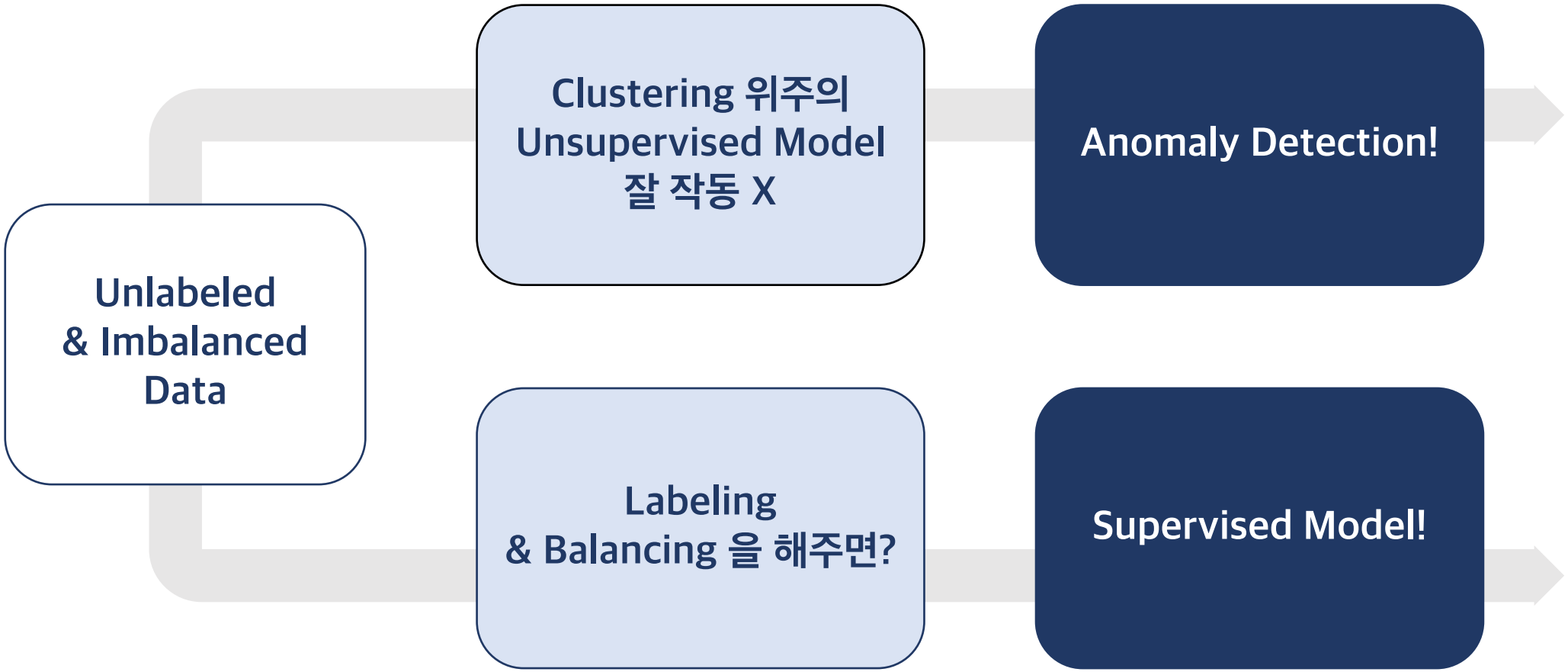
신용카드 사기 거래 탐지 AI

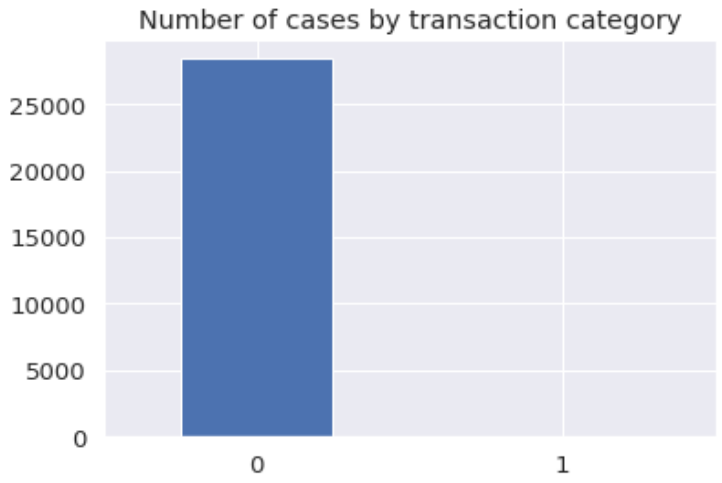
YBIGTA 22기
김세중 김정환 안서연 양재훈 엄소현 정병주

목차

프로젝트 flow


Unlabeled
& Imbalanced
Data
Clustering 위주의
Unsupervised Model
잘 작동 X
Anomaly Detection!
Labeling
& Balancing 을 해주면?
Supervised Model!

Scoring Metric



Data Imbalance

평가 기준 : Macro-F1 Score

✓ 클래스/레이블 별 F1 Score의 평균

✓ 모든 class에 동등한 중요성 부여

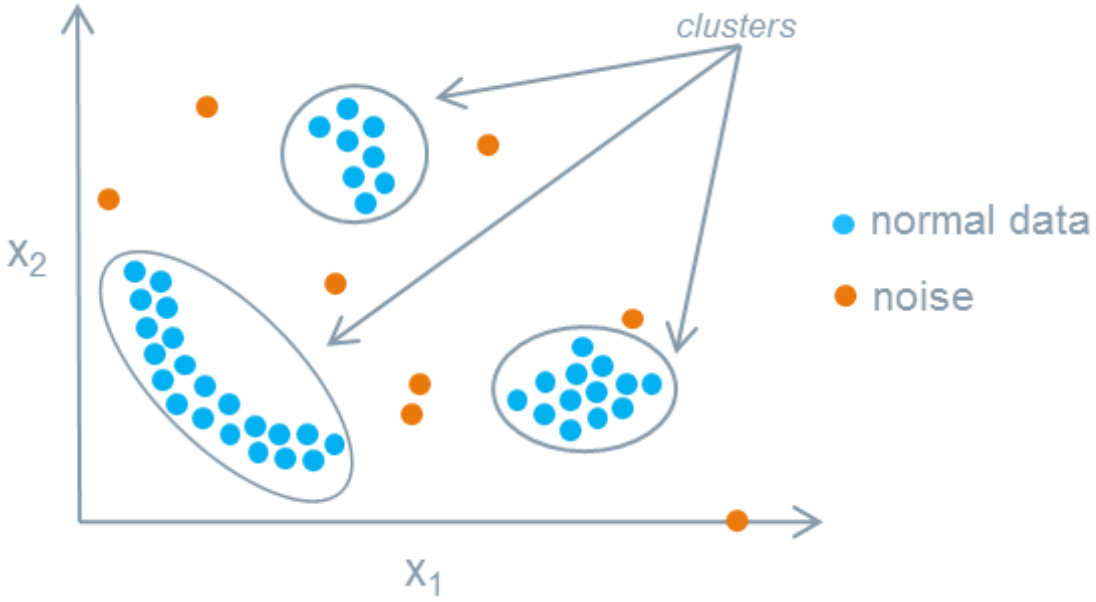
✓ Data Imbalance 가 심할 때 사용하기 적절한 메트릭

Dacon 점수 산출 : Public / Private

✓ Public? Test set 중 30%의 데이터로 채점한 결과

✓ Private? Test set 중 70%의 데이터로 채점한 결과

Anomaly Detection ?



이상치 탐지

사기 거래를 이상치로 판단

Anomaly Detection

단일 모델 결과

Model	Validation Macro F1 score
Elliptic Envelope	0.9236
Isolation Forest	0.8156
One Class SVM	0.6893
Local Outlier Factor	0.7498
Auto Encoder	0.9166

➔ 좋은 성능을 보인 모델들에 대해 앙상블 진행

Anomaly Detection

앙상블 결과

Model combination	Voting method	Validation macro F1 score
AE + EE + IF + SVM	Hard voting	0.8967
AE + EE + IF	Hard voting	0.9236
AE + SVM + IF	Hard voting	0.8075
AE + SVM	Hard voting	0.6665
EE + IF	Hard voting	0.8729

Best !!

