重複購買的意願預測

成員：S09350204/廖書賢/s09350204@thu.edu.tw

一、視覺化

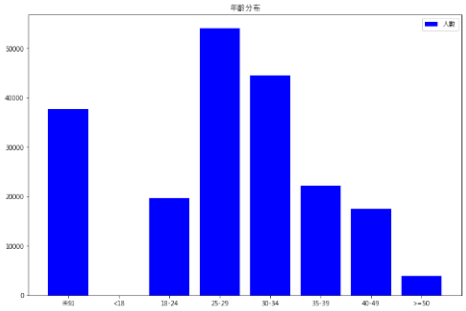


圖 1-1 用戶年齡分布

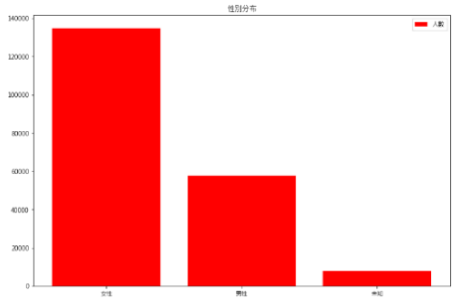


圖 1-2 用戶性別分布

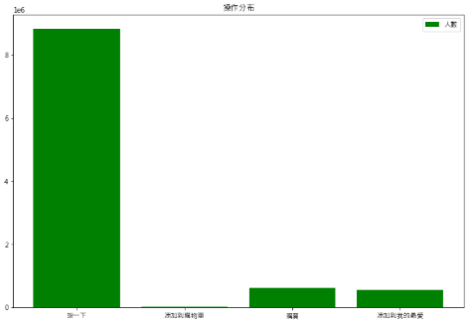


圖 1-3 用戶操作分布

根據圖 1-1~圖 1-3，可以得出以下結論

1. 用戶主要集中的年齡在 25-29 歲。
2. 女性用戶占大多數。
3. 大多數用戶的操作都是點擊，添加到購物車的操作次數是最低。
4. 年齡的未知比性別的未知多。

二、特徵建立

user_id	merchant_id	label	data	age_range	gender	uniq_item_id	total_cat_id	total_time_temp	clicks	shopping_cart	purchases	favourites	purchases_ctr
231552	3828	1.0	train	5.0	0.0	48	15	3	78	0	5	0	0.064103
231552	2124	0.0	train	5.0	0.0	4	1	1	6	0	1	0	0.166667
298368	2981	0.0	train	4.0	0.0	2	1	3	4	0	1	0	0.250000
36480	4730	0.0	train	6.0	1.0	1	1	1	2	0	1	0	0.500000
105600	1487	0.0	train	6.0	1.0	8	1	3	20	0	1	0	0.050000

2.1. 影響重複購買的因素

性別和年齡、口碑、用戶喜好和賣家商品的相似性。

2.2. age_range, gender: 性別和年齡

2.3. 用戶-賣家建立以下特徵

1. uniq_item_id: 交互過的商品
2. total_cat_id: 交互過的品類
3. Total_time_temp: 交互天數
4. clicks: 點擊的操作次數
5. shopping_cart: 添加購物車的操作次數
6. purchases: 購買的操作次數
7. favourites: 添加到我的最愛的操作次數
8. purchases_ctr: 購買點擊率(口碑)

三、模型

3.1. 正樣本比例

```
訓練集正樣本比例:  0.06406670808792114
驗證集正樣本比例:  0.06585057079792023
```

訓練集和驗證集的正樣本比例基本上一樣。

3.2. GridSearchCV，選擇模型最佳的參數

參數的設定參考各個有關模型參數的網站(網站在 找出最佳模型參數.ipynb 的參考資料附)。

選擇參數的數值的方式: 自己先測試幾個常用的數值，並依據測試出的最佳參數做調整，例如:邏輯迴歸中，參數 C 原本設定[0.1, 1, 10, 100]，最佳參數 C 得出是 0.1，這時候，我就想說參數 C 是不是越小，就是最佳參數，所以我就設定[0.01, 1, 10]，但得出結果還是 0.1。

下面參數我就依照這樣的方式去做設定，有些的確是有得到更好的參數，但大多數都像參數 C 的狀況一樣。

3.2.1. RandomForest

1. 參數設定

```
GridSearchCV(cv=3, estimator=RandomForestClassifier(),
             param_grid={'max_depth': [1, 5, 10],
                         'min_samples_leaf': [1, 10, 50],
                         'min_samples_split': [1, 2, 100],
                         'n_estimators': [50, 100]},
             scoring='roc_auc')
```

2. 最佳參數

```
{'max_depth': 5,
 'min_samples_leaf': 50,
 'min_samples_split': 2,
 'n_estimators': 50}
```

3.2.2. LogisticRegression

1. 參數設定

```
GridSearchCV(cv=5, estimator=LogisticRegression(),
             param_grid={'C': [0.01, 0.1, 1], 'penalty': ['l1', 'l2'],
                         'solver': ['liblinear', 'saga', 'lbfgs', 'newton-cg']},
             scoring='roc_auc')
```

