Final Project Proposal

Team member

109061517 邱俊嘉 / 109061520 陳俊宇 / 109061807 吳亞澤

Deadline: 11/24

Proposal format:

PDF

-Title

臺灣海洋廢棄物預測

-Methods

1 資料處理與分析

1.1 train dataset

下圖為 AIea 提供的 train dataset 之數量分布

level	number
1	11
2	46
3	29
4	28
5	43
6	48
7	51
8	34
9	20
10	9
total	319

總共有 319 筆 sample，共 10 種 level(海廢等級)，各 level 的數量分布不均勻，因此會做 upsample 讓每個 level 的數量都複製增加到 51 筆，所以最後總 train data 數會是 510 筆，這樣可以避免 data imbalance 的問題，不然 model 會傾向猜數量最多的那類

1.2 缺失值以-1 取代

某些 sample 的部分特徵值可能會有缺失，像是下圖中的紅框處，此時我們會以-1 來取代這種缺失的特徵值

1暴露岩岸	2暴露人造結構物	3暴露岩盤
0	0	1
0	0	1
0	0	1
nan	nan	nan

➔

1暴露岩岸	2暴露人造結構物	3暴露岩盤
0	0	1
0	0	1
0	0	1
-1	-1	-1

1.3 特徵值若為中文或英文就以整數代號取代

station、location 和 country 這三個 feature 的特徵值為中文或英文，這會讓 model 無法學習，所以我們將其改成整數代號，如下圖所示

Station	Season	County	Location
E02	1	宜蘭縣	大溪
E02	2	宜蘭縣	大溪
E02	3	宜蘭縣	大溪
E02	4	宜蘭縣	大溪
E03	1	宜蘭縣	頭城
E03	2	宜蘭縣	頭城
E03	3	宜蘭縣	頭城
E03	4	宜蘭縣	頭城
E05	1	宜蘭縣	清水港尾
E05	2	宜蘭縣	清水港尾
E05	3	宜蘭縣	清水港尾
E05	4	宜蘭縣	清水港尾
E06	1	宜蘭縣	無尾港

➔

Station	Season	County	Location
0	1	0	0
0	2	0	0
0	3	0	0
0	4	0	0
1	1	0	1
1	2	0	1
1	3	0	1
1	4	0	1
2	1	0	2
2	2	0	2
2	3	0	2
2	4	0	2
3	1	0	3

1.4 將所有 feature 做個別的 z-score standardization

因為不同 feature 的數值分布範圍差異甚大，某些 feature 只有 0 或 1，某些 feature 從 0 到 87，如果不做 standardization 的話，某些 feature 值會 dominate 整個 model 的學習，導致 model 學不起來，所以我們會做 z-score standardization，其公式如下圖所示，每個 feature 轉換後都會變成平均值為 0、標準差為 1 的標準高斯分布

假設資料的平均數與標準差分別為 μ 及 σ ，Z分數標準化可利用下列公式進行：

$$Z = \frac{X - \mu}{\sigma} \sim N(0,1)$$

2 模型架構

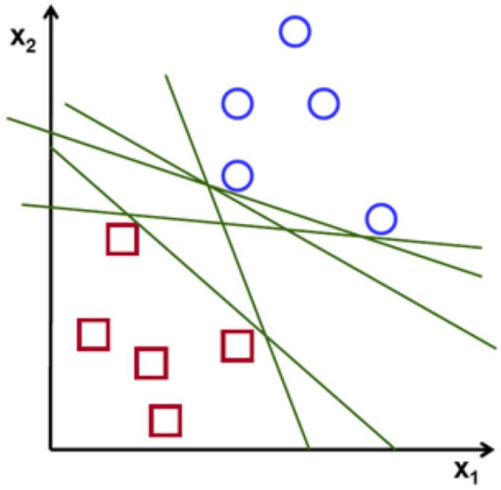
我們的 model 主要分成 2 大類，一類是機器學習方法，像是 SVM、Random Forest，另一類是使用 neural network 的深度學習方法，像是 DNN，下面舉幾個比較有名的架構來做細部說明

