SOCAR

보험 사기 분류 모델링

(머신러닝 프로젝트, 3조)

김도겸　류승환　임현수

목차

0. 프로젝트 소개

1. 베이스라인 및 목표

2. 진행 과정

3. 결과 및 최종 선택 모델

4. 추후 개선점, 아쉬웠던 점, 의문

SOCAR

0. 프로젝트 소개

프로젝트 소개

1. 주제

: 머신러닝을 통해 자동차 보험 사기 여부 예측 모델 개발

2. 기대효과

: 보험 사기 여부 실사 시, 조사대상 건수 최소화를 통한 효율적 리소스 운영

1. 베이스라인 및 목표

베이스라인 설정

1. 기준

: recall 값 **0.29** 이상

2. 이유

: 기타 전처리 없이 오버샘플링(SMOTE)만 진행 시 결과값
→ 기타 전처리 추가에 따른 성능 향상 기대

```
============
name: LogisticRegression
accuracy_score: 0.7926946491509131
[[2472  642]
 [   5    2]]
              precision    recall  f1-score   support

           0       1.00      0.79      0.88      3114
           1       0.00      0.29      0.01         7

    accuracy                           0.79      3121
   macro avg       0.50      0.54      0.45      3121
weighted avg       1.00      0.79      0.88      3121

============
name: DecisionTree
accuracy_score: 0.6529958346683755
[[2036 1078]
 [   5    2]]
              precision    recall  f1-score   support

           0       1.00      0.65      0.79      3114
           1       0.00      0.29      0.00         7

    accuracy                           0.65      3121
   macro avg       0.50      0.47      0.40      3121
weighted avg       1.00      0.65      0.79      3121
```

목표

1. 주목표

: recall 최대화

2. 부목표

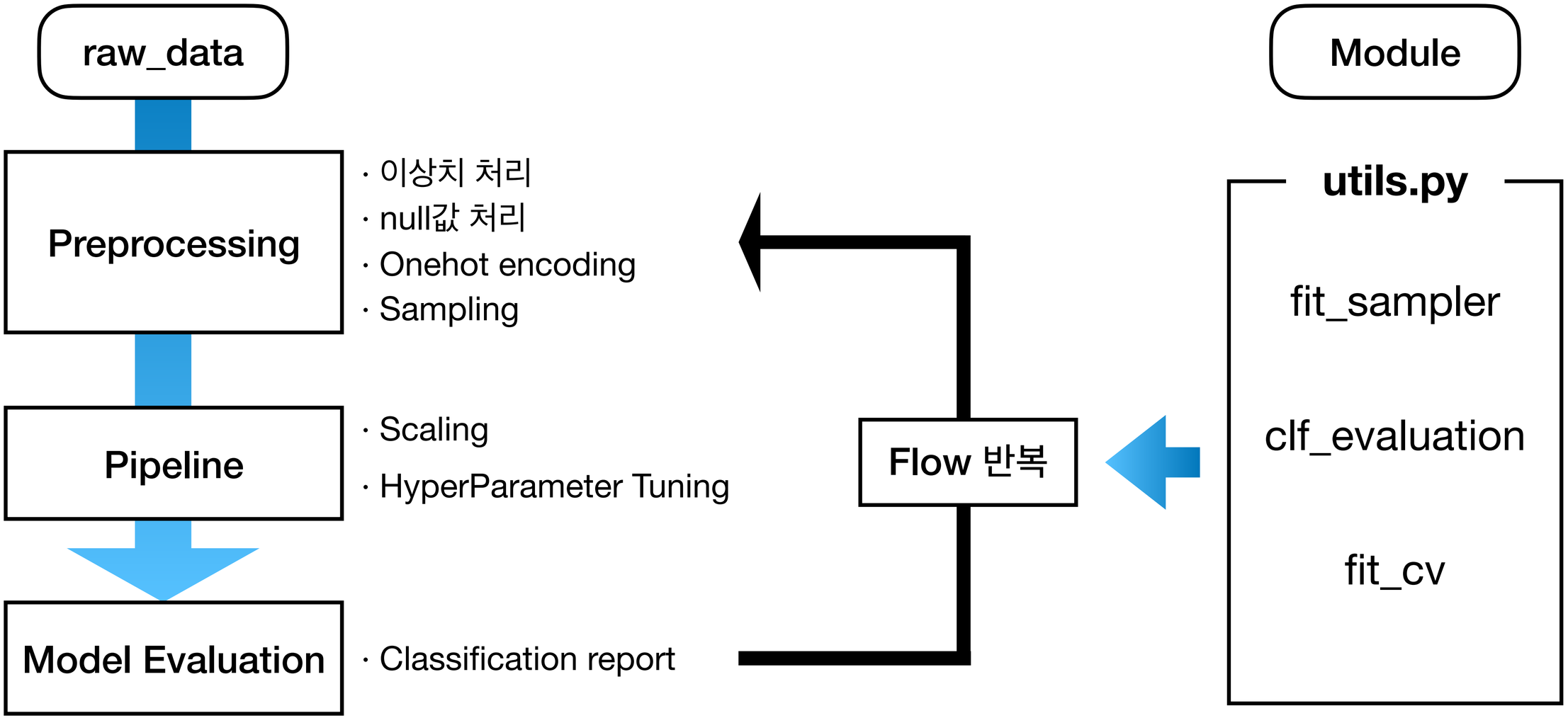
: recall값이 동일할 경우, accuracy 최대화

```
============
name: LogisticRegression
accuracy_score: 0.7926946491509131
[[2472  642]
 [   5    2]]
              precision    recall  f1-score   support

           0       1.00      0.79      0.88      3114
           1       0.00      0.29      0.01         7

    accuracy                           0.79      3121
   macro avg       0.50      0.54      0.45      3121
weighted avg       1.00      0.79      0.88      3121

============
name: DecisionTree
accuracy_score: 0.6529958346683755
[[2036 1078]
 [   5    2]]
              precision    recall  f1-score   support

           0       1.00      0.65      0.79      3114
           1       0.00      0.29      0.00         7

    accuracy                           0.65      3121
   macro avg       0.50      0.47      0.40      3121
weighted avg       1.00      0.65      0.79      3121
```

