一、介紹

作業一將透過六份來自 Kaggle 與 UCI 網站的指定資料，建置監督式學習與半監督式學習的一般化分類模型，監督式學習會將資料分割成 65%訓練集、10% 驗證集與 25%測試集，半監督式學習會從資料中選取 50 筆作為訓練集，其餘為測試集。最後使用 AUC 和 Accuracy 來進行模型評估。

二、資料探索

	#Features	#Categorical	#Continuous	Size	#Pos	#Neg	% of Pos
Income	14	8	6	32561	7841	24720	24.0
Arcene	10000	0	10000	200	88	112	44.0
Bank	16	9	7	45211	5289	39922	11.7
BlastChar	20	17	3	7043	1869	5174	26.5
Shoppers	17	2	15	12330	1908	10442	15.5
Shrutime	11	3	8	10000	2037	7963	20.4

▲ 表一　資料基本描述

表一為各指定資料的基本描述，可以看到除了 ARCENE 資料，其他資料的目標變數都相當不平衡，後面預計使用 SMOTE 方法來彌補不足的樣本。

三、資料前處理

因為無法單單對某資料進行處理，故不將變數進行繪圖等分析，前處理將進行以下的步驟：

(1) 將所有目標變數轉換為 0 和 1
(2) 過濾多於 100 個種類的類別變數
(3) 對類別變數進行標籤編碼 (Label encoding)
(4) 對數值變數進行標準化
(5) 使用 SMOTE 方法來克服目標變數不平衡問題

四、建模

從資料探索知道，指定資料都有目標變數不平衡的問題，以下將對使用 SMOTE 處理前後進行比較。模型上並沒有特別參數調整，在最後一欄選擇 Voting Classifier，匯集前面的 Logistic Regression, Random Forest, XGBoost, CatBoost, LightGBM 五個模型，並採用概率制度的 Soft Voting 進行預測。

接下來的四個頁面為下列四種方法的 AUC 與 Accuracy 結果：

- 監督式學習 Supervised (without SMOTE)
- 半監督式學習 Semi-Supervised (without SMOTE)
- 監督式學習 Supervised (with SMOTE)
- 半監督式學習 Semi-Supervised (with SMOTE)

- 監督式學習 Supervised (without SMOTE)

AUC						
	Income	Arcene	Bank	BlastChar	Shopper	Shrutime
KNN	0.86	0.88	0.82	0.78	0.82	0.80
Logistic Reg.	0.91	0.92	0.93	0.84	0.91	0.75
Random Forest	0.90	0.90	0.94	0.83	0.94	0.84
MLP	0.87	0.85	0.92	0.83	0.92	0.76
SVM	0.90	0.92	0.89	0.84	0.89	0.73
XGBoost	0.91	0.84	0.94	0.80	0.93	0.82
CatBoost	0.93	0.88	0.95	0.85	0.94	0.86
LightGBM	0.91	0.81	0.94	0.84	0.95	0.86
VotingClassifier	0.92	0.89	0.95	0.84	0.94	0.85

Accuracy						
	Income	Arcene	Bank	BlastChar	Shopper	Shrutime
KNN	0.83	0.77	0.94	0.77	0.89	0.84
Logistic Reg.	0.85	0.81	0.94	0.80	0.90	0.81
Random Forest	0.85	0.82	0.94	0.79	0.91	0.86
MLP	0.82	0.74	0.93	0.76	0.89	0.80
SVM	0.85	0.84	0.94	0.80	0.91	0.80
XGBoost	0.86	0.76	0.94	0.77	0.90	0.84
CatBoost	0.87	0.73	0.95	0.80	0.91	0.86
LightGBM	0.86	0.71	0.95	0.80	0.91	0.86
VotingClassifier	0.92	0.89	0.95	0.84	0.94	0.85

- 半監督式學習 Semi-Supervised (without SMOTE)

AUC						
	Income	Arcene	Bank	BlastChar	Shopper	Shrutime
KNN	0.79	0.76	0.64	0.75	0.66	0.62
Logistic Reg.	0.84	0.83	0.78	0.77	0.78	0.64
Random Forest	0.82	0.77	0.74	0.78	0.83	0.68
MLP	0.83	0.82	0.74	0.74	0.75	0.64
SVM	0.80	0.84	0.74	0.73	0.78	0.65
XGBoost	0.91	0.84	0.94	0.80	0.93	0.82
CatBoost	0.82	0.74	0.77	0.77	0.86	0.66
LightGBM	0.78	0.70	0.68	0.76	0.61	0.64
VotingClassifier	0.82	0.76	0.75	0.78	0.83	0.70

Accuracy						
	Income	Arcene	Bank	BlastChar	Shopper	Shrutime
KNN	0.80	0.69	0.88	0.73	0.85	0.77
Logistic Reg.	0.80	0.76	0.88	0.74	0.86	0.77
Random Forest	0.80	0.71	0.88	0.76	0.86	0.80
MLP	0.81	0.73	0.88	0.72	0.86	0.73
SVM	0.79	0.77	0.85	0.71	0.85	0.76
XGBoost	0.78	0.64	0.86	0.72	0.88	0.74
CatBoost	0.79	0.70	0.88	0.75	0.88	0.79
LightGBM	0.79	0.65	0.86	0.75	0.82	0.76
VotingClassifier	0.79	0.71	0.88	0.76	0.85	0.80