AE : Auto Encoder

EE : Elliptic Envelope

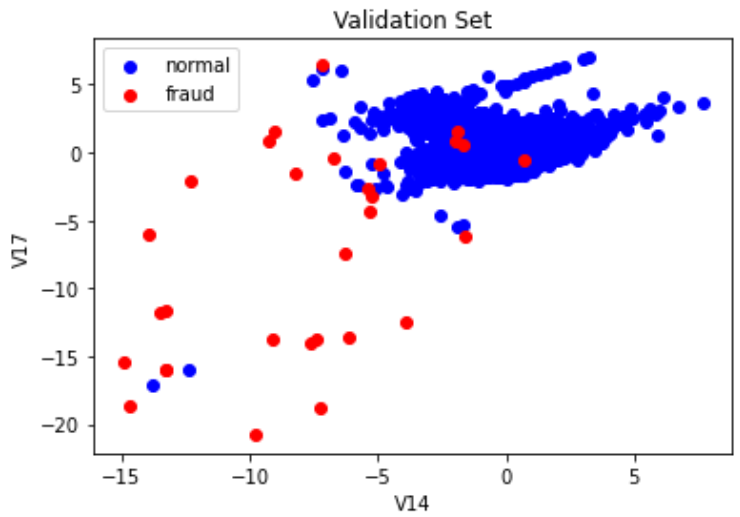
IF : Isolation Forest

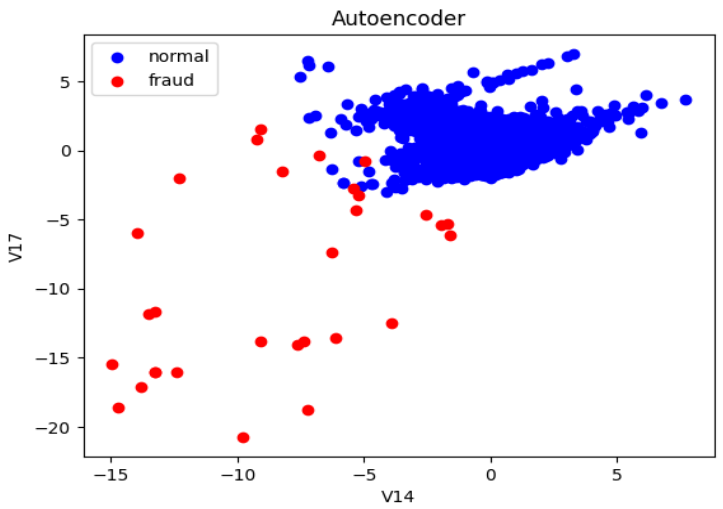
SVM : One Class SVM

Anomaly Detection

Best 앙상블에서 사용된 단일 모델 (1) **Auto Encoder**

Model combination	Voting method	Validation macro F1 score
AE + EE + IF	Hard voting	0.9236



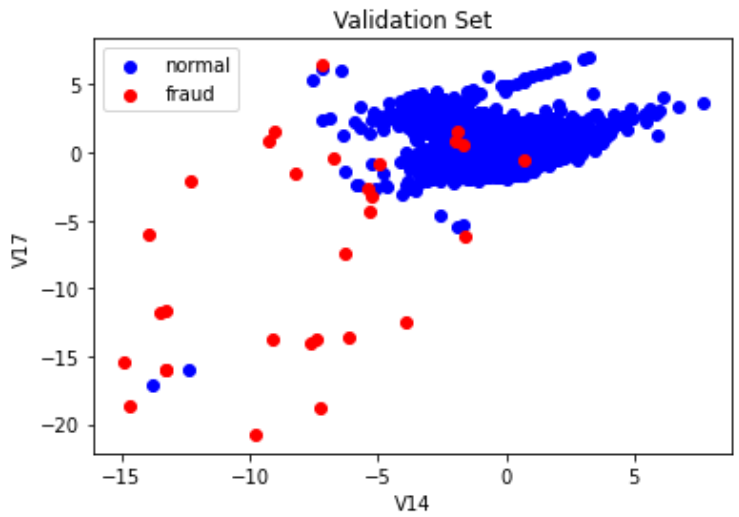


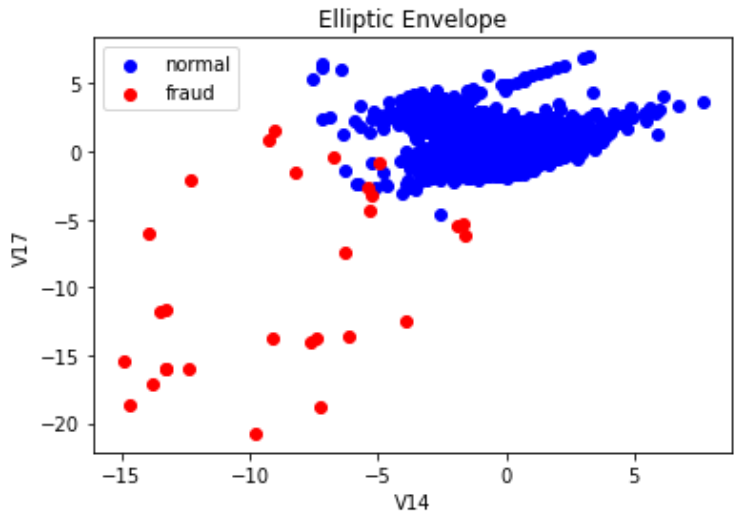
Macro F1 Score : 0.9166

Anomaly Detection

Best 앙상블에서 사용된 단일 모델 (2) **Elliptic Envelope**

Model combination	Voting method	Validation macro F1 score
AE + EE + IF	Hard voting	0.9236



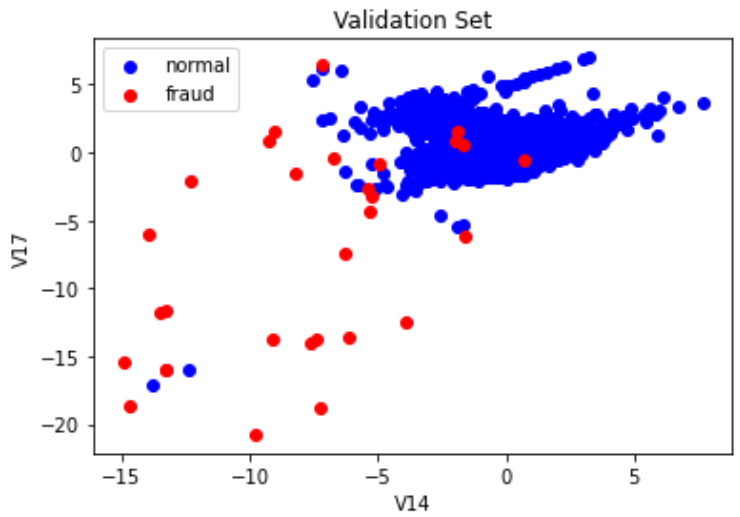


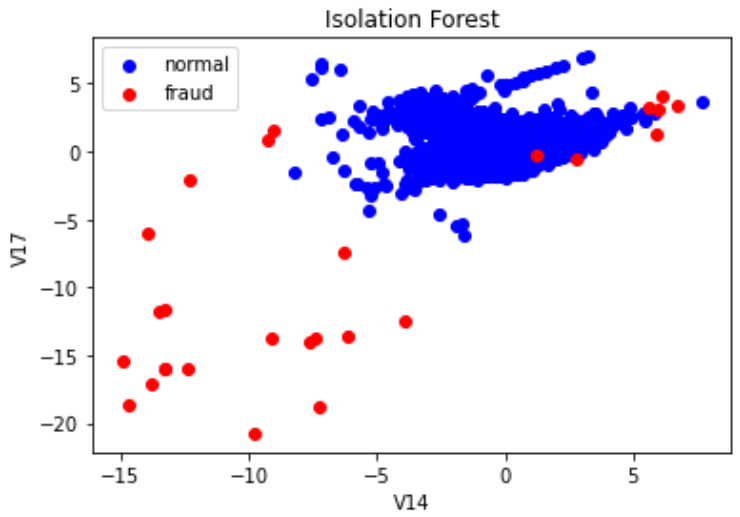
Macro F1 Score : 0.9236

Anomaly Detection

Best 앙상블에서 사용된 단일 모델 (3) **Isolation Forest**

Model combination	Voting method	Validation macro F1 score
AE + EE + IF	Hard voting	0.9236