2. 最佳參數

```
{'C': 0.1, 'penalty': 'l1', 'solver': 'liblinear'}
```

3.2.3. XGBoost

1. 參數設定

```
GridSearchCV(cv=3,
             estimator=XGBClassifier(base_score=None, booster=None,
                                     callbacks=None, colsample_bylevel=None,
                                     colsample_bynode=None,
                                     colsample_bytree=None,
                                     early_stopping_rounds=None,
                                     enable_categorical=False, eval_metric=None,
                                     gamma=None, gpu_id=None, grow_policy=None,
                                     importance_type=None,
                                     interaction_constraints=None,
                                     learning_rate=None, max_bin=None,
                                     max_ca...
                                     max_leaves=None, min_child_weight=None,
                                     missing=nan, monotone_constraints=None,
                                     n_estimators=100, n_jobs=None,
                                     num_parallel_tree=None, predictor=None,
                                     random_state=None, reg_alpha=None,
                                     reg_lambda=None, ...),
             param_grid={'eta': [0.1, 0.2], 'eval_metric': ['auc'],
                         'gamma': [1, 5, 50], 'max_depth': [1, 5, 50],
                         'min_child_weight': [10, 100, 500],
                         'objective': ['binary:logistic'], 'subsample': [0.5]},
             scoring='roc_auc')
```

2. 最佳參數

```
{'eta': 0.1,
 'eval_metric': 'auc',
 'gamma': 1,
 'max_depth': 1,
 'min_child_weight': 10,
 'objective': 'binary:logistic',
 'subsample': 0.5}
```

3.3.4. LightGBM

1. 參數設定

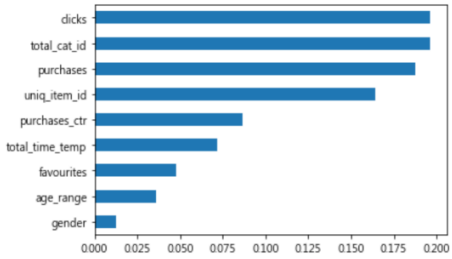
```
GridSearchCV(cv=3, estimator=LGBMClassifier(),
             param_grid={'boosting_type': ['gbdt', 'dart', 'goss'],
                         'learning_rate': [0.01, 0.05],
                         'max_depth': [50, 60, 70], 'min_split_gain': [0.05],
                         'n_estimators': [100, 500],
                         'num_leaves': [10, 30, 100], 'subsample': [0.5]},
             scoring='roc_auc')
```

2. 最佳參數

```
{'boosting_type': 'goss',
 'learning_rate': 0.05,
 'max_depth': 50,
 'min_split_gain': 0.05,
 'n_estimators': 100,
 'num_leaves': 10,
 'subsample': 0.5}
```

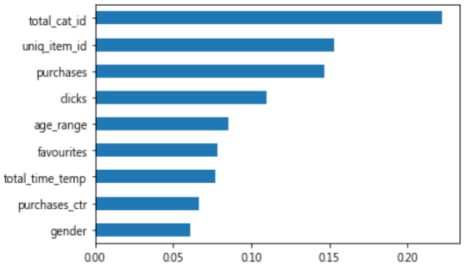
3.3. 各個模型 Top 10 features

3.3.1 RandomForest



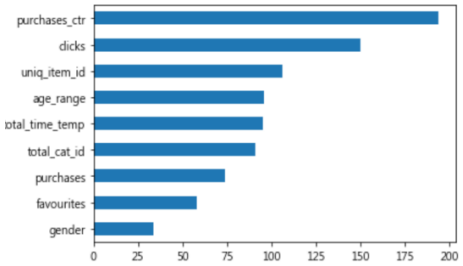
排名前三的重要特徵: 用戶-賣家點擊的操作次數、用戶-賣家交互過的品類數量、用戶-賣家購買的操作次數。

3.3.2. XGBoost



排名前三的重要特徵: 用戶-賣家交互過的商品品類數量、用戶-賣家交互過的商品數量、用戶-賣家購買的操作次數。

3.3.3. LightGBM



排名前三的重要特徵: 用戶-賣家購買點擊率、用戶-賣家點擊的操作次數、用戶-賣家交互過的商品數量。

3.3. 比較各個模型的準確率(準確率: roc_auc_score)

3.3.1. 比較沒調過任何模型參數

	model	auc
0	LogisticRegression	0.629195
3	LightGBM	0.603576
1	RandomForest	0.580038
2	XGBoost	0.565893

auc 分數: LogisticRegression > LightGBM > RandomForest > XGBoost

3.3.2. 比較調過最佳的模型參數

	model	auc
2	XGBoost	0.635106
0	LogisticRegression	0.631018
3	LightGBM	0.626414
1	RandomForest	0.625070

auc 分數: XGBoost > LogisticRegression > LightGBM > RandomForest

1. 通過比較各個模型的 auc 分數，最終選擇 XGBoost 為最佳預測模型，auc 分數為 0.635106。在重要特徵排名中，用戶-賣家交互過的商品品類數量、用戶-賣家交互過的商品數量、用戶-賣家購買的操作次數，對模型影響最大。

2. 最佳模型的預測結果儲存到 csv (result.csv)。

三、結論

3.1. 針對那些新消費者在未來可能成為忠實顧客

根據視覺化和最佳模型的特徵，得出以下幾點

1. 用戶集中在 25-29 歲和女性。
2. 影響重複購買的因素有商品品類數量、商品數量、購買的操作次數。

符合以上兩點消費者，在未來最有可能為忠實顧客。

3.2. 預測消費者再次購買的機率

根據最佳模型的預測結果，消費者會再次上門購買的機率約為 63%。

四、心得

這次的實作過程中，遇到很多困難點，例: 建立特徵的時候，要想什麼樣的因素會影響到重複購買、決定選擇什麼迴歸模型、調整模型的參數等。遇到這些的困難點的時候，我非常地開心，因為表示還可以讓這個專題變得更好，當解決這些問題也非常有成就感。未來的目標希望可以把這個專題變得有參考價值，我覺得還有很多問題沒有考量進去，而且這個題目應該是一個跨領域，所以要找不同的科系一起研究這個專題，會變得更有參考性。