Introduction To Data Science Fall 2021 Assignment

一、小組各成員的姓名、系級與學號

1. 何子安，心理 110，E44065020
2. 黃思媛，統計 110，H24064080
3. 羅盼寧，統計 108，H24041066

二、競賽敘述與目標

透過過往的客戶資料進行分析，預測出未來銀行即將可能流失的客戶。其具體的預測結果為，該客戶最終是否會離開銀行（即未來不再與該銀行進行交易）。

競賽中提供的兩個資料集分別為：訓練資料集「*train.csv*」與測試資料集「*test.csv*」。且資料集中紀錄了每個客戶的基本個人資訊與金融行為。

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
0	551	15806307	S2336	720	S2	Male	38	5	114051.97	2	0	1	107577.29	0
1	6897	15709621	S1500	682	S0	Female	54	4	62397.41	1	1	0	113088.6	1
2	4588	15619340	S1865	672	S0	Female	31	5	119903.67	1	1	1	132925.17	0
3	291	15620746	S1672	592	S2	Female	40	4	104257.86	1	1	0	110857.33	0
4	1673	15646372	S2532	753	S2	Male	42	5	120387.73	1	0	1	126378.57	0
5	648	15649129	S1548	575	S0	Female	42	5	104472.9	1	1	1	71641.38	0
6	6113	15729557	S37	572	S0	Male	37	6	135715.66	1	1	0	115928.95	0
7	8957	15579112	S750	753	S1	Female	34	6	124281.61	1	1	0	89136.06	0
8	1678	15680895	S2079	546	S0	Male	46	3	62397.41	2	1	1	79809.09	0
9	5202	15580935	S252	657	S0	Male	45	4	141238.54	2	0	0	95281.51	0
10	4868	15738150	S1717	617	S2	Male	35	4	62397.41	2	1	1	132607.99	0

附表一，訓練資料集，其中包含需預測的目標項「Exited」。

Feature Name	Meaning
RowNumber	ID of each bank's customer record
CustomerId	ID of each bank's customer
Surname	Anonymized surname of the customer
CreditScore	Credit score of the customer. Higher credit score means better banking behaviors
Geography	Anonymized zip code of the customer
Age	Age of the customer
Tenure	Number of tenures (不動產) of the customer
Balance	Amount of money the customer has in a bank account
NumOfProducts	Number of financial products of the customer
HasCrCard	Does the customer have credit card? (1: True, 0: False)
IsActiveMember	Whether the customer has frequent transactions in the bank? (1: True, 0: False)
EstimatedSalary	Estimated and perturbed salary money of the customer
Exited	Whether the customer does eventually leave (exit) the bank? (1: True, 0: False)

附表二，資料集特徵項的對應解釋。

訓練資料集包含 13 欄特徵項，及 1 欄目標項「Exited」（顧客是否離開，即預測目標）；而測試資料集則只包含一樣的 13 欄特徵項，依照這些特徵項去預測出目標項。

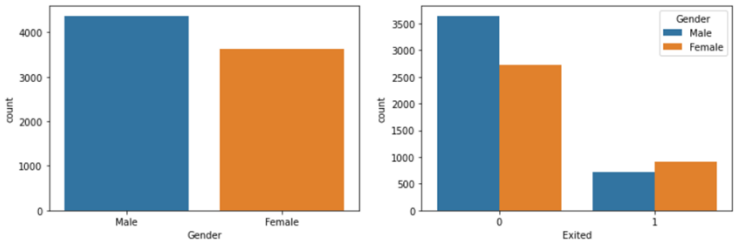
上傳預測結果時，需按照提供的規格「*upload.csv*」將結果上傳至競賽網站。網站上有即時各組於「Accuracy」（準確度）、「Precision」（精準度）、「F-Score」（F 分數）項目作排名，並且依據所上傳的檔案中最新、最佳的預測結果計算出排名。

競賽最終排名是以三項指標的加權平均計算，權重分別為「Accuracy」30%、「Precision」30%、「F-Score」40%。其中「F-Score」的比重最重，意味著我們不只需注重準確地預測出目標項（0/1）的結果，同時也希望能找出銀行將會流失的是哪些客戶。