SOCAR

2. 진행 과정

순서도


raw_data
Preprocessing
· 이상치 처리
· null값 처리
· Onehot encoding
· Sampling
Pipeline
· Scaling
· HyperParameter Tuning
Model Evaluation
· Classification report
Flow 반복
Module
utils.py
fit_sampler
clf_evaluation
fit_cv

순서도

이상데이터 제거	null값 제거	OneHot Encoding	Sampling	Scaling	Hyper Parameter Tuning
1가지 (제거)	**2**가지 (제거/유지)	**2**가지 (적용/미적용)	**5**가지 · SMOTE · Borderline SMOTE · SVMSMOTE · ADASYN · RandomOver Sampling	**4**가지 · 미적용 · MinMax · Standard · Robust	**10**가지 · 5가지 알고리즘 · 2가지 경우 (class_weight tuning 여부)

총 **800**가지 경우에 대해 시행

전처리: 이상데이터 제거

대여시간과 사고발생시각 비교 시, 납득하기 어려운 데이터 존재

(ex. 11~20시 대여, 21시 이후 사고 발생)

	start_hour	accident_hour	gap_start_acci	duration	test_set
43	4	1	-3	2	0
1504	4	1	-3	1	0
1762	4	1	-3	2	0
1766	4	1	-3	2	1
11706	4	1	-3	2	0
12031	4	1	-3	2	0
12665	4	1	-3	2	0

(※ 추후 마스킹 예정)

→ 총 **41개** 데이터 발견, 삭제 후 진행

전처리: Null값 제거, Encoding, Sampling

Null값 제거

6개 features 제거

- repair_cost
- insure_cost
- acc_type1
- insurance_site_aid_YN
- police_site_aid_YN
- total_prsn_cnt

OneHot Encoding

9~13개 features 제거

(Null값 제거 시 9개, 포함 시 13개)

- car_model
- sharing_type
- age_group
- has_previous _accident
- …

Sampling

5가지 경우

- SMOTE
- BorderlineSMOTE
- SVMSMOTE
- ADASYN
- RandomOverSampling

(※ 추후 마스킹 예정)

