2023年5月4日 星期四

Logistic Regression

Ranking Card

- Dataset：rankingcard.csv

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 149391 entries, 0 to 149390
Data columns (total 11 columns):
 #   Column                                Non-Null Count   Dtype  
---  ------                                --------------   -----  
 0   SeriousDlqin2yrs                      149391 non-null  int64  
 1   RevolvingUtilizationOfUnsecuredLines  149391 non-null  float64
 2   age                                   149391 non-null  int64  
 3   NumberOfTime30-59DaysPastDueNotWorse  149391 non-null  int64  
 4   DebtRatio                             149391 non-null  float64
 5   MonthlyIncome                         120170 non-null  float64
 6   NumberOfOpenCreditLinesAndLoans       149391 non-null  int64  
 7   NumberOfTimes90DaysLate               149391 non-null  int64  
 8   NumberRealEstateLoansOrLines          149391 non-null  int64  
 9   NumberOfTime60-89DaysPastDueNotWorse  149391 non-null  int64  
 10  NumberOfDependents                    145563 non-null  float64
dtypes: float64(4), int64(7)
memory usage: 12.5 MB
```

- Data Preprocessing：

    1. 刪除重複值
    2. 填補缺失值
        1. NumberOfDependents 用平均值填補
        2. MonthlyIncome 用隨機回歸森林填補
    3. 利用描述性統計處理異常值
        1. 處理 age 異常值
        2. 刪除異常違約值

Before

	count	mean	std	min	1%	10%	25%	50%	75%	90%	99%	max
SeriousDlqin2yrs	149391.0	0.066999	0.250021	0.0	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	1.0
lvingUtilizationOfUnsecuredLines	149391.0	6.071087	250.263672	0.0	0.0	0.003199	0.030132	0.154235	0.556494	0.978007	1.093922	50708.0
age	149391.0	52.306237	14.725962	0.0	24.0	33.000000	41.000000	52.000000	63.000000	72.000000	87.000000	109.0
NumberOfTime30-59DaysPastDueNotWorse	149391.0	0.393886	3.852953	0.0	0.0	0.000000	0.000000	0.000000	0.000000	1.000000	4.000000	98.0
DebtRatio	149391.0	354.436740	2041.843455	0.0	0.0	0.034991	0.177441	0.368234	0.875279	1275.000000	4985.100000	329664.0
MonthlyIncome	149391.0	5425.628057	13252.745101	0.0	0.0	0.180000	1800.000000	4420.000000	7416.000000	10800.000000	23205.000000	3008750.0
mberOfOpenCreditLinesAndLoans	149391.0	8.480892	5.136515	0.0	0.0	3.000000	5.000000	8.000000	11.000000	15.000000	24.000000	58.0
NumberOfTimes90DaysLate	149391.0	0.238120	3.826165	0.0	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	3.000000	98.0
NumberRealEstateLoansOrLines	149391.0	1.022391	1.130196	0.0	0.0	0.000000	0.000000	1.000000	2.000000	2.000000	4.000000	54.0
NumberOfTime60-89DaysPastDueNotWorse	149391.0	0.212503	3.810523	0.0	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	2.000000	98.0
NumberOfDependents	149391.0	0.740393	1.108272	0.0	0.0	0.000000	0.000000	0.000000	1.000000	2.000000	4.000000	20.0

After

	count	mean	std	min	1%	10%	25%	50%	75%	90%	99%	max
SeriousDlqin2yrs	149165.0	0.066188	0.248612	0.0	0.0	0.000000	0.000000	0.000000	0.000000	0.00000	1.000000	1.0
olvingUtilizationOfUnsecuredLines	149165.0	6.078770	250.453111	0.0	0.0	0.003174	0.030033	0.153615	0.553698	0.97502	1.094061	50708.0
age	149165.0	52.331076	14.714114	21.0	24.0	33.000000	41.000000	52.000000	63.000000	72.00000	87.000000	109.0
NumberOfTime30-59DaysPastDueNotWorse	149165.0	0.246720	0.698935	0.0	0.0	0.000000	0.000000	0.000000	0.000000	1.00000	3.000000	13.0
DebtRatio	149165.0	354.963542	2043.344496	0.0	0.0	0.036385	0.178211	0.368619	0.876994	1277.30000	4989.360000	329664.0
MonthlyIncome	149165.0	5429.669800	13261.846549	0.0	0.0	0.180000	1800.000000	4433.000000	7418.000000	10800.00000	23250.000000	3008750.0
mberOfOpenCreditLinesAndLoans	149165.0	8.493688	5.129841	0.0	1.0	3.000000	5.000000	8.000000	11.000000	15.00000	24.000000	58.0
NumberOfTimes90DaysLate	149165.0	0.090725	0.486354	0.0	0.0	0.000000	0.000000	0.000000	0.000000	0.00000	2.000000	17.0
NumberRealEstateLoansOrLines	149165.0	1.023927	1.130350	0.0	0.0	0.000000	0.000000	1.000000	2.000000	2.00000	4.000000	54.0
NumberOfTime60-89DaysPastDueNotWorse	149165.0	0.065069	0.330675	0.0	0.0	0.000000	0.000000	0.000000	0.000000	0.00000	2.000000	11.0
NumberOfDependents	149165.0	0.740911	1.108534	0.0	0.0	0.000000	0.000000	0.000000	1.000000	2.00000	4.000000	20.0

4. 處理樣本不均勻的問題 (違約總是少數)