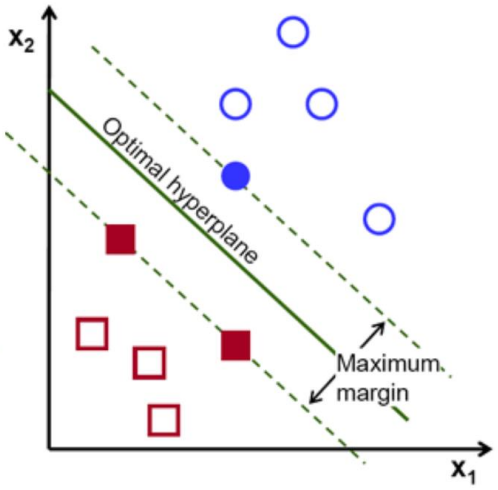
2.1 支援向量機(Support vector machine)(SVM)

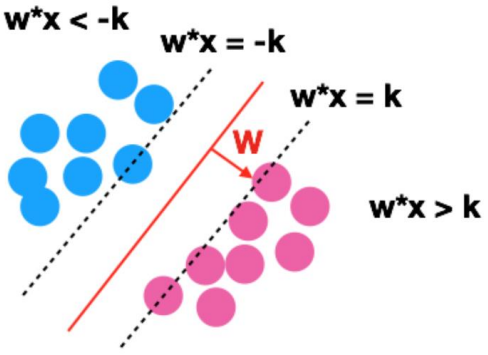
2.1.1 目的

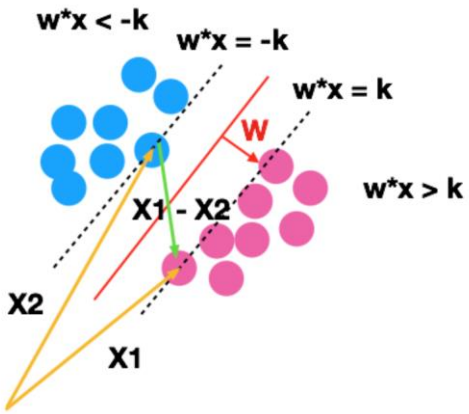
SVM 是一種有堅實理論基礎的新穎的小樣本學習方法，它基本上不涉及概率測度及大數定律等，因此不同於現有的統計方法，從本質上看，它避開了從歸納到演繹的傳統過程，實現了高效的從訓練樣本到預測樣本的「轉導推理」，因此大大簡化了通常的分類和迴歸等問題，故也選用此法

2.1.2 原理









$$\text{Margin} = \frac{W * \overrightarrow{X1 - X2}}{2\|w\|} = \frac{2k}{2\|w\|}$$

$$J(\theta) = C[\sum_{i=1}^{m} y^{(i)} Cost_1(\theta^T(x^{(i)}) + (1 - y^{(i)}) Cost_0(\theta^T(x^{(i)})] + \frac{1}{2}\sum_{j=1}^{n} \theta_j^2$$

$$m = number\ of\ samples, \quad n = number\ of\ features$$

虛線上的點 X1,X2 其實就是所謂的支援向量(Support vector)，我們主要是利用支援向量來算出 Margin，並最大化 Margin(最小化 Cost Function)

2.1.3 流程

載入 train data → 將特徵資料標準化 → 載入 SVM 中的 SVC，並設定 kernel（線性或非線性），並將 Probability 設為 True → 分析預測結果。

2.1.4 優缺點

優點:

1. 切出來的線很漂亮，擁有最大 margin 的特性
2. 可以很容易透過更換 Kernel，做出非線性的線（非線性的決策邊界）

缺點:

1. 效能較不佳，由於時間複雜度為 $O(n^2)$當有超過一萬筆資料時，運算速度會慢上許多

2.2 隨機森林模型(Random Forest)(RF)

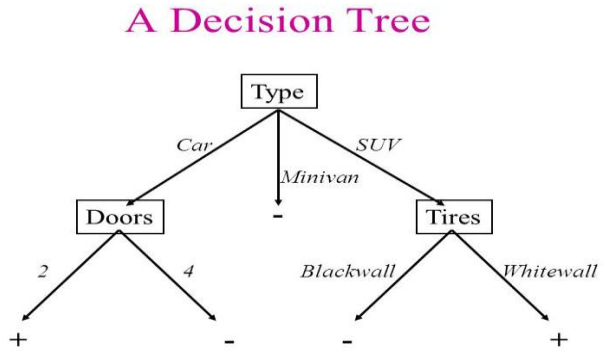
2.2.1 目的

random forest classifier 的時間複雜度較低，為 O(n*log(n)*d*k)，也更適合處理高維度的資料。在分析上能顯示各個特徵的重要性，還能利用 oob error 快速評估模型的表現，非常適合用於本次的實驗