三、特徵處理與分析

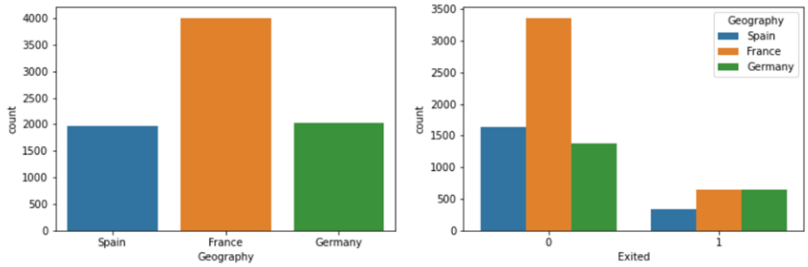
特徵項之分析

藉由 *Gender* 這個變數，我們可以發現資料中男性的比例大於女性，但以最終是否離開銀行區分的情況下，則會發現女性最終離開的比例較高。



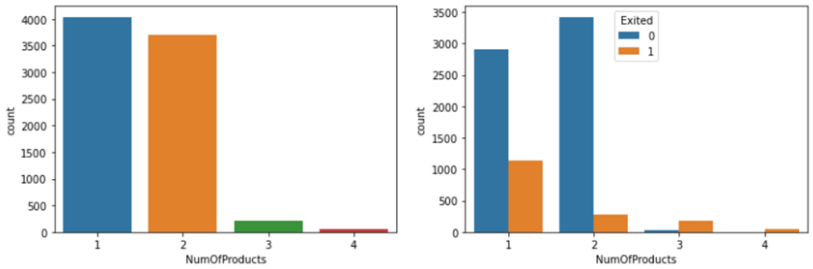
附圖一

從 *Geography* 此變數裡，看見 *France* 比例最高，但只看最終選擇離開銀行的客戶時，會發現 *France* 與 *Germany* 的人數不相上下。