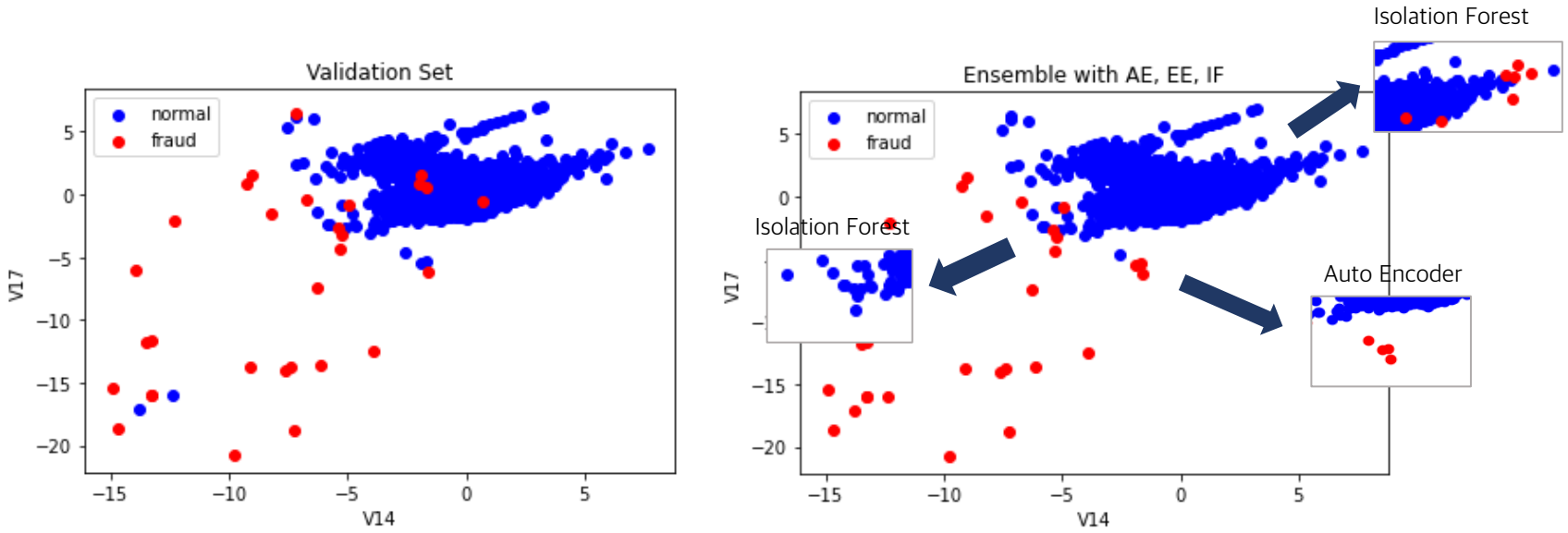




Macro F1 Score : 0.8136

Anomaly Detection

Best 앙상블 : **Auto Encoder + Elliptic Envelope + Isolation Forest**

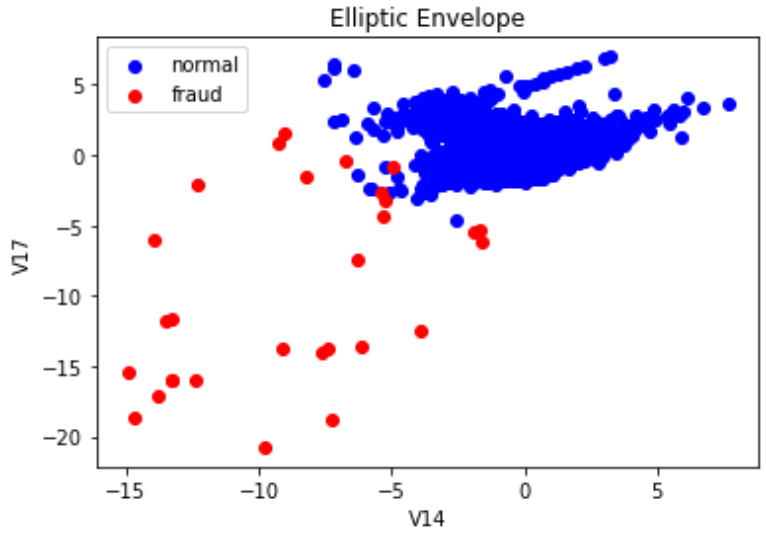


Test 결과 Public: 0.9277 (공동 192위, 상위 23%)

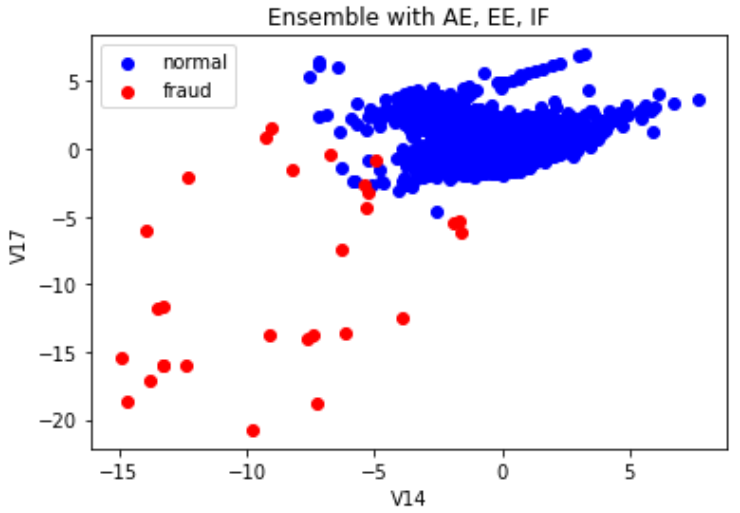
Private: 0.9095 (공동 44위, 상위 6%)

Anomaly Detection

한계점



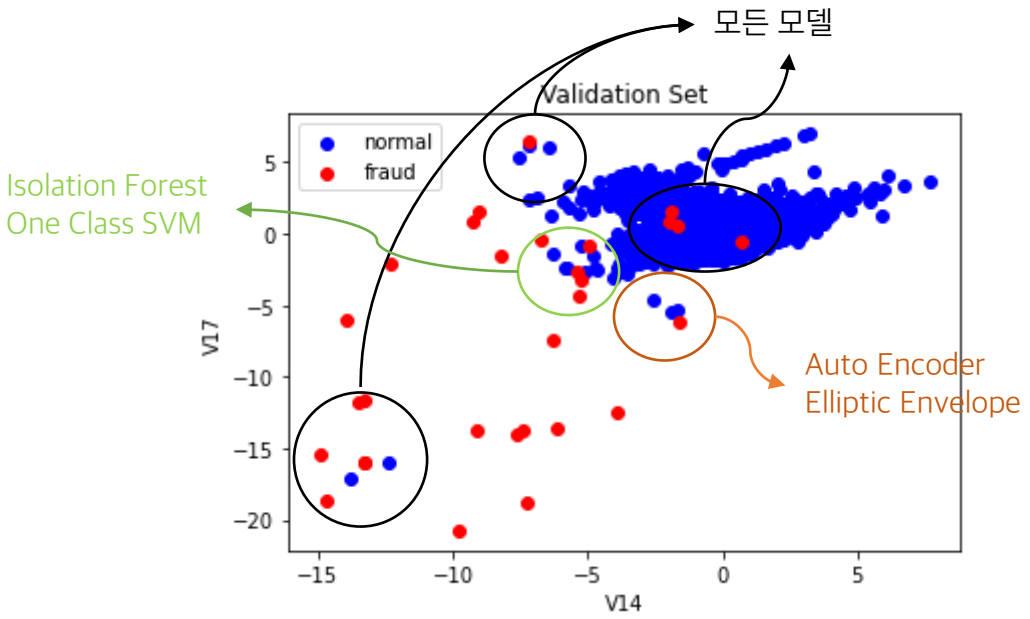
=



그러나, Elliptic Envelope 단일 모델과 같은 성능

Anomaly Detection

한계점



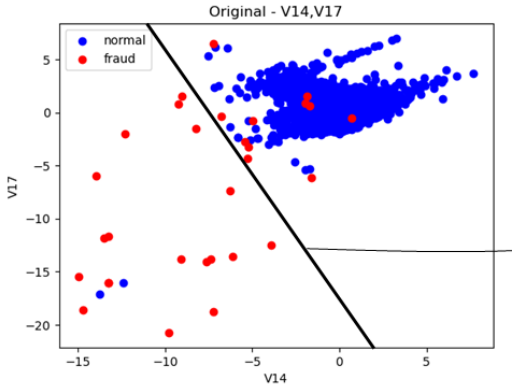
1. 모델들이 공통적으로 잘 못 잡아내는 부분 존재 (경계 부분, 정상같은 사기, 사기같은 정상)
2. **Voting**을 통해 상호보완하기를 기대했지만 어떤 방식을 취해봐도 결국 **trade-off** 관계
   (왼쪽 경계 부분을 맞히면 오른쪽 경계 부분은 틀림)

