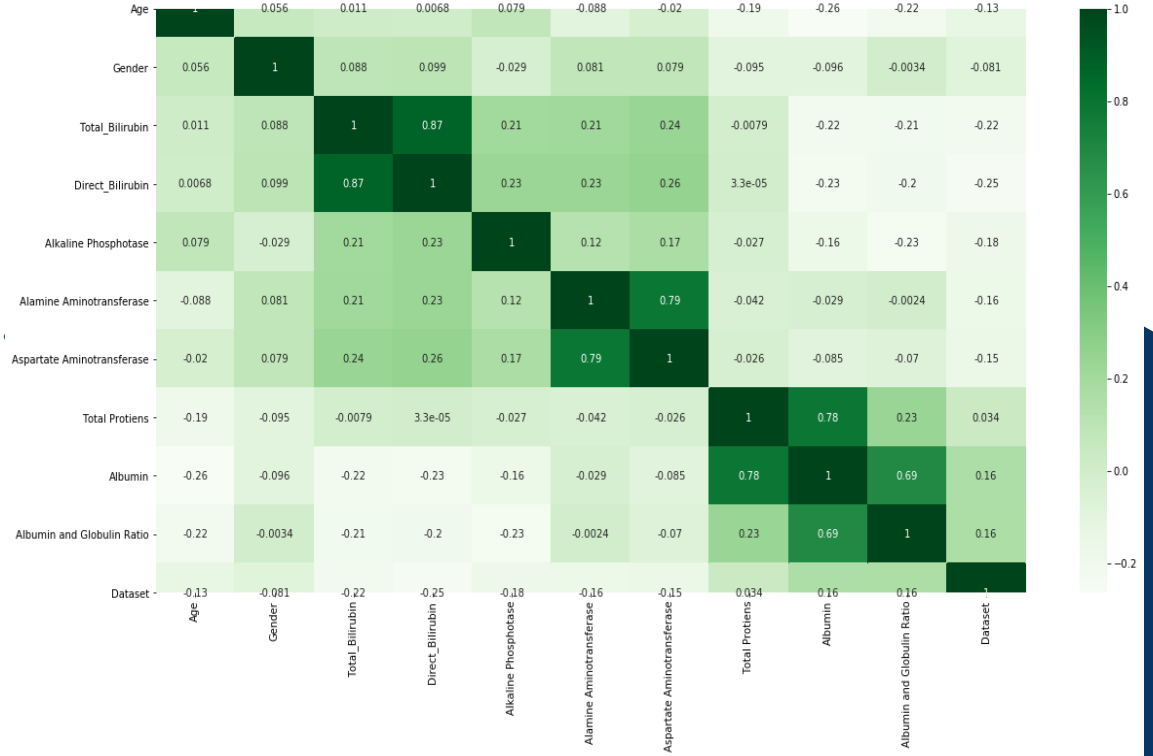
觀察資料相關性

=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探討

1.使用全部特徵

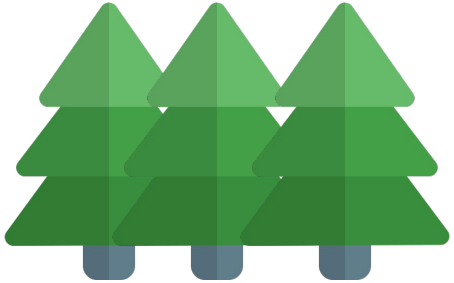
2.使用部分特徵**(**刪除**=Gender=、= Total Proteins=**特徵**)**。