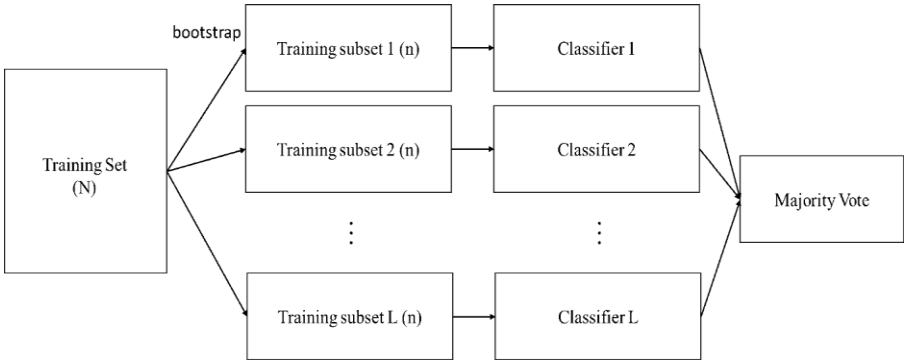
2.2.2 原理

在講 Random forest 之前，必須要先介紹 Decision tree 和 Bagging 這兩個概念

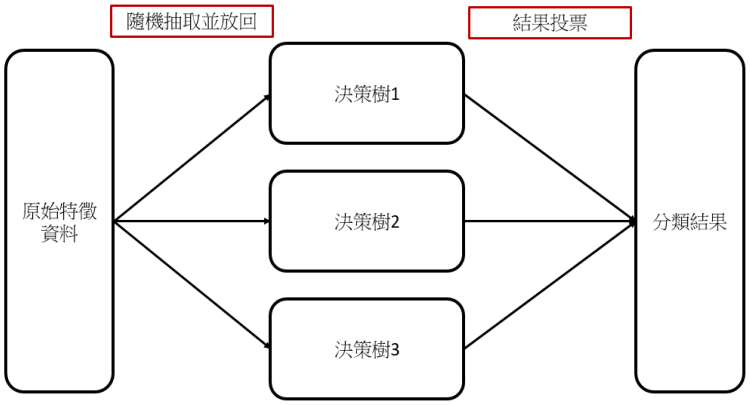
Decision tree：透過一層一層的決策，逐步篩選出符合的結果，如下圖所示



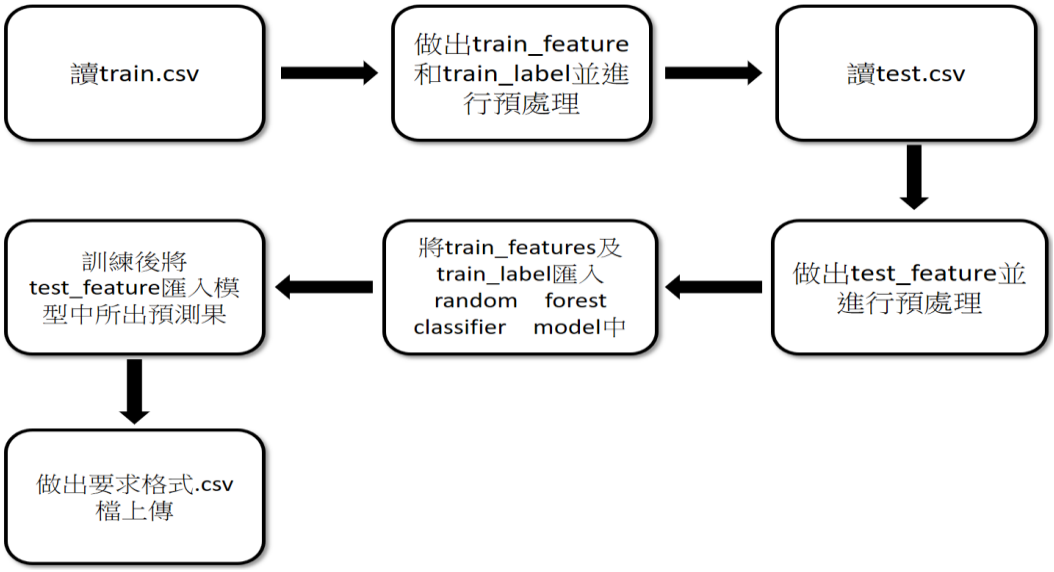
Bagging：從訓練資料中隨機抽取，取出後放回(bootstrap)，利用抽出之樣本訓練多個分類器，每個分類器的權重一致，最後用投票方式(Majority vote)得到最終結果，如下圖所示