Supervised Model

Idea

Train data labeling → Over sampling → Supervised learning

EDA



SMOTE

XGBoost
CatBoost
AdaBoost
LGBM
+
Ensemble
Stacking

Supervised Model

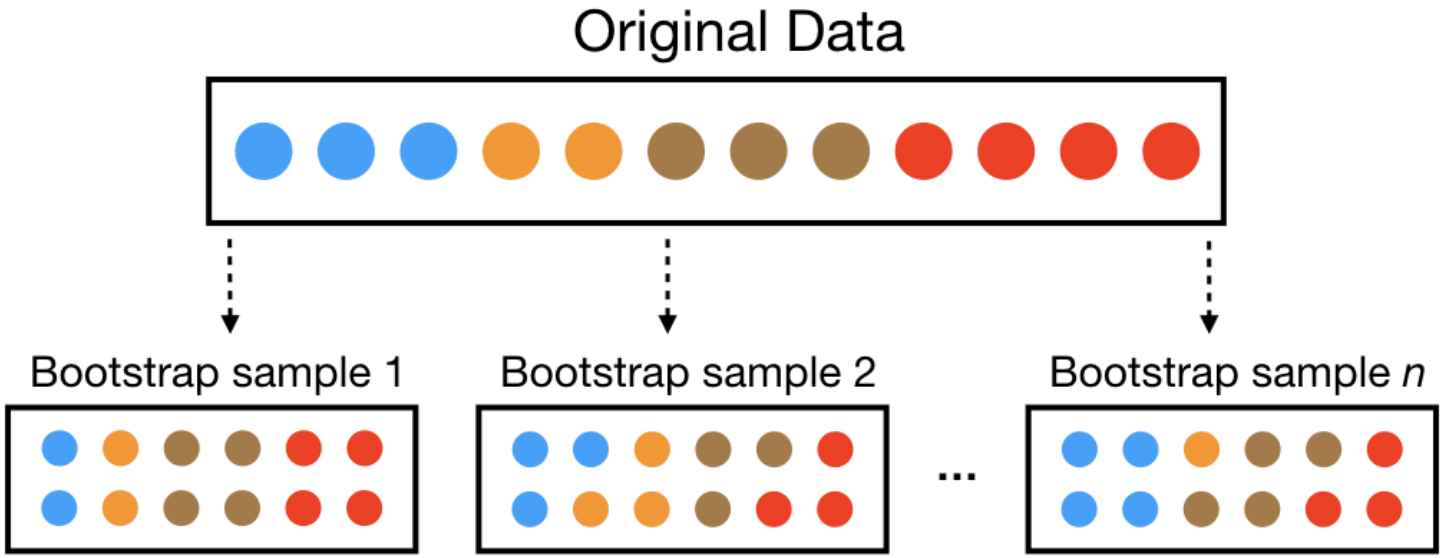
모델 결과 요약

	Labeling + Oversampling	모델	조합	Valid Macro f1 score
1	EDA + SMOTE	XGBoost	단일	0.9181
2	EDA + SMOTE	CatBoost	단일	0.9181
3	EDA + SMOTE	Ada + XGBoost + LGBM + CatBoost	Hard Voting Ensemble	0.9073
4	EDA + SMOTE	Ada + XGBoost + LGBM + CatBoost	Soft Voting Ensemble	0.9073
5	EDA + SMOTE	Ada + XGBoost + LGBM + CatBoost	CV Stacking	0.9181

많은 시도를 해봤지만, 모델의 하이퍼파라미터 튜닝으로는 성능 향상에 한계가 있다고 느낌…

Supervised Model

Further?



많은 시도를 해봤지만, 모델의 하이퍼파라미터 튜닝으로는 성능 향상에 한계가 있다고 느낌…

➔ 일종의 bootstrap처럼 **서로 다른 data set을 각각 모델링하여 앙상블** 하는 방식 고안

Supervised Model

서로 다른 data set을 앙상블 해보자!

- None
- SMOTE
- Adasyn

- EDA를 통한 labeling
- Elliptic Envelope를 통한 labeling
- KNN을 통한 labeling

- Xgboost
- LGBM
- Decisiontree
- Adaboost

Supervised Model

서로 다른 data set을 앙상블 해보자!