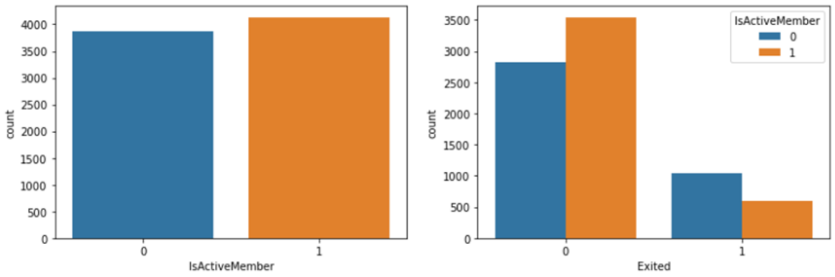
附圖二

在變數 *NumOfProducts* 中可以看到大多數顧客擁有一至二項金融產品，而擁有三項以上的金融產品的顧客，比較可能在最終選擇離開銀行。



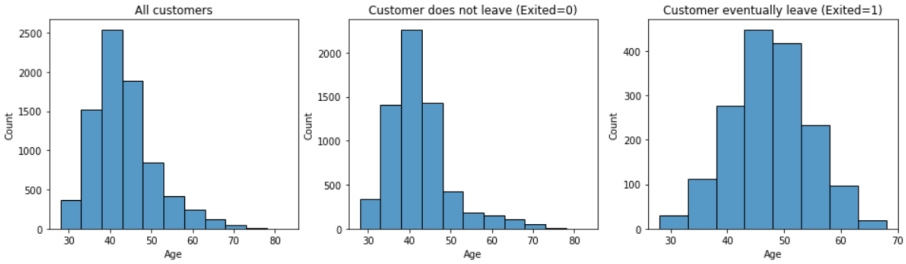
附圖三

IsActiveMember 這項變數顯示在銀行中有頻繁交易的客戶略多於沒有頻繁交易的客戶，而沒有頻繁進行交易的客戶，有較高可能在最終選擇離開銀行。



附圖四

從變數 *Age* 得知客戶大多落在 40 歲左右，但最終選擇離開的客戶年齡層稍高於未選擇離開的客戶。



附圖五

特徵處理

1. 有嘗試將資料中的類別資料轉換為以數字型態表
   - *Gender*：(Male, Female) 換為 (1, 0)，*Geography*：(Spain, France, Germany) -> (2, 1, 0)
2. 之後則嘗試將資料中的類別資料以 one-hot encoding 來表示
3. 在供模型訓練的特徵上去除 *RowNumber*, *CustomerId*, *Surname*；以 *CreditScore*, *Geography*, *Gender*, *Age*, *Tenure*, *Balance*, *NumOfProducts*, *HasCrCard*, *IsActiveMember*, *EstimatedSalary* 共 10 維特徵作訓練資料。
4. 在共 8000 筆的資料中，再將其分為 6000 筆訓練資料和 2000 筆測試資料。（test size = 0.25）
5. 訓練模型的特徵也對其做標準化
6. 在 DNN 模型上會對 *HasCrCard* 與 *IsActiveMember* 的值從（0，1）換為（-1，1）。
7. 在 DNN 的模型訓練上會對 label（0，1）做 one-hot encoding 為讓模型輸出兩個值。
8. 在 DNN 模型上資料處理流程中有使用 Synthetic Minority Oversampling Technique （SMOTE）來處理資料不平衡的方式。

四、預測訓練模型

Boosting 方法：

1. XGBoost，CatBoost - 透過人工調整參數的方式，試圖進步模型。調整參數時，先選定一個參數後，透過嘗試此參數在不同數值下建立出的模型結果，記錄下 Accuracy Score, Precision Score, F1 score，選擇最佳的 final score 作為調整此參數的依據。

learning_rate	Max_depth	subsample	colsample_bytree
0.01	5	0.88	0.85

附表三，XGBoost 參數設定。

learning_rate	Max_depth	subsample
0.01	6	0.8

附表四，CatBoost 參數設定。

2. Random Forest - 一次只設定一種參數，使用與網站排行相同計算方式之加權分數找出最佳的參數設定值。若該模型之加權分數與基本模型之加權分數相比，並無顯著提升(設定為 0.1)，代表該參數對於提升模型效果無明顯影響，則後續的探討不列入使用。比較使用多項參數的模型，之間的表現結果，找出最佳模型。評估測試表現的資料集是「自切訓練模型」（訓練：測試=6000:2000）。

名稱	n-estimators	criterion	max_depth	min samples split	min samples leaf
最終選定的值	165		12	17	12
備註		沒差		（最終使用的參數）	

名稱	max features	class weight	max samples	n_jobs
最終選定的值	2	{0:6, 1:1}	628	
備註		無明顯效果		無明顯趨勢

附表五，Random Forest 參數設定。

其他：

1. SVM – 逐個嘗試該模型可能的幾種參數調整，同樣以「自切訓練/測試資料集」來評估模型的效能。如測試結果有更佳的表現再上傳至網站進行公開排行榜的評估。

	Kernel	Categorical to numeric	Categorical to One-hot
模型基線參數	rbf	o	x
最終模型參數選擇	rbf	x	o

附表六，SVM 參數設定。

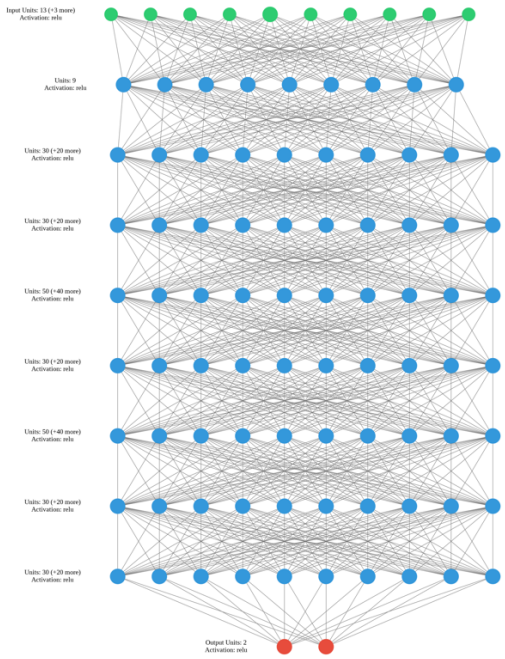
2. Deep Learning – 先訓練基線模型，再盡可能逐個嘗試調整該模型的參數，同樣以「自切訓練/測試資料集」來評估模型的效能。如測試結果有更佳的表現再上傳至網站進行公開排行榜的評估。全連結層之間皆使用 relu 激活函數，除輸出層使用 sigmoid 激活函數。

metrics	設定	activation	設定	early stopping	設定
accuracy	none	input layer	relu	monitor	val_accuracy
precision	thresholds=0.7	hidden layers	relu	min delta	0.01
f1-score	num_classes=2	output layer	sigmoid	patience	300
				verbose	1
				model	max

