산탄데르 고객 만족 예측 - 분류

학습 내용

- LightGBM 모델을 활용한 예측

대회 설명

- URL :https://www.kaggle.com/competitions/santander-customer-satisfaction/overview (https://www.kaggle.com/competitions/santander-customer-satisfaction/overview)
- 어떤 고객이 행복한 고객입니까? 이를 예측하는 대회
- 평가지표 : AUC - ROC-AUC(ROC 곡선 영역)

데이터 설명

- 데이터 다운로드 : https://www.kaggle.com/c/santander-customer-satisfaction/data (https://www.kaggle.com/c/santander-customer-satisfaction/data)
- 370개의 피처로 이루어진 데이터
- 피처 이름은 전부 익명처리되어 있음.
- 클래스 레이블 명은 TARGET
  - 값이 1이면 불만을 가지고 있음.
  - 값이 0이면 만족한 고객

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib
```

In [2]:

```
train = pd.read_csv("../../dataset/santander_customer/train.csv", encoding='latin-1'
test = pd.read_csv("../../dataset/santander_customer/test.csv", encoding='latin-1')
sub = pd.read_csv("../../dataset/santander_customer/sample_submission.csv")

train.shape, test.shape, sub.shape
```

Out[2]:

```
((76020, 371), (75818, 370), (75818, 2))
```

In [3]:

```
## ID 제외한 열 선택
train = train.loc[  :, "var3":  ]
train.head()
```

Out[3]:

	var3	var15	imp_ent_var16_ult1	imp_op_var39_comer_ult1	imp_op_var39_comer_ult3	imp_op_v
0	2	23	0.0	0.0	0.0	
1	2	34	0.0	0.0	0.0	
2	2	23	0.0	0.0	0.0	
3	2	37	0.0	195.0	195.0	
4	2	39	0.0	0.0	0.0	

5 rows × 370 columns

In [4]:

```
# 피처와 레이블를 지정.(입력, 출력 나누기))
X = train.iloc[:, :-1]
y = train['TARGET']

from sklearn.model_selection import train_test_split

X_train , X_test, y_train, y_test = train_test_split(X, y,
                                                     test_size=0.2, random_state=0)

X_train.shape, X_test.shape, y_train.shape, y_test.shape
```

Out[4]:

```
((60816, 369), (15204, 369), (60816,), (15204,))
```

In [5]:

```
from lightgbm import LGBMClassifier
from sklearn.metrics import roc_auc_score
```

In [6]:

```
%%time

lgbm_model = LGBMClassifier(n_estimators = 500)
evals = [(X_test, y_test)]

lgbm_model.fit(X_train, y_train,
               early_stopping_rounds=100,
               eval_metric='auc',
               eval_set=evals,
               verbose=True)
```

```
[1]     valid_0's auc: 0.817384 valid_0's binary_logloss: 0.165046
Training until validation scores don't improve for 100 rounds
[2]     valid_0's auc: 0.818903 valid_0's binary_logloss: 0.160006
[3]     valid_0's auc: 0.827707 valid_0's binary_logloss: 0.156323
[4]     valid_0's auc: 0.832155 valid_0's binary_logloss: 0.153463
[5]     valid_0's auc: 0.834677 valid_0's binary_logloss: 0.151256
[6]     valid_0's auc: 0.83419  valid_0's binary_logloss: 0.149407
[7]     valid_0's auc: 0.837155 valid_0's binary_logloss: 0.147942
[8]     valid_0's auc: 0.837996 valid_0's binary_logloss: 0.146565
[9]     valid_0's auc: 0.839603 valid_0's binary_logloss: 0.145427
[10]    valid_0's auc: 0.839867 valid_0's binary_logloss: 0.14447
[11]    valid_0's auc: 0.839887 valid_0's binary_logloss: 0.14375
[12]    valid_0's auc: 0.839856 valid_0's binary_logloss: 0.143201
[13]    valid_0's auc: 0.839997 valid_0's binary_logloss: 0.142632
[14]    valid_0's auc: 0.840001 valid_0's binary_logloss: 0.142149
[15]    valid_0's auc: 0.84079  valid_0's binary_logloss: 0.14171
[16]    valid_0's auc: 0.840096 valid_0's binary_logloss: 0.141372
[17]    valid_0's auc: 0.839711 valid_0's binary_logloss: 0.141201
[18]    valid_0's auc: 0.839128 valid_0's binary_logloss: 0.141044
```

- 수행 시간이 상당히 줄어들었음. 2초

In [7]:

```
pred_prob = lgbm_model.predict_proba(X_test)[:, 1]
lgbm_roc_score = roc_auc_score(y_test, pred_prob, average='macro')
print("ROC AUC : {0:.4f}".format(lgbm_roc_score))
```

```
ROC AUC : 0.8408
```

하이퍼 파라미터 튜닝

In [8]:

```
%%time

from sklearn.model_selection import GridSearchCV

lgbm_model01 = LGBMClassifier(n_estimators = 500)

params = {"max_depth":[32, 64, 128, 160],
          "min_child_samples":[60, 100],
          "num_leaves":[32, 64],
          "subsample":[0.6, 0.8, 1]}

gridcv = GridSearchCV(lgbm_model01, param_grid=params, cv=3)

gridcv.fit(X_train, y_train,
           early_stopping_rounds=30,
           eval_metric='auc',
           eval_set = [(X_train, y_train), (X_test, y_test)])
```

```
[1]     valid_0's auc: 0.820235 valid_0's binary_logloss: 0.156085
valid_1's auc: 0.81613  valid_1's binary_logloss: 0.164992
Training until validation scores don't improve for 30 rounds
[2]     valid_0's auc: 0.825778 valid_0's binary_logloss: 0.150951
valid_1's auc: 0.821835 valid_1's binary_logloss: 0.159874
[3]     valid_0's auc: 0.832262 valid_0's binary_logloss: 0.147158
valid_1's auc: 0.826533 valid_1's binary_logloss: 0.156346
[4]     valid_0's auc: 0.83865  valid_0's binary_logloss: 0.144126
valid_1's auc: 0.833166 valid_1's binary_logloss: 0.1534
[5]     valid_0's auc: 0.842822 valid_0's binary_logloss: 0.141725
valid_1's auc: 0.836448 valid_1's binary_logloss: 0.151167
[6]     valid_0's auc: 0.844702 valid_0's binary_logloss: 0.139642
valid_1's auc: 0.837094 valid_1's binary_logloss: 0.149356
[7]     valid_0's auc: 0.847144 valid_0's binary_logloss: 0.13794
valid_1's auc: 0.837965 valid_1's binary_logloss: 0.147853
[8]     valid_0's auc: 0.848277 valid_0's binary_logloss: 0.136499
valid_1's auc: 0.837663 valid_1's binary_logloss: 0.146543
[9]     valid_0's auc: 0.849328 valid_0's binary_logloss: 0.135326
valid_1's auc: 0.837413 valid_1's binary_logloss: 0.145528
```

In [11]:

```
print("GridSearchCV  최적 파라미터 : ", gridcv.best_params_ )
```

```
GridSearchCV  최적 파라미터 :  {'max_depth': 32, 'min_child_samples': 60,
'num_leaves': 64, 'subsample': 0.6}
```

In [12]:

```
pred_prob = gridcv.predict_proba(X_test)[:, 1]
lgbm_roc_score = roc_auc_score(y_test, pred_prob, average='macro')
print("ROC AUC : {0:4f}".format(lgbm_roc_score))
```

```
ROC AUC : 0.841281
```

- 최적의 파라미터
  - 'max_depth': 32,

- 'min_child_samples': 60,
- 'num_leaves': 64,
- 'subsample': 0.6

최종 모델

In [13]:

```
%%time

lgbm_model_1 = LGBMClassifier(n_estimators=1000,
                              max_depth=32,
                              min_child_samples=60,
                              num_leaves=64,
                              subsamle=0.6)

evals = [(X_test, y_test)]
lgbm_model_1.fit(X_train, y_train, early_stopping_rounds=100,
                 eval_metric='auc', eval_set=evals,
                 verbose=True)
```

```
[LightGBM] [Warning] Unknown parameter: subsamle
[1]     valid_0's auc: 0.820192 valid_0's binary_logloss: 0.164812
Training until validation scores don't improve for 100 rounds
[2]     valid_0's auc: 0.826488 valid_0's binary_logloss: 0.159486
[3]     valid_0's auc: 0.833867 valid_0's binary_logloss: 0.155607
[4]     valid_0's auc: 0.835902 valid_0's binary_logloss: 0.15279
[5]     valid_0's auc: 0.837887 valid_0's binary_logloss: 0.150685
[6]     valid_0's auc: 0.83821  valid_0's binary_logloss: 0.148674
[7]     valid_0's auc: 0.838396 valid_0's binary_logloss: 0.147187
[8]     valid_0's auc: 0.839675 valid_0's binary_logloss: 0.145756
[9]     valid_0's auc: 0.839506 valid_0's binary_logloss: 0.144762
[10]    valid_0's auc: 0.839484 valid_0's binary_logloss: 0.143878
[11]    valid_0's auc: 0.839971 valid_0's binary_logloss: 0.143068
[12]    valid_0's auc: 0.84034  valid_0's binary_logloss: 0.142366
[13]    valid_0's auc: 0.840786 valid_0's binary_logloss: 0.141853
[14]    valid_0's auc: 0.840533 valid_0's binary_logloss: 0.14142
[15]    valid_0's auc: 0.839717 valid_0's binary_logloss: 0.141168
[16]    valid_0's auc: 0.840062 valid_0's binary_logloss: 0.140774
[17]    valid_0's auc: 0.839837 valid_0's binary_logloss: 0.140555
```

In [14]:

```
pred_prob = lgbm_model_1.predict_proba(X_test)[:, 1]
lgbm_roc_score = roc_auc_score(y_test, pred_prob, average='macro')
print("ROC AUC : {0:4f}".format(lgbm_roc_score))
```

```
ROC AUC : 0.841281
```