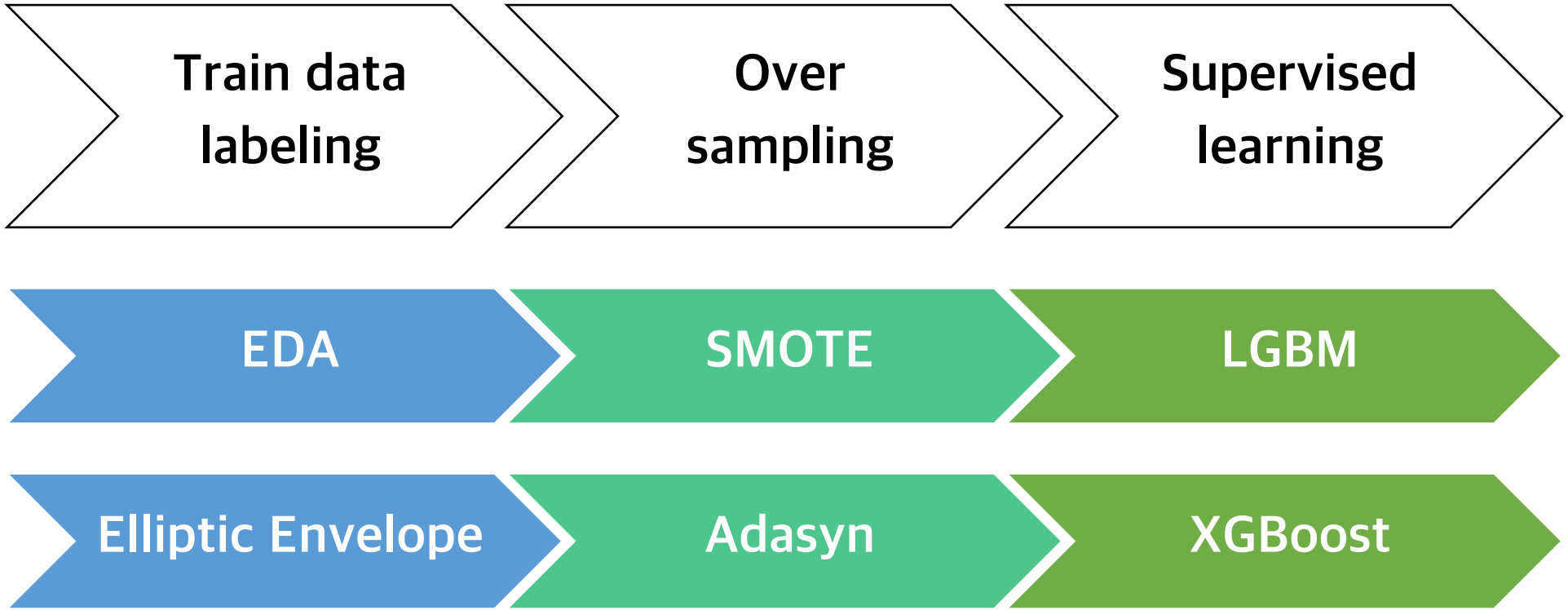
- None
- **SMOTE**
- **Adasyn**

- **EDA를 통한 labeling**
- **Elliptic Envelope를 통한 labeling**
- KNN을 통한 labeling

- **Xgboost**
- **LGBM**
- Decisiontree
- Adaboost

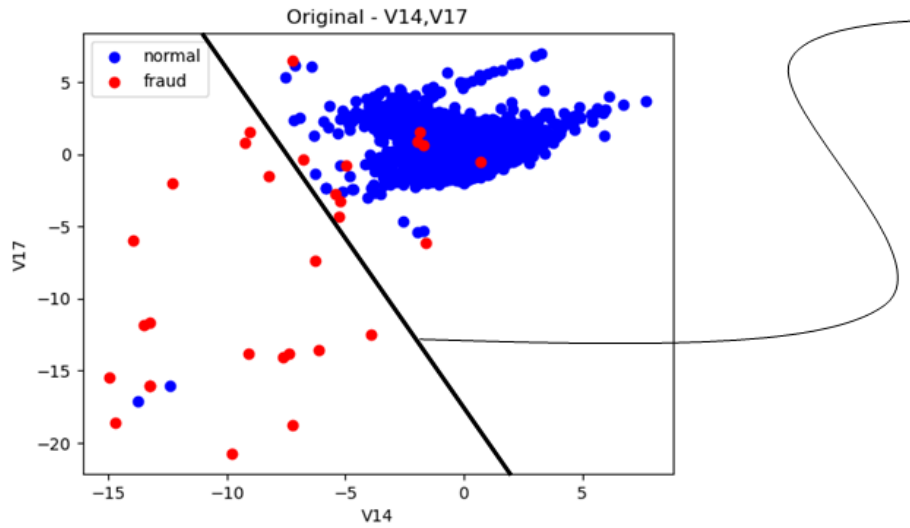
Supervised Model

단일 모델 set



Supervised Model

EDA SMOTE LGBM



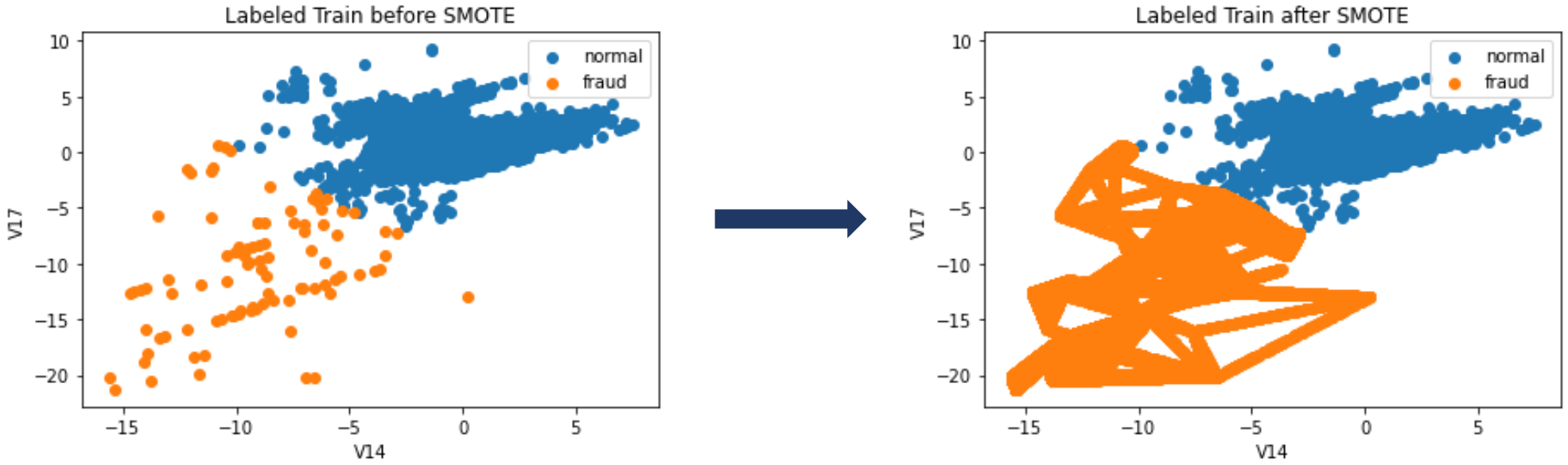
EDA를 통해 class의 대략적 분류 가능

: Target 변수인 Class와 상관계수가 가장 높았던 V14, V17 변수를 통해 기준 설정 후 train data labeling

```
train['Class'] = ((train.V14 + train.V17 + 10) < 0)
```