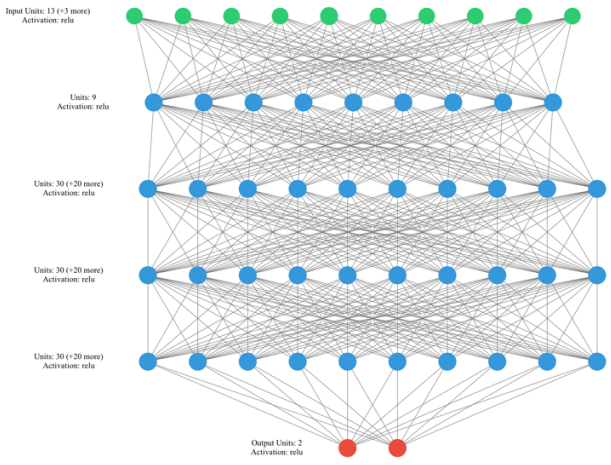
附表七，DNN 模型超參數設定。

compile	設定	fit	設定
loss	binary_crossentropy	epochs	1000
optimizer	rmsprop	batch_size	50
metrics	accuracy, precision, f1-score	callbacks	early_stopping

附表八，DNN 模型其他超參數設定。



附圖六，DNN 模型基線架構，其中有三層使用 L2 regularizer。



附圖七，最終使用的 DNN 模型架構，只有其中一層使用 L2 regularizer。

五、預測結果分析

XGBoost，CatBoost：

XGBoost(split train data)	Accuracy Score	Precision Score	F1 score
Label Encoding	85.65%	68.40%	56.18%
未調整參數	85.75%	67.47%	57.75%
learning_rate 調整後	87.30%	77.50%	59.42%
max_depth 調整後	87.40%	78.15%	59.62%
subsample 調整後	87.55%	78.19%	60.41%
colsample_bytree 調整後	87.55%	79.65%	59.64%

附表九

CatBoost(split train data)	Accuracy Score	Precision Score	F1 score
Label Encoding	87.35%	75.29%	61.02%
未調整參數	87.45%	76.06%	61.29%
learning_rate 調整後	88.15%	79.22%	63.03%
max_depth 調整後	88.15%	79.22%	63.03%
subsample 調整後	88.15%	79.22%	63.03%

附表十

XGBoost(upload)	Accuracy Score	Precision Score	F1 score
Label Encoding	0.86	0.6615	0.6056
未調整參數	0.8725	0.7407	0.6107
調整參數後	0.88	0.7959	0.6195

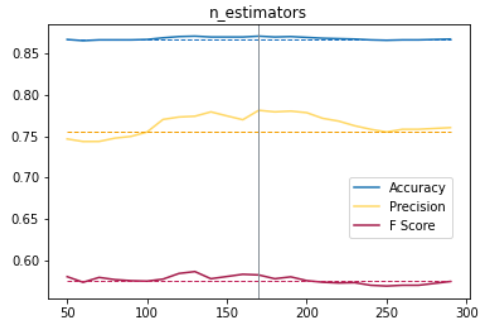
附表十一

CatBoost(upload)	Accuracy Score	Precision Score	F1 score
Label Encoding	0.87	0.6984	0.6286
調整參數後	0.87	0.7119	0.6176

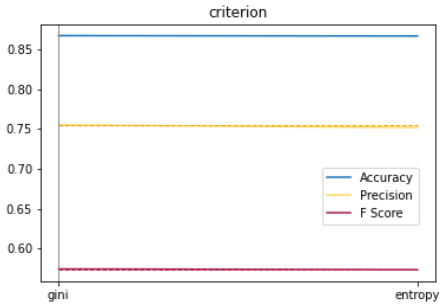
附表十二

可以發現總體來說，CatBoost 的表現比較容易優於 XGBoost，但因為 CatBoost 本身就有自動調整參數的能力，也因此在後續人工調參數時的進步不大。