- 監督式學習 Supervised (with SMOTE)

AUC						
	Income	Arcene	Bank	BlastChar	Shopper	Shrutime
KNN	0.93	0.83	0.98	0.86	0.96	0.85
Logistic Reg.	0.94	0.92	0.98	0.91	0.96	0.86
Random Forest	0.96	0.91	1.00	0.92	0.99	0.95
MLP	0.94	0.92	0.99	0.90	0.99	0.90
SVM	0.93	0.92	0.94	0.91	0.95	0.87
XGBoost	0.97	0.89	1.00	0.92	0.99	0.96
CatBoost	0.97	0.90	1.00	0.91	0.99	0.95
LightGBM	0.93	0.86	0.97	0.87	0.98	0.90
VotingClassifier	0.96	0.91	1.00	0.92	0.99	0.95

Accuracy						
	Income	Arcene	Bank	BlastChar	Shopper	Shrutime
KNN	0.86	0.74	0.95	0.78	0.89	0.72
Logistic Reg.	0.86	0.84	0.93	0.82	0.89	0.78
Random Forest	0.90	0.82	0.97	0.84	0.95	0.87
MLP	0.88	0.83	0.97	0.54	0.95	0.78
SVM	0.85	0.85	0.87	0.82	0.90	0.79
XGBoost	0.90	0.73	0.97	0.84	0.95	0.89
CatBoost	0.90	0.79	0.97	0.82	0.95	0.89
LightGBM	0.84	0.77	0.91	0.79	0.92	0.81
VotingClassifier	0.90	0.78	0.97	0.84	0.95	0.86

- 半監督式學習 Semi-Supervised (with SMOTE)

AUC						
	Income	Arcene	Bank	BlastChar	Shopper	Shrutime
KNN	0.77	0.81	0.76	0.74	0.71	0.66
Logistic Reg.	0.87	0.86	0.88	0.81	0.82	0.73
Random Forest	0.87	0.82	0.87	0.81	0.89	0.75
MLP	0.85	0.83	0.87	0.79	0.81	0.71
SVM	0.83	0.86	0.86	0.80	0.82	0.72
XGBoost	0.82	0.75	0.83	0.79	0.88	0.72
CatBoost	0.86	0.83	0.87	0.81	0.90	0.74
LightGBM	0.81	0.73	0.79	0.79	0.88	0.73
VotingClassifier	0.87	0.83	0.86	0.81	0.89	0.75

Accuracy						
	Income	Arcene	Bank	BlastChar	Shopper	Shrutime
KNN	0.71	0.74	0.69	0.70	0.66	0.62
Logistic Reg.	0.78	0.78	0.79	0.74	0.74	0.67
Random Forest	0.78	0.74	0.78	0.75	0.82	0.67
MLP	0.77	0.74	0.79	0.73	0.73	0.65
SVM	0.75	0.78	0.78	0.73	0.74	0.66
XGBoost	0.74	0.68	0.76	0.71	0.84	0.66
CatBoost	0.77	0.73	0.78	0.74	0.85	0.68
LightGBM	0.73	0.66	0.72	0.72	0.85	0.67
VotingClassifier	0.78	0.74	0.78	0.75	0.82	0.68

五、發現、難處與心得

從上方幾張結果表格比較監督式與半監督式的差異，可以看到監督式學習表現是比較好的，畢竟也訓練了較多的樣本，所以出現此結果不是太意外。但如果比較使用 SMOTE 前後的差異，發現在監督式學習下，都可見 AUC 和 Accuracy 提升，但在半監督式學習下卻是下降的。我猜想是沒有 SMOTE 的半監督式學習有嚴重的過度擬合，在只取 50 個樣本作為訓練集狀況下，有很大的機率過半數都是負樣本，而導致預測時正樣本無法分類出來，所以當 SMOTE 下去平衡後，正確率就下降了。結果來說，使用 SMOTE 方法下去平衡正負樣本依然是好的選擇。

表現最佳的模型通常是 Random Forest、XG Boost 和 Cat Boost，Voting Classifier 裡包含了這三個模型，最後結果看起來也很不錯，但同時也因為受限於這幾個模型的預測能力，所以有時候無法打敗所有的模型，但距離最佳的模型。如果使用 Grid Search CV 等方法下去調參，提升模型的預測能力再進行投票，也許就能更順利地打敗其他模型。

距離上次使用這些模型好像有一段時間了，這次作業真的有喚醒很多記憶，也把過去沒有努力學習的 Boost 模型認真看過，受歡迎的 Boost 模型都是確實都得到較高的分數。尋找一般化模型其實不簡單，我認為是因為每份資料都有在類別變數和數值變數的數量不同，像是 Arcene 資料變數很多且不包含類別變數，加上無法針對一份資料下去調整模型，資料前處理就變得更重要，但建置一般化模型也限制了個別資料的前處理，後面導致有些模型預測時不太順利，像 SVM 在資料二、資料四上面就執行很久覺得很折磨。困難大概是資料前處理不是很滿意，對於要全部統一處理並不是很有想法，資料往下若進行變數篩選並且配合模型調參，應該能出現更好的結果。希望可以透過這學期的課程，來更熟悉資料分析，累積更多實作經驗。