Supervised Model

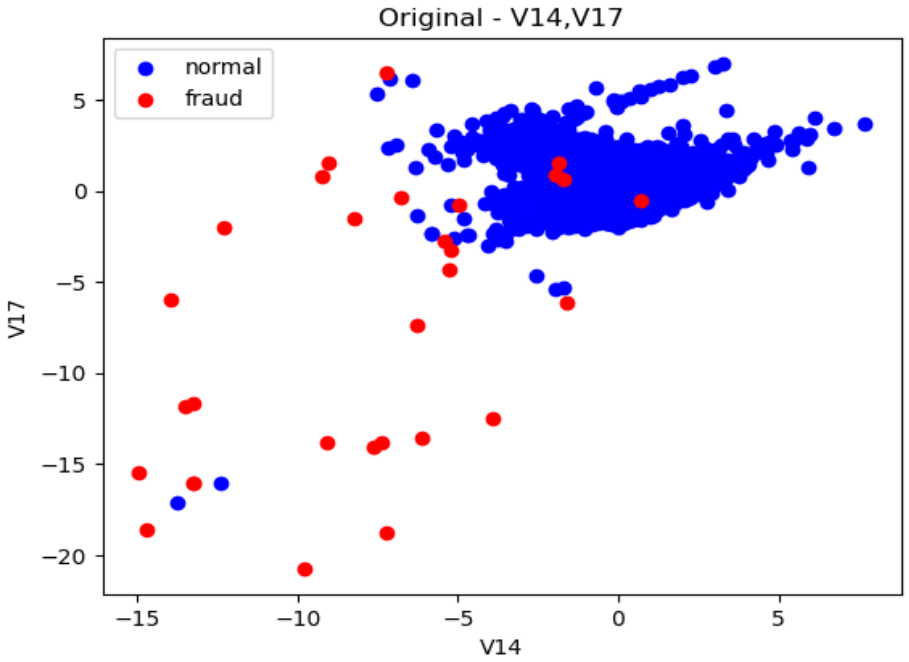
EDA SMOTE LGBM

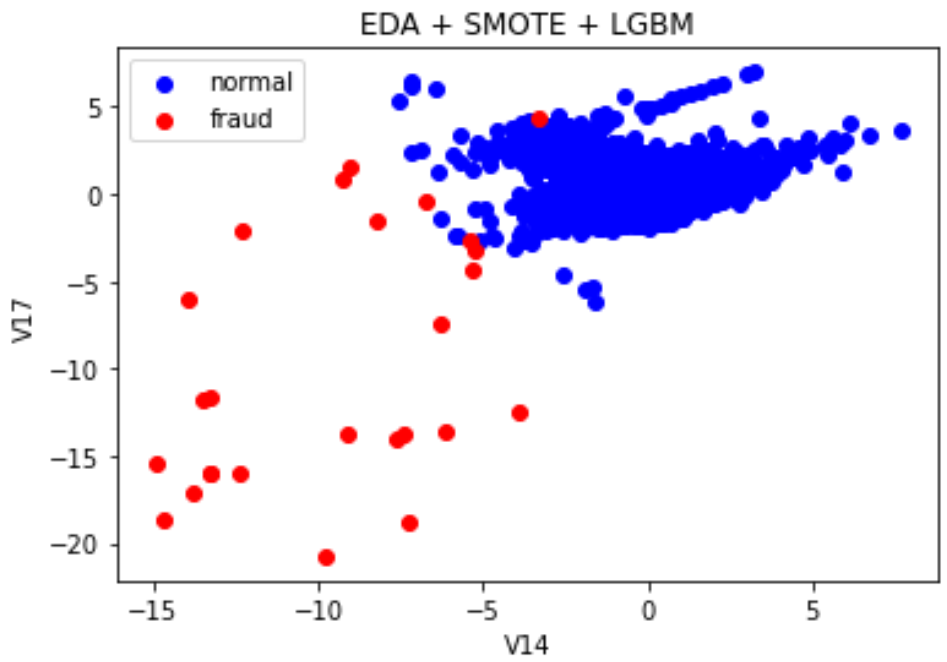


SMOTE를 통해 Minority에 해당하는 fraud data에 대해 Over Sampling

Supervised Model

EDA SMOTE LGBM

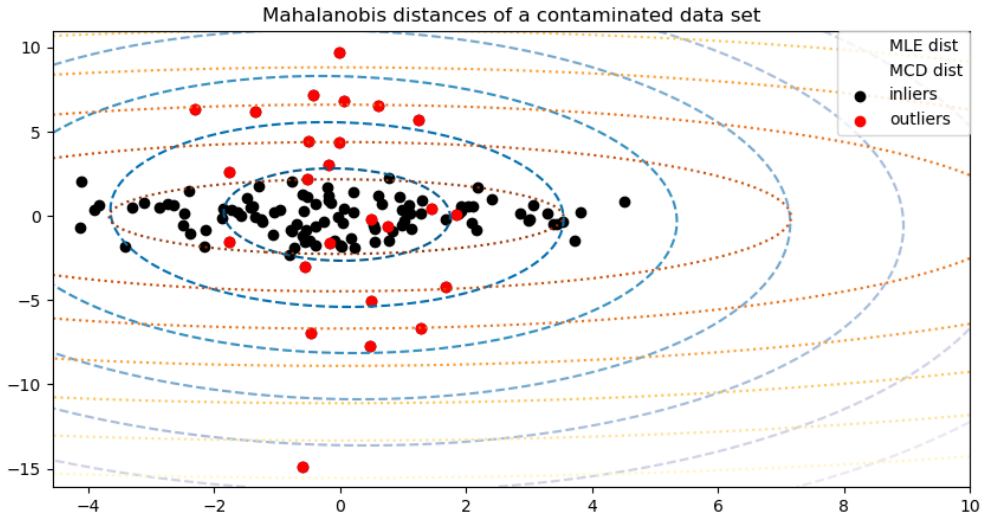


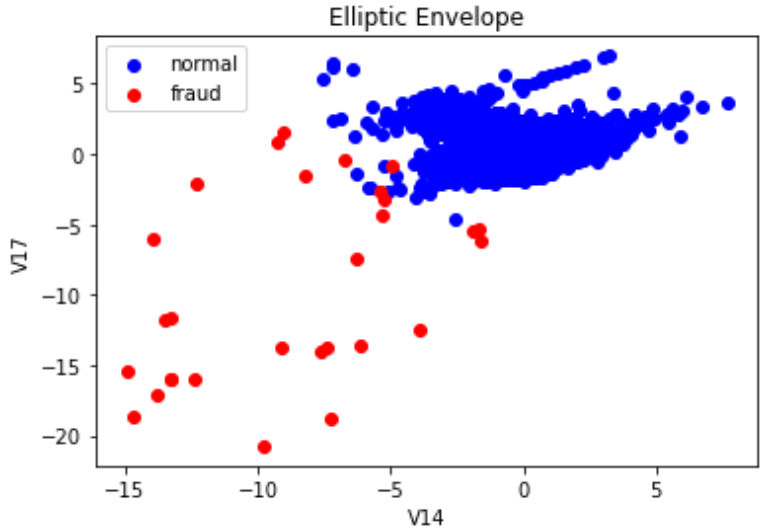


Validation Macro f1 score : 0.9106

Supervised Model

EE | Adasyn | XGBoost

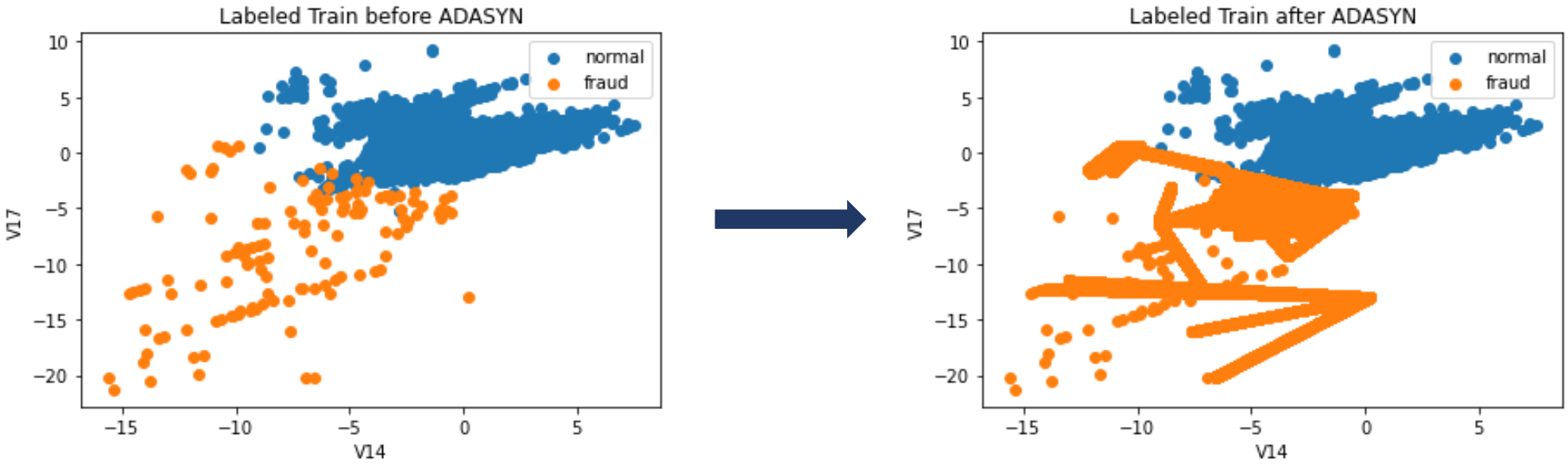




Anomaly Detection에서 가장 성능이 좋았던 Elliptic Envelope의 prediction 결과로 train data labeling

Supervised Model

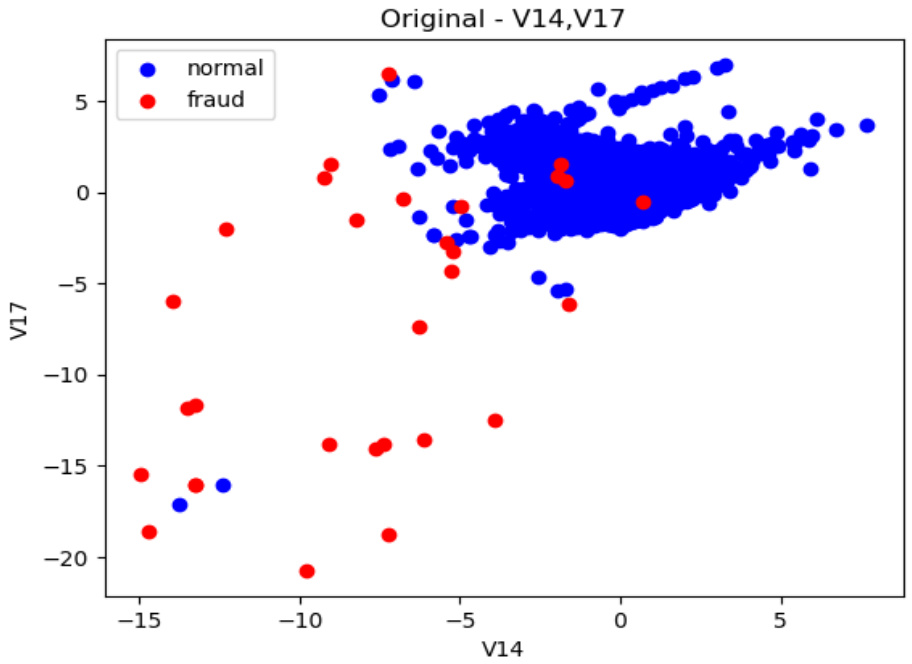
EE | Adasyn | XGBoost

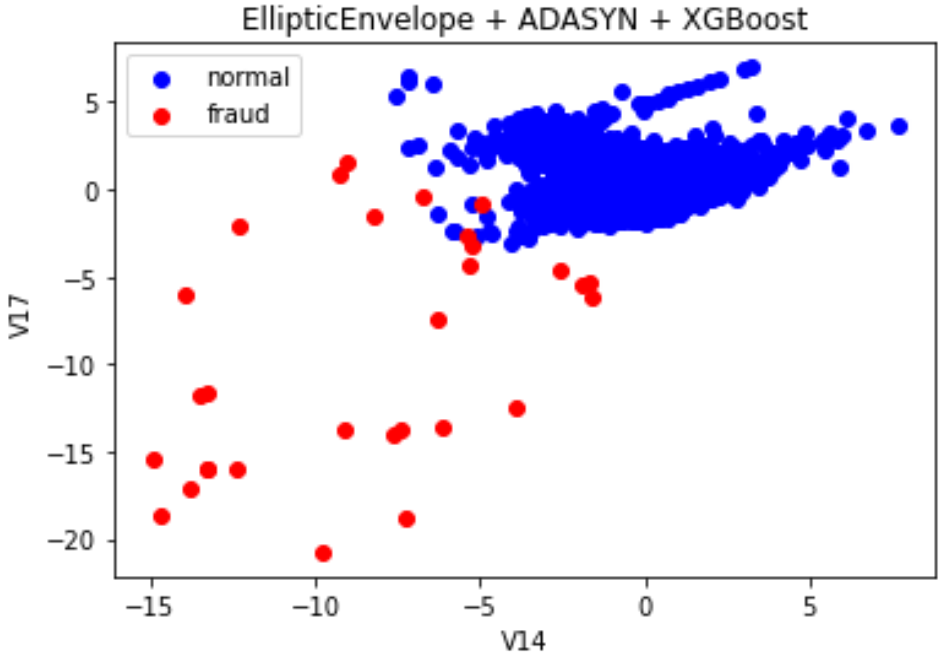


ADASYN을 통해 Minority에 해당하는 fraud data에 대해 Over Sampling
(SMOTE와 ADASYN의 차이? 랜덤한 오차를 부여함)