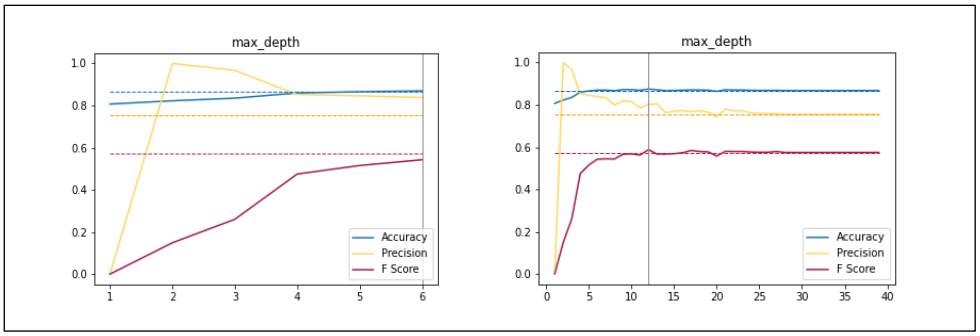
Random Forest：垂直線為表現最佳之模型使用的參數值，水平虛線為基本模型(無調整任何參數)的評估表現。



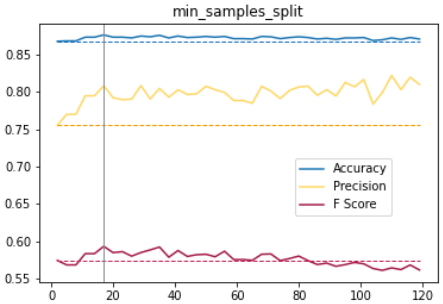
附圖八，n_estimators 100-200 時 precision 較基本模型好，accuracy 也有些微提升。



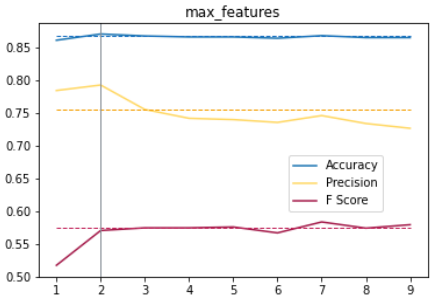
附圖九，criterion 對模型表現並無影響。



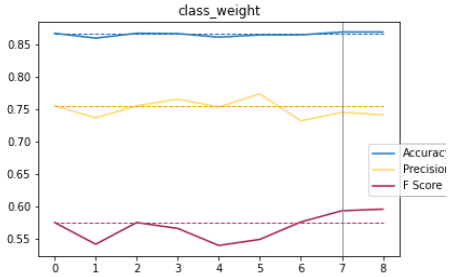
附圖十，max_depth 在 4 之後較靠近基本模型表現，隨著 precision 漸降與 f score 漸升，在 12 時表現最好；25 之後幾乎和基本模型沒差別，推測因為無限制的決策樹之深度也都不會超過 25，所以參數的設定與否不影響結果。



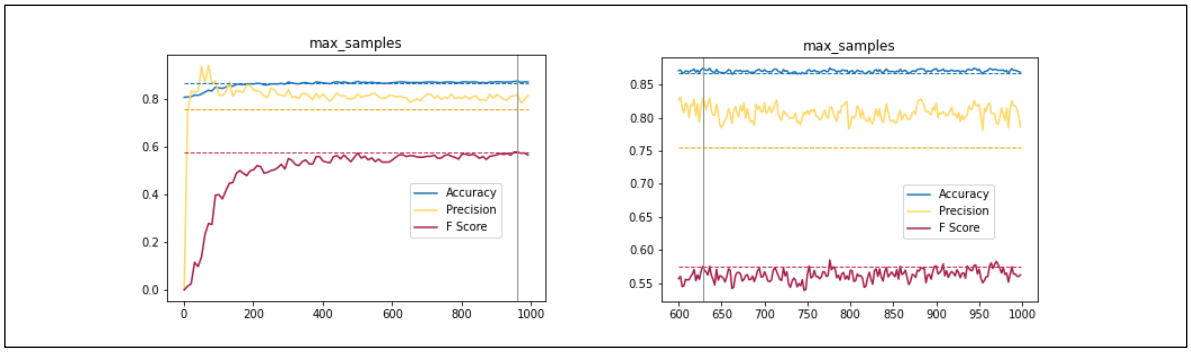
附圖十一，min_samples_leaf 在 accuracy 先升後降；precision 大幅提升；f score 則是隨值遞減，在 16 左右之後表現甚至不如基本模型。因此最佳的值落在 precision 表現佳而 f score 又還沒降低時的 12。