Random forest：是一個結合 bagging 及 decision trees 的演算法，如其名字所述，由多個 decision tree 所組成，每顆決策樹獨立運算出結果，並透過投票得到最後的分類結果，如下圖所示



2.2.3 流程



2.2.4 優缺點

優點

1. 時間複雜度低，適合處理大量高維度的資料
2. 附有 feature importance 以及 oob error 等有利分析的功能
3. 訓練速度快
4. 能夠平衡失衡資料集的誤差
5. 對於缺失值以及離群值的敏感度低
6. 能夠避免 overfitting
7. 能解決回歸與分類兩種問題

缺點

1. 對於資料數量少，或是低維度的資料，分類結果較差
2. 在某些 noise 較大的分類或迴歸問題上會過擬合
3. 相對於 Decision tree，需要更長的時間以及更多的儲存空間作運算

2.3 深度神經網路(Deep Neural Network)(DNN)

2.3.1 目的

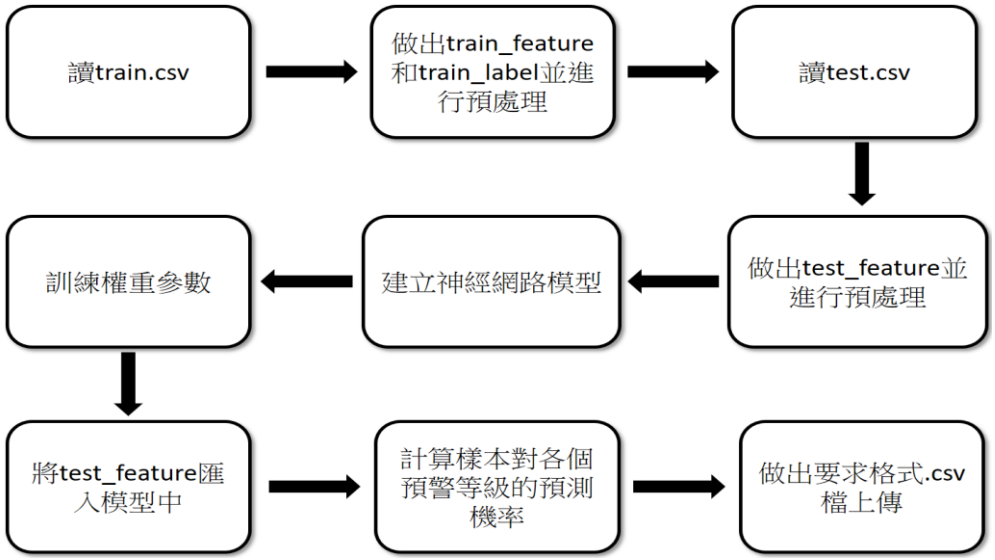
在機器學習的世界中，神經網路就像是人類的大腦神經結構，而神經元就像是大腦的神經細胞，是神經網路中最基礎的結構，在它們互相結合下，可以建構龐大的運作網路，能大大的提高預測準確率，也非常適合處理分類問題，故也選用此方法

2.3.2 原理

1. 建立輸入層、多層隱藏層、輸出層和每個層的神經元，每個神經元都有一個輸出，其輸出稱為激勵值(介於 0~1)，每一層神經元中激勵值的操作結果會影響下一層的激勵值，一層一層之間激勵值的傳遞最後即可得到輸出判斷結果

2. 前一層的每個神經元的輸出分別乘其對應的「權重值」最後相加 ➔ 減下一層神經元的「偏置值」➔ 帶入激勵函式將其轉為介於 0~1 的「激勵值」
3. 數學形式:
   $0 \leq$ 激勵值 $= f(a1w1+a2w2+\cdots\cdots+anwn - b) \leq 1$