Pipeline: Scaling, hyperparameter tuning

Scaling

4가지 경우

- 미적용
- MinMax
- Standard
- Robust

HyperParameter Tuning

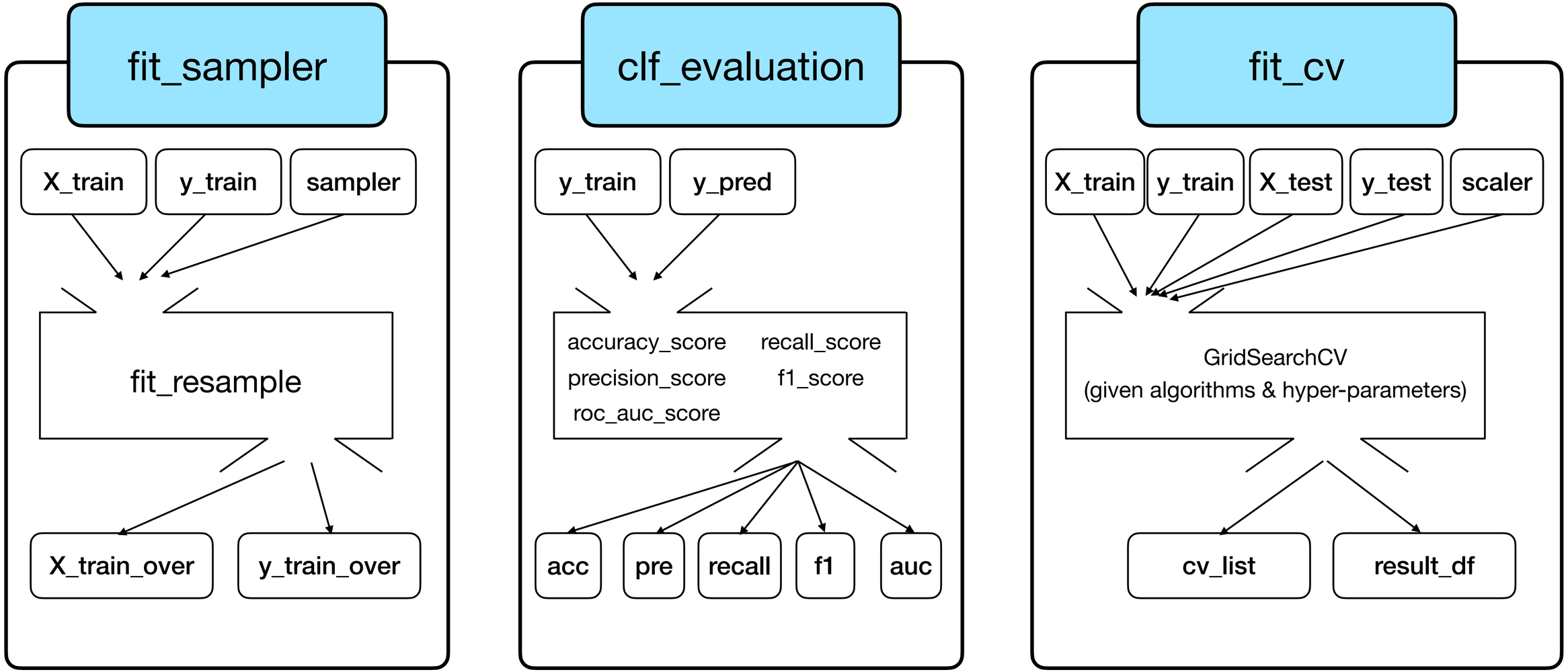
5가지 알고리즘

- **Logistic Regression**
- **Decision Tree**
  - max_depth: [3, 4, 6, 8, 10, 30]
  - max_features: [None, sqrt, log2]
- **Random Forest**
  - n_estimators: [50, 100, 200, 400]
  - max_depth: [4, 6, 8, 10, 30]
- **Light GBM**
  - n_estimators: [50, 100, 200, 400]
  - num_leaves: [4, 8, 16]
- **Support Vector Classification**
  - C: [0.1, 1.0]

2가지 경우
(class_weight 튜닝)

- 미적용
- 적용([
  {0: 0.01, 1: 1.0},
  {0: 0.005, 1:1},
  'balanced'
  ])

Module 설명


fit_sampler
X_train
y_train
sampler
fit_resample
X_train_over
y_train_over
clf_evaluation
y_train
y_pred
accuracy_score
precision_score
roc_auc_score
recall_score
f1_score
acc
pre
recall
f1
auc
fit_cv
X_train
y_train
X_test
y_test
scaler
GridSearchCV
(given algorithms & hyper-parameters)
cv_list
result_df

3. 결과 및 최종 선택 모델

결과 및 최종 선택 모델

1. 결과

	class_weight_YN	null_del	encoded	scaler	sampler	classifier	train accuracy	train recall	test accuracy	test recall
1	0	0.0	1.0	None	BdlSMOTE	SVC	0.632224	0.999219	0.284204	1.000000
2	0	0.0	0.0	None	SVMSMOTE	LogisticReg	0.774697	0.995547	0.684396	0.857143
3	1	1.0	0.0	None	SVMSMOTE	RandomForest	0.862384	0.995547	0.817687	0.714286
4	1	1.0	1.0	SD	SVMSMOTE	LogisticReg	0.884418	0.998381	0.745274	0.571429
5	0	0.0	0.0	None	RandomOverSampler	LightGBM	0.939082	1.000000	0.830503	0.428571

2. 최종 선택 모델

- 1st (**Best**): **2**번째 모델 (test recall=0.86, test accuracy=0.68)
- 2nd: recall과 accuracy 중 우선순위에 따라 1번째 혹은 3번째 모델

4. 추후 개선점, 아쉬웠던 점, 의문

추후 개선점, 아쉬웠던 점, 의문

1. 추후 개선점

- **Null값 처리**
  : 클러스터링을 통한 대체값 등 삭제 외 다른 방법 적용

- **hyper-parameter 튜닝**
  : 더 다양한 파라미터 (ex. SVC에서 gamma 값) 및 세분화된 튜닝

2. 아쉬웠던 점

- **세분화된 시간 데이터**
  : 시간대가 아닌 구체적 시간 및 날짜 데이터였다면…

3. 의문점

- Logistic Regression과 RandomForest의 경우, 인코딩 미적용한 경우가 성능이 더 좋게 나타남
  → 알고리즘 특성상 인코딩 효과가 미미하거나, 오히려 미적용하는 것이 성능이 더 우수할 수 있나?

Q & A

SOCAR

E.O.D