附圖十二，max_features precision<3 的表現比基本模型好，>3 則較差，f score 則是 1 表現差、>2 之後與基本模型差不多，因此選擇綜合表現最好的 2，推測原因為主要影響的特徵值有兩個。



附圖十三，1(離開銀行)比重較重時，precision 表現較好，但 f score 表現較差；0(不離開銀行)比重較重時，則相反。但整體模型表現與基本模型差異太小，因此最後沒有使用這個參數調整。



附圖十四，f score 相較基本模型表現較差，遞增至 600 後就無明顯進步。三項指標在>200 後有較穩定的趨勢(值太小時會有 overfitted 的問題)，第二張圖可看出來>600 無特別趨勢改變(應該都是隨機值造成的小變動)。

	parameter	value	model score	induced
0	n_estimators	165	0.7328	0.0161
1	max_depth	12	0.7383	0.0216
2	min_samples_split	17	0.7420	0.0253
3	min_samples_leaf	12	0.7368	0.0201
4	max_features	2	0.7271	0.0104
5	class_weight	{0: 6, 1: 1}	0.7215	0.0048
6	max_samples	628	0.7413	0.0246

附圖十五，”induced”為與基本模型相比提升之加權分數，可以看出“min_samples_split” 、 “max_samples” 有最好的表現。

雖然相較基本模型都有較好的表現，但是卻沒有比「只用單一參數的模型」好，因此最後選擇表現最好的" min_samples_split" 參數。最終 random forest 模型的表現：accuracy 為 0.8760，precision 為 0.808，f score 為 0.5934。使用最終模型預測的競賽資料表現(網站提供)：accuracy 為 0.88，precision 為 0.7736，f score 為 0.6308。

DNN：由於 DNN 模型的超參數眾多，因此在調整超參數時並沒有一一紀錄下來，但從一次次的調整當中也能夠感受都哪些超參數對模型的效能有著較大的影響。一開始的基線模型就使用了 8 層隱藏層，其中的神經元數量也被設為 30 或 50。在逐步增加每一層的神經元後發現模型準確率並沒有得到顯著的提升，反而在神經元數量越來越多之後準確率就開始緩慢下降。但在逐步加深隱藏層之後，卻能得到模型準確率會明顯下降的趨勢。由此可得在 DNN 模型效能上，調整隱藏層的數量比起調整神經元的數量更能影響 DNN 的表現。原先的訓練 *batchsize* 是設為 200，但參考台大李宏毅於 Youtube 上課程的說明，小 *batchsize* 對 DNN 的表現會更勝於大 *batchsize*（在不考慮 GPU 的情況下，本人也沒有 GPU）。另外，在隱藏層與神經元數量都很多的時候，模型較容易出現 Over-fitting 的（即在訓練集上的表現都非常好，但在測試集上的表現都不好）。在最先出現 over-fitting 的時候，想到的方法是在隱藏層中加上 *regularizer* 來某程度上壓抑模型的表現，便在最開始的 8 層隱藏層中的其中 4 層加上 *regularizer*（0.01），但似乎模型的表現被壓制太多以至於在訓練集上也得不到比較好的效果。所以最終選擇只在 5 層隱藏層中的其中一層有加上 *regularizer*，同時這也是模型最佳的表現的一次。

SVM：意外得到最佳結果的模型。除特徵處理之外，其餘模型的超參數都使用預設的數值（*kernel=rbf*，*class_weight=None*），就得到了所有上傳結果中最好的一次。另外，在分割資料集為訓練 6000 筆，測試 2000 筆時，所得到的效能表現會比直接用 8000 筆來訓練並在網站上進行評估的結果來得要好。由於時間關係，並未有時間得出結論是什麼原因導致這樣的情況。個人推測為在切分訓練與測試資料集時所使用的 *random_state* 參數剛好能將資料切分為對未知資料所能得到最好表現的分佈。