Supervised Model

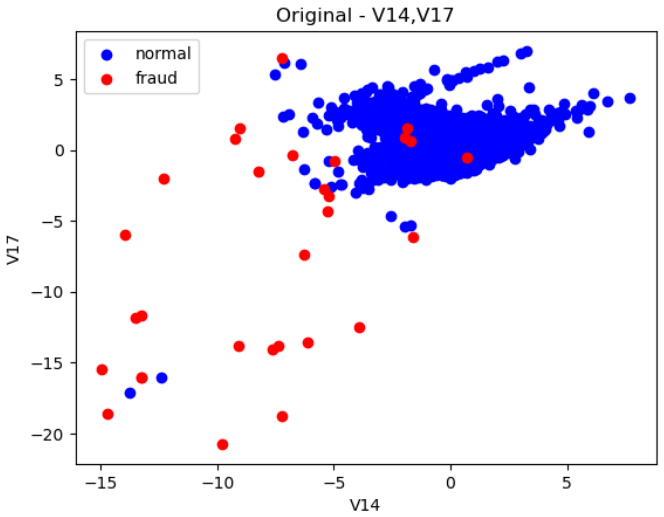
EE Adasyn XGBoost

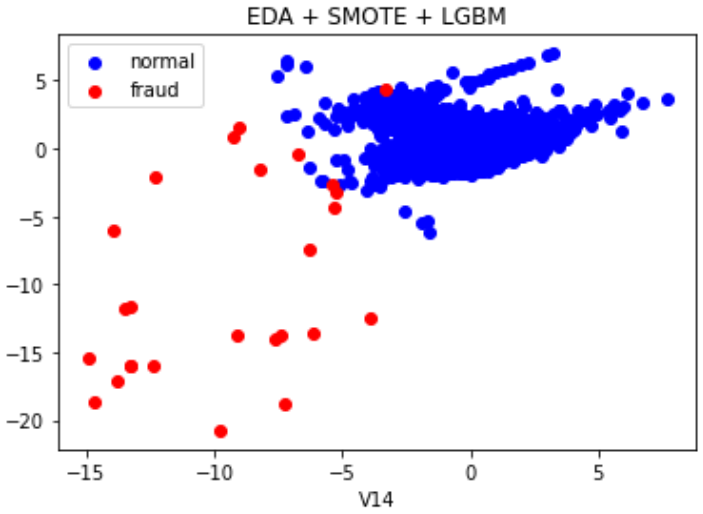


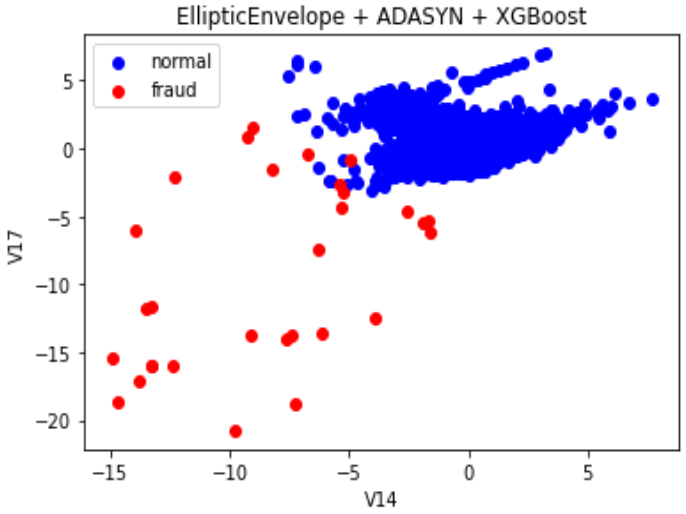


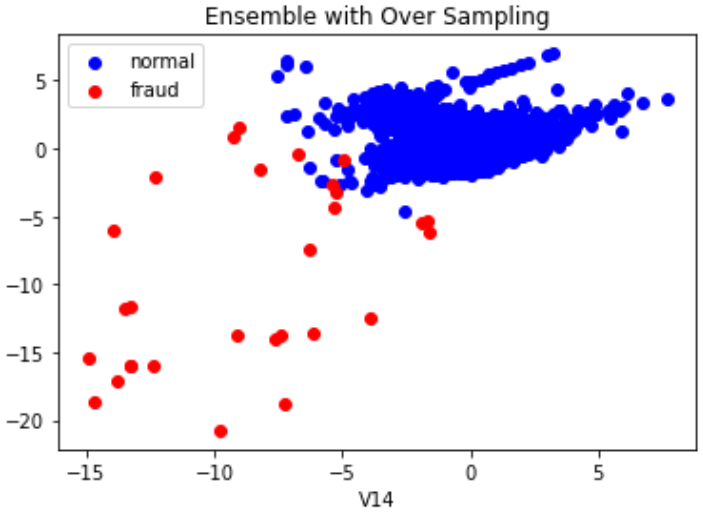
Validation f1 score : 0.9106

Supervised Model









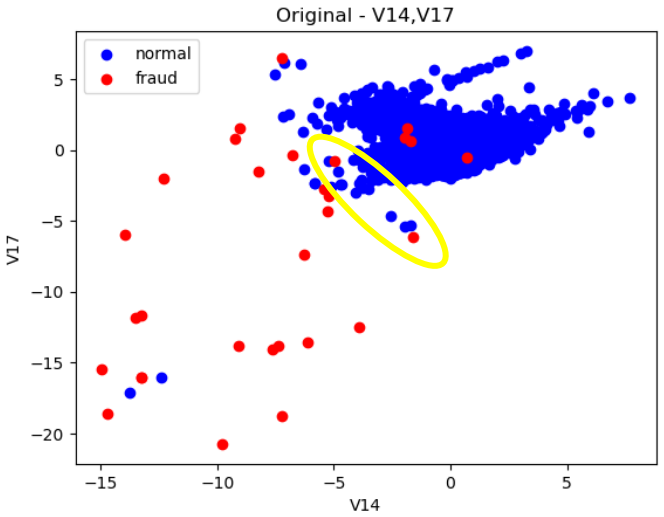
앙상블 결과

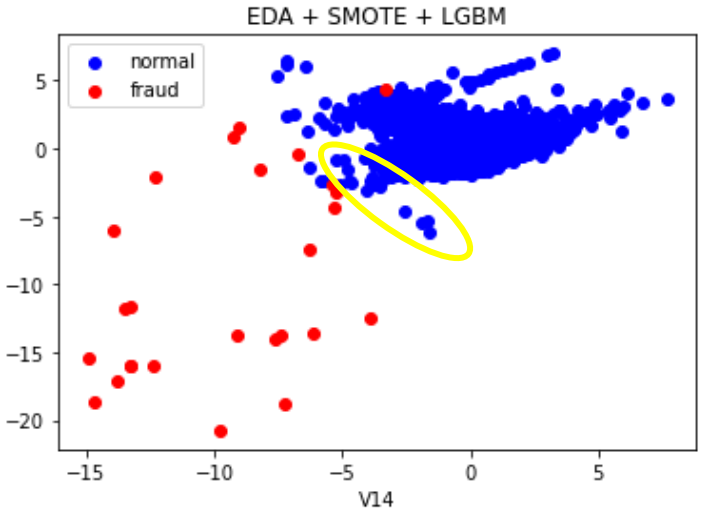
EDA → SMOTE → LGBM

Elliptic Envelope → Adasyn → XGBoost

가장 좋은 성능을 보였던 2개의 set를 앙상블

Supervised Model



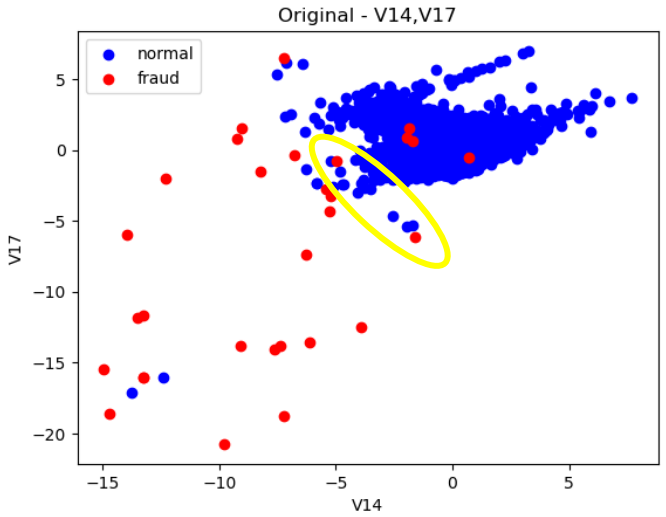


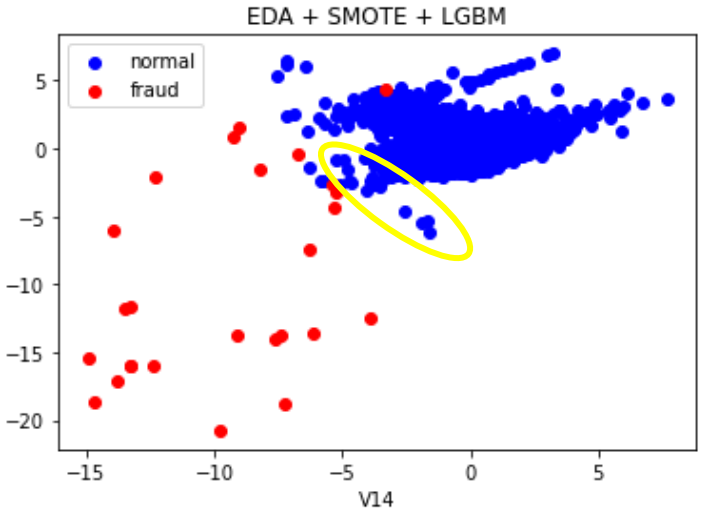