2.3.3 流程



2.3.4 優缺點

優點

1. 可以建構非線性的模型
2. 有良好的推廣性，對於未知的輸入亦可得到正確的輸出
3. 可以接受不同種類的變數作為輸入，適應性強
4. 可應用的領域相當廣泛

缺點

1. 以迭代方式更新鍵結值與閥值，計算量大，相當耗費電腦資源
2. 訓練的過程中無法得知需要多少神經元個數，太多或太少的神經元均會影響系統的準確性，因此往往需以試誤的方式得到適當的神經元個數

3 初步實驗結果

除了上面詳細說明的方法之外，我們還試了很多其他 ML 的方法，下圖一的實驗結果為對 train data 做 5-fold cross validation 後的估計結果，圖二為 Gradient Boosting Classifier(GBC)在 AIdea public leaderboard 的評估結果

圖一

Model(5-fold cross validation)		Accuracy	AUC	Recall	Precision	F1	Kappa	MCC	LogLoss	TT (Sec)
ML	Gradient Boosting Classifier(GBC)	32.49%	47.08%	28.47%	29.09%	29.53%	0.2262	0.231	1.8104	0.188
	Linear Discriminant Analysis(LDA)	31.00%	50.45%	27.07%	30.78%	29.39%	0.2167	0.2211	6.3248	0.006
	Random Forest Classifier(RFC)	29.29%	48.10%	25.57%	25.89%	26.11%	0.19	0.1943	1.646	0.046
	Ridge Classifier(RC)	28.92%		23.63%	24.27%	25.68%	0.1868	0.1906		0.003
	Extra Trees Classifier(ETC)	27.87%	47.97%	26.25%	24.53%	24.81%	0.1767	0.1806	2.5934	0.05
	Naive Bayes(NB)	27.17%	44.66%	24.91%	29.14%	26.04%	0.1794	0.1845	17.2528	0.006
	Light Gradient Boosting Machine(LGBM)	26.53%	47.63%	21.70%	23.01%	24.11%	0.1597	0.1623	2.0362	0.121
	Extreme Gradient Boosting(EGB)	26.15%	46.63%	22.08%	22.82%	23.89%	0.1553	0.158	2.0384	0.602
	Logistic Regression(LR)	26.11%	46.72%	21.00%	21.29%	23.14%	0.152	0.1543	1.6226	0.514
	Decision Tree Classifier(DTC)	23.07%	39.49%	19.86%	24.17%	22.28%	0.1252	0.127	18.5922	0.006
	Ada Boost Classifier(ABC)	22.27%	41.07%	15.48%	9.32%	12.70%	0.0875	0.1054	1.6858	0.02
	K Neighbors Classifier(KNC)	20.21%	40.81%	17.53%	19.04%	18.16%	0.09	0.0922	11.132	0.279
	Quadratic Discriminant Analysis(QDA)	19.17%	37.49%	15.80%	14.89%	14.88%	0.0746	0.0789	19.6982	0.006
	SVM - Linear Kernel	13.58%		10.94%	2.88%	4.36%	0.0073	0.0059		0.01
DL	DNN	12.85%	51.44%	4.70%	12.14%	6.47%	0.06865	0.015131	2.3361886	

圖二

Model(AIdea public leaderboard)		Kappa
ML	Gradient Boosting Classifier(GBC)	0.390643

4 分析

- 到目前為止結果都不是很理想，可能是因為有效 train data 數只有 319 筆的關係，導致 DNN 這種 DL 方法無法發揮，因此表現是最差的
- ML 方法雖然普遍表現較好，但是應該還有很大的進步空間
- Gradient Boosting Classifier(GBC)的表現最好，之後應該會由這個架構去做後續改良

5 未來工作

未來工作主要分為改良 feature 和改良 model 兩大方向

5.1 改良 feature

根據我們過去的經驗用 AI 方法來處理問題時，feature 是非常重要的，同一個 train data 如果處理成不同 feature，最後的 performance 甚至可能會差到十幾%，所以 feature 要如何改良應該會成為關鍵