六、感想與心得

何子安 心理 110 E44065020：

自己不是相關領域背景下的學生，第一次上統計系的課實在是滿滿的收穫，每一次上課都覺得很神奇，因為對我來說都是新知識。但每次上課看到人數越來越少，為的內心真的是很恐慌，會害怕什麼時候就輪到我撐不住要說再見，但還好我活下來了。非常感謝隊友滿滿的耐心與熱情，可以每次都幫我解惑，惡補相關的背景知識。也終於見識到了政德老師的魅力，難怪每次開課時都爆滿。期待接下來還有機會再繼續修政德老師的課，跟著老師的腳步一步步稱霸資料科學領域！整個課程中我覺得最棒的是不但老師的課程盡可能包含了所有基礎的內容讓大家可以很好的入門，還有助教每星期的「課外」知識講解，補足了我很多即使不是主要，但卻大大有用的知識/技巧。也多虧課程中的 html，css 作業，讓我有機會接下一位老師的專案，幫忙設計一位醫生的前端網頁。

黃思媛 統計 110 H24064080：

在這次作業中，我主要負責 Catboost 和 XGBoost 的部分，雖然之前也曾經使用過這兩個 package，但當時都僅僅是直接使用預設的數值，這是我第一次試著調整參數。雖然最後因為時間的關係，在來不及了解交叉驗證的程式碼該如何設置的情況下，放棄了這個部分，但看著做為測試集的部分，一點一點地提高預測的成績，仍然非常有成就感。想給競賽網站的建議有三個，主要都是對上傳歷史紀錄的部分：一是希望可以把 final score 顯示出來，我覺得這個成績也能作為一種參考；再來則期望能標記出目前在排名頁面的（本組當前最佳結果）是哪一次上傳結果，如果要繼續改進時，比較容易確認要找哪一份程式碼，而不用一一對照各個分數找到數值相符的那個；最後還冀望能在這個頁面也擁有像是排名頁面一樣的選擇排序方式功能，讓我們可以看到依照上傳時間、某項特定成績等等的不同上傳結果排序。

羅盼寧 統計 108 H24041066：

首先謝謝組員的 cover 和包容，透過與他們的討論有很棒的學習！這次競賽我有很多收穫，最多的部分就是 sklearn 和 python 的熟悉與進步吧。雖然是作業形式，但是卻是一個沒有標準解答，也沒有滿分成果的競賽，自由度很高的同時也讓人更有動力自學新的東西。覺得很有趣的是，跑出模型的程式只有短短不到 10 行(不計算資料前處理的話)，這當中該如何設定的參數卻可能要花上百行程式去找出來。每次上傳完資料看評估的數據時心情真的是既期待又怕受傷害~~(難道這就是統計的浪漫)~~。然後排行榜能夠看到往年組別的同學表現這點，我覺得很讚。最後就是，競賽網站雖然陽春，但上傳預測資料完都會回饋一個可愛的貓貓圖，我很喜歡 XD 或許可以多放幾張不同的療癒圖每次隨機出現，增加上傳資料的動力?!

七、工作分配

何子安	SVM，DNN，報告整合，特徵處理
黃思媛	XGBoost，CatBoost，特徵處理與分析，資料分析流程建議
羅盼寧	RandomForest，特徵處理與分析，資料分析流程建議

八、Github

何子安 心理 110 E44065020：

Github Repository:
https://github.com/onnnnn/Intro_DS_2021
Github Page:
https://onnnnn.github.io/broccoli/templates/index.html

黃思媛 統計 110 H24064080：

GitHub Repository:
https://github.com/co24064080/hw-of-introduction-to-data-science
GitHub Page:
https://co24064080.github.io/hw-of-introduction-to-data-science/HW3/html_css/hw3.html

羅盼寧 統計 108 H24041066：

Github Repository:
https://github.com/Loplumning/2021DataScience
Github Page:
https://loplumning.github.io/2021DataScience/hw3/my%20page.html