앙상블 결과

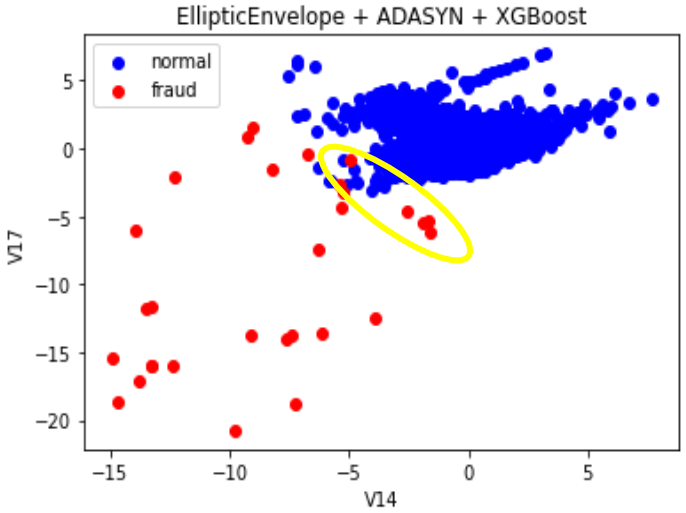
EDA → SMOTE → LGBM

경계 부분에 있는 fraud data 탐지 X

Supervised Model







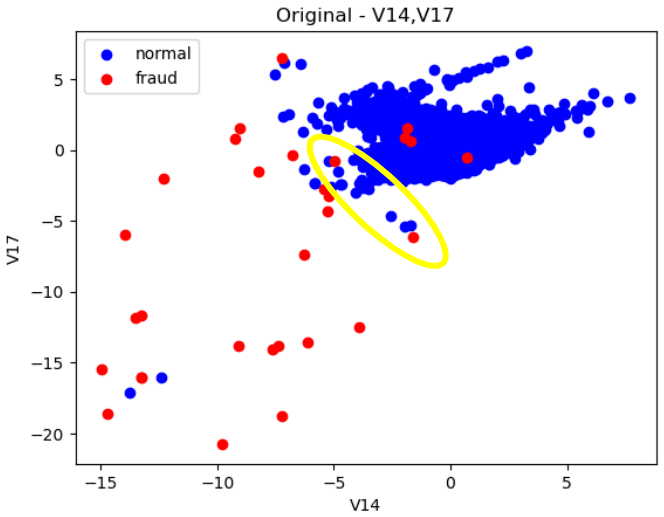
앙상블 결과

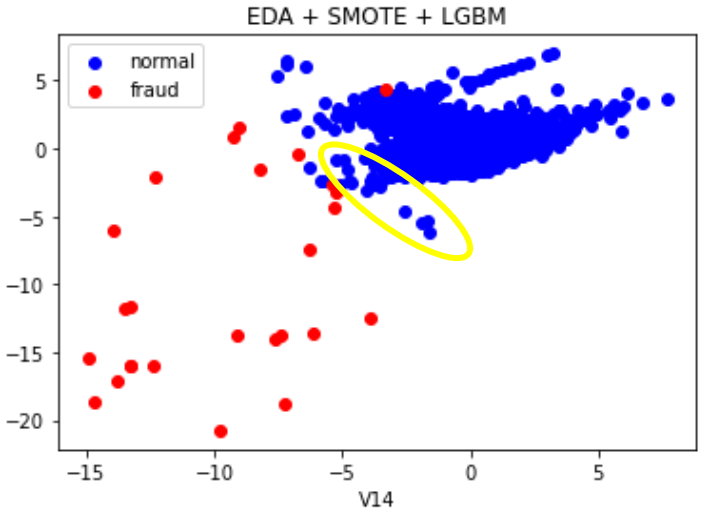
EE → Adasyn → XGBoost

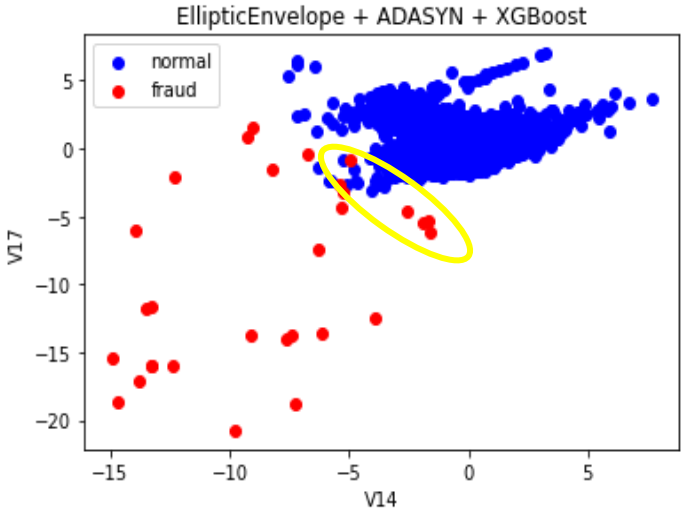
경계에 있는 fraud data 탐지하지만,

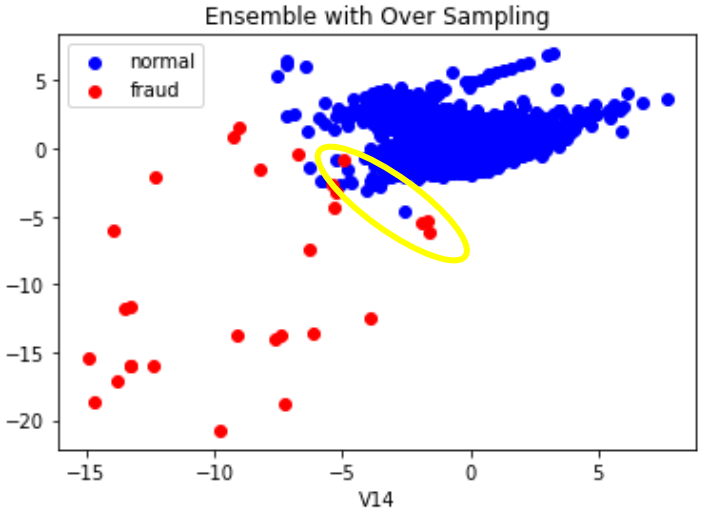
normal data를 fraud data로 잘못 탐지하기도

Supervised Model









앙상블 결과

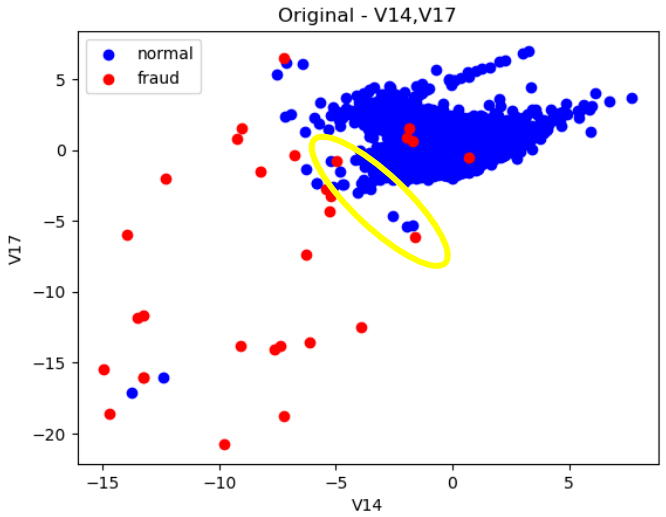
EDA → SMOTE → LGBM

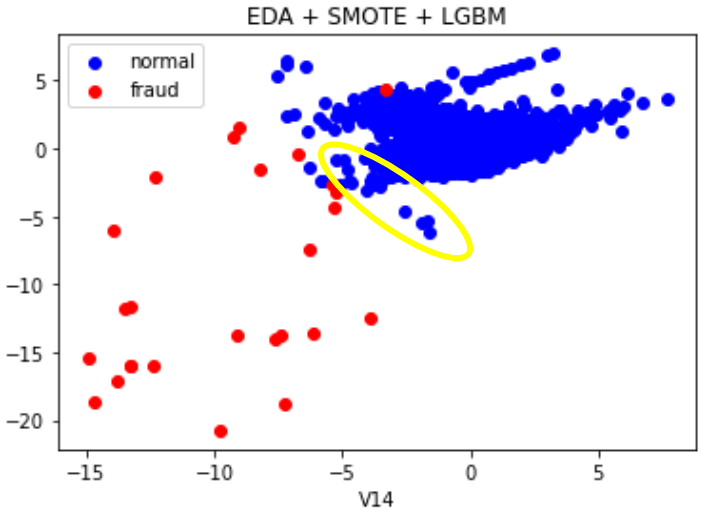
EE → Adasyn → XGBoost