5.1.1 缺失值處理: 以同一個地點的其他 3 個 season 的眾數來填補缺失值

我們目前是用-1 來取代缺失值，這個做法可能對於 model 的學習來說不太好，而且其實缺失值還蠻多的，佔了快 25%，應該要更妥善處理，所以我們後來有想到一個可能的解決方案，那就是以同一個地點的其他 3 個 season 的眾數來填補缺失值，因為經過觀察，發現缺失值都是海岸地形種類，從直觀上來想，海岸地形因為會影響人類活動，理論上來說應該也會和海廢等級有關，因此海岸

地形這個 feature 應該要用到，所以缺失值勢必要用某個值去填補，不能直接去掉。此外，我們還發現缺失值會發生在每一個地點的第 4 個 season，而且同一個地點的其他 3 個 season 的海岸地形種類十分接近，代表第 4 個 season 很有可能也是同一個海岸地形，這其實蠻合理的，畢竟同一個地點的海岸地形通常不會因為 season 的不同而變化，不過同一個地點在不同 season 時，某些海岸地形會略有不同，所以我們最後決定以同一個地點的其他 3 個 season 的眾數來填補缺失值

Station	Season	County	Location	Lat	Lon	縣市	海岸段	Region	Seat	Shore shap	Substrate ty	1暴露岩岸	2暴露人造	3暴露岩盤	4沙灘	5砂礫混合	6礫石灘	7開闊潮間	8遮蔽岩岸	9遮蔽潮間	10遮蔽濕
E02	1	宜蘭縣	大溪	24.92528	121.8857	16	5	1	4	2	3	0	0	1	0	0	0	1	0	0	0
E02	2	宜蘭縣	大溪	24.92528	121.8857	16	5	1	4	2	3	0	0	1	0	0	0	0	0	0	0
E02	3	宜蘭縣	大溪	24.92528	121.8857	16	5	1	4	2	3	0	0	1	0	0	0	0	0	0	0
E02	4	宜蘭縣	大溪	24.92528	121.8857	16	5	1	4	2	3										
E03	1	宜蘭縣	頭城	24.8573	121.8334	16	5	1	4	1	4	0	0	0	1	0	0	0	0	0	0
E03	2	宜蘭縣	頭城	24.8573	121.8334	16	5	1	4	1	4	0	0	0	1	0	0	0	0	0	0
E03	3	宜蘭縣	頭城	24.8573	121.8334	16	5	1	4	1	4	0	0	0	1	0	0	0	0	0	0
E03	4	宜蘭縣	頭城	24.8573	121.8334	16	5	1	4	1	4										

5.1.2 將可能會誤導 model 的 feature 去掉

根據觀察，同一個地點的海廢等級通常不會差異太大，所以地點可能會是一個重要因素，以經緯度來表示地點應該是最適當的，但是資料中除了經緯度還給了 Station、County、Location、縣市、海岸段、Region，這些 feature 我們目前是先轉成代號再丟到 model 裡去 train，但是這些代號可能會誤導 model，舉例來說，有可能 2 個地點的 Station 的代號差異很大，但是其實這 2 個地點並沒有差很遠，而且這樣也不好做 data augmention，所以可能會試著把除了經緯度以外代表地點的 feature 去掉

5.1.3 feature 的特徵值為代號

這些 train data 的 feature 除了經緯度以外，所有特徵值都是代號，這種類型的特徵值可能和一般的數值型特徵值處理方式略不同，無法和圖像 pixel 一樣直接丟到 model 去學

5.1.4 改變 normalize 的方式

原本是用 z-score standardization，但是 z-score standardization 會將分布會強行轉成標準高斯分布，原本的分布可能會跑掉，所以當原本的分布很不對稱時會出問題，所以未來可以試試看用 Min-Max Normalization，公式如下圖所示

最小值最大值正規化的用意，是將資料等比例縮放到 [0, 1] 區間中，可利用下列公式進行轉換：

$$X_{nom} = \frac{X - X_{min}}{X_{max} - X_{min}} \in [0, 1]$$

5.1.5 找出和海廢等級比較有關係的 feature

海廢等級代表海岸的汙染程度，海廢等級越高就代表海岸越髒，因此海廢等級和垃圾量有關，而垃圾量和人類活動有關，而人類活動可能和地點、季節、地形(ex: 沙灘)、垃圾類型這些 feature 有關，所以我們可以去做這個 feature 和海廢等級的關係圖、correlation 或是 p-value 來找出關鍵 feature 並增加權重