模型

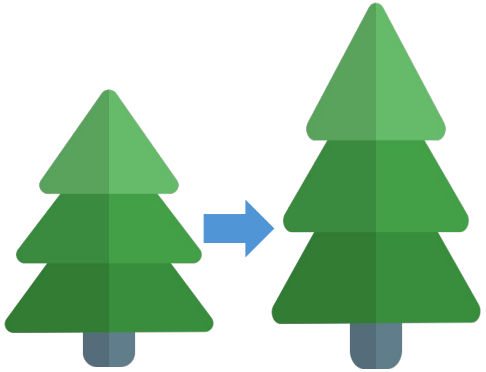
決策樹

機森林



隨

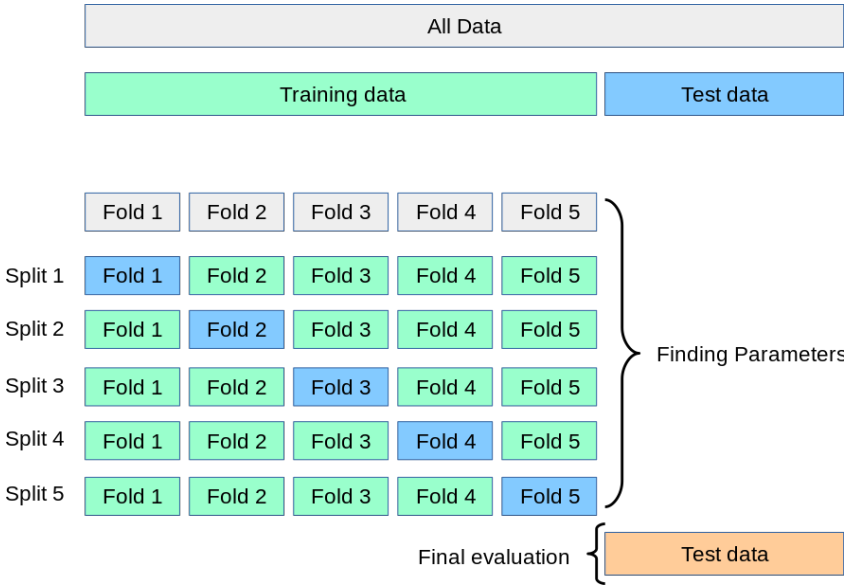
XGBoost



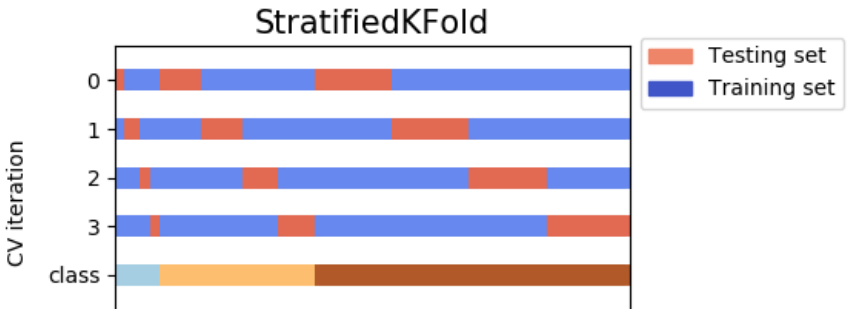
03模型架構分析與改善

交叉驗證

Kfold



Stratifiedkfold



初步測試

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。

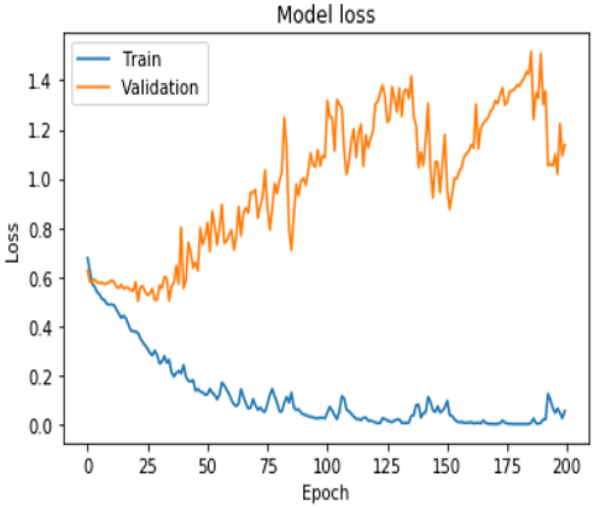
分類器	test accuracy	mean test accuracy	std
LogisticRegression	73%	69%	0.08
DecisionTree	82%	81%	0.05
RandomForest	86%	83%	0.07
XGBoost	81%	76%	0.07
KNN	71%	71%	0.06
SVM	73%	66%	0.05
ANN	81%	35%	0.31

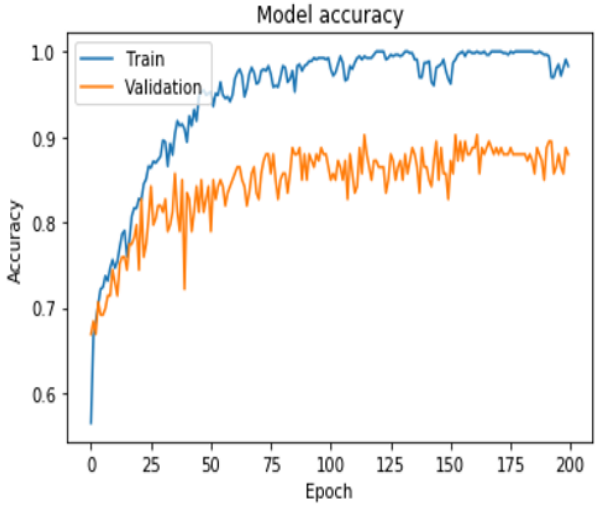
ANN模型深度

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%

ANN模型深度

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。





Ensemble learning

結合多個「弱學習器」來建構一個更強穩模型。
使用**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。

```
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))
```

03模型架構分析與改善

超參數調整

GridSearchCV+Stratifiedkfold (5個fold)

```
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
```

```
...: 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_est
0.759843 (0.016394) with: {'criterion': 'gini', 'max_features': 'auto', 'n_e
0.826373 (0.026849) with: {'criterion': 'gini', 'max_features': 'auto', 'n_e
0.814240 (0.019029) with: {'criterion': 'gini', 'max_features': 'auto', 'n_e
0.818831 (0.031443) with: {'criterion': 'gini', 'max_features': 'auto', 'n_e
0.811267 (0.028048) with: {'criterion': 'gini', 'max_features': 'auto', 'n_e
0.817270 (0.021600) with: {'criterion': 'gini', 'max_features': 'auto', 'n_e
0.796024 (0.034095) with: {'criterion': 'gini', 'max_features': 'sqrt', 'n_e
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}
```

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
	criterion	gini, entropy
	max_features	auto, sqrt, log2
XGBoost	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

超參數調整

GridSearchCV+Stratifiedkfold (5個fold)

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
	criterion	gini, entropy
	max_features	auto, sqrt, log2
XGBoost	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

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%

模型績效比較

取全部特徵，決策樹分類效果最為突出。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。改善模型平均測試準確率比初步測試提升了**1%**。

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

模型(部分特徵)	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

結論

研究成果

- 改善後模型平均測試準確率84%，比初步測試時提升了1%提升了1%，有小幅度改善。
- 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。本研究之結果更具公信力，能輔助醫生進行肝病診斷。

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。
- 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。

THANKS!