```
#處理樣本不均勻的問題 => 目標是改變整體樣本數來達成樣本數均勻
#imblearn 專門用來處理不平衡之數據集，相較於 sklearn 快

import imblearn
from imblearn.over_sampling import SMOTE

sm = SMOTE(random_state=42)
X,y = sm.fit_resample(X,y)

n_oversample = X.shape[0]

pd.Series(y).value_counts()

n_1_oversample = pd.Series(y).value_counts()[1]
n_0_oversample = pd.Series(y).value_counts()[0]

print("樣本數: {}; 1: {:.2%}; 0: {:.2%}".format(n_oversample,n_1_oversample/n_oversample,n_0_oversample/n_oversample))
```

```
樣本數: 278584; 1: 50.00%; 0: 50.00%
```

- 分箱
  - 意義：將不同的屬性的人分成不同類別 (離散化連續變量)
  - 模型：IV model 優勢為圖表可視化，方便入門理解
  - 步驟
    1. 將連續型變量分成數量較多的分類型變量
    2. 確保每組都要包含兩種類別的樣本，不然 IV 值無法計算
    3. 進行卡方檢驗，P值很大的組進行合併，直到組數小於 N
    4. 觀察 IV 值的變化，選出最佳箱數

- 分箱結果

  手動分箱：無法分成 20 組，因此自己觀察分組，並用正負無窮取代手動分箱的上下限，避免新資料進來導致錯誤。

```
auto_bins = {
    "RevolvingUtilizationOfUnsecuredLines":6,
    "age":5,
    "DebtRatio":4,
    "MonthlyIncome":5,
    "NumberOfOpenCreditLinesAndLoans":5
}

#無法自動最佳分箱
hand_bins ={
    "NumberOfTime30-59DaysPastDueNotWorse": [0,1,2,13],
    "NumberOfTimes90DaysLate": [0,1,2,17],
    "NumberRealEstateLoansOrLines": [0,1,2,4,54],
    "NumberOfTime60-89DaysPastDueNotWorse": [0,1,2,8],
    "NumberOfDependents":[0,1,2,3]
}
#用正負無窮取代手動分箱的上下限，避免新資料進來導致錯誤
hand_bins = {k:[-np.inf,*v[:-1],np.inf] for k,v in hand_bins.items()}
```

- 計算各箱的 WOE 並映射到數據中

	RevolvingUtilizationOfUnsecuredLines	age	DebtRatio	MonthlyIncome	NumberOfOpenCreditLinesAndLoans	NumberOfTime30-59DaysPastDueNotWorse	NumberOfTimes90DaysLate
0	2.205113	-0.278936	0.037491	-0.273465	-0.059444	0.353286	0.235255
1	0.667901	1.003828	0.037491	-0.273465	-0.059444	0.353286	0.235255
2	-2.037016	-0.278936	-0.389667	-0.273465	-0.059444	-0.875047	-1.754253
3	2.205113	1.003828	-0.389667	-0.273465	0.125124	0.353286	0.235255
4	-1.074589	-0.278936	-0.389667	0.350106	0.125124	0.353286	0.235255
...	...	...	...	...	...	...	...
195003	-1.074589	-0.520698	-0.389667	0.070519	0.125124	-1.374388	0.235255
195004	-1.074589	-0.278936	0.037491	-0.273465	0.125124	-0.875047	0.235255
195005	-1.074589	-0.278936	-0.389667	0.350106	0.125124	0.353286	0.235255
195006	-0.474468	1.003828	0.037491	0.350106	0.125124	0.353286	0.235255
195007	-1.074589	-0.278936	0.176333	0.155484	0.125124	0.353286	0.235255

195008 rows × 11 columns

- 模型訓練結果
  - 準確率：0.788623
  - ROC 曲線：0.87

    其中 AUC (Area Under Curve) 代表在ROC曲線底下的區域面積，也是大家常用的評估指標之一。ROC底下的面積越大越好，表示曲線更靠近左上方。