5.1.6 做 data augmentation

等做好 feature selection 找到關鍵性的 feature 後，就可以去做 data augmentation 來增加 train data 數，以避免過擬合問題

5.2 改良 model

5.2.1 用 LSTM

或許可以換成用 LSTM 之類考慮時序關係的 model 架構

5.2.2 ensemble learning

如果未來可以用多個不同的 model 得到不錯的結果，就可以拿這些 model 去做 ensemble learning，也就是將所以 model 的 prob array 取平均來得到最後的 prob array 和預測結果

-References

1. PATTERN RECOGNITION AND MACHINE LEARNING
2. PYTHON 機器學習與深度學習特訓班
3. Github
4. [資料分析&機器學習] 第 4.1 講 ：Kaggle 競賽-鐵達尼號生存預測-(前 16% 排名)
   https://medium.com/jameslearningnote/%E8%B3%87%E6%96%99%E5%88%86%E6%9E%90-%E6%A9%9F%E5%99%A8%E5%AD%B8%E7%BF%92-%E7%AC%AC4-1%E8%AC%9B-kaggle%E7%AB%B6%E8%B3%BD-%E9%90%B5%E9%81%94%E5%B0%BC%E8%99%9F%E7%94%9F%E5%AD%98%E9%A0%90%E6%B8%AC-%E5%89%8D16-%E6%8E%92%E5%90%8D-a8842fea7077
5. 我如何分析客戶流失預測？Kaggle 比賽思路分享
   https://medium.com/finformation%E7%95%B6%E7%A8%8B%E5%BC%8F%E9%81%87%E4%B8%8A%E8%B2%A1%E5%8B%99%E9%87%91%E8%9E%8D/%E6%88%91%E5%A6%82%E4%BD%95%E5%88%86%E6%9E%90%E5%AE%A2%E6%88%B6%E6%B5%81%E5%A4%B1%E9%A0%90%E6%B8%AC-kaggle%E6%AF%94%E8%B3%BD%E6%80%9D%E8%B7%AF%E5%88%86%E4%BA

%AB-daecd888a91

6. [資料分析&機器學習] 第 3.4 講：支援向量機(Support Vector Machine)介紹
   https://medium.com/jameslearningnote/%E8%B3%87%E6%96%99%E5%88%86%E6%9E%90-%E6%A9%9F%E5%99%A8%E5%AD%B8%E7%BF%92-%E7%AC%AC3-4%E8%AC%9B-%E6%94%AF%E6%8F%B4%E5%90%91%E9%87%8F%E6%A9%9F-support-vector-machine-%E4%BB%8B%E7%B4%B9-9c6c6925856b
7. [機器學習 ML NOTES]Kaggle 比賽心得(2%經歷)
   https://medium.com/@super135799/%E6%A9%9F%E5%99%A8%E5%AD%B8%E7%BF%92-ml-notes-kaggle%E6%AF%94%E8%B3%BD%E5%BF%83%E5%BE%97-2-%E7%B6%93%E6%AD%B7-7e8667cf1dc6
8. Feature Engineering 特徵工程中常見的方法
   https://vinta.ws/code/feature-engineering.html
9. Basic feature analysis (Date+Categorical+Revenue)
   https://www.kaggle.com/super13579/basic-feature-analysis-date-categorical-revenue
10. R 筆記 – Ensemble Learning(集成學習)
    https://rpubs.com/skydome20/R-Note16-Ensemble_Learning
11. 機器學習模型的時間複雜度
    https://kknews.cc/zh-tw/code/zyv254a.html
12. 機器學習: Ensemble learning 之 Bagging、Boosting 和 AdaBoost
    https://medium.com/@chih.sheng.huang821/%E6%A9%9F%E5%99%A8%E5%AD%B8%E7%BF%92-ensemble-learning%E4%B9%8Bbagging-boosting%E5%92%8Cadaboost-af031229ebc3
13. 隨機森林 RF 算法的原理（一）
    https://www.twblogs.net/a/5c8a02b2bd9eee35cd6a97fc
14. 隨機森林(RANDOM FOREST)的底層概念、操作細節，與推薦相關資源
    http://notebookpage1005.blogspot.com/2018/03/random-forest.html
15. 隨機森林（Random forest,RF）的生成方法以及優缺點
    https://www.itread01.com/content/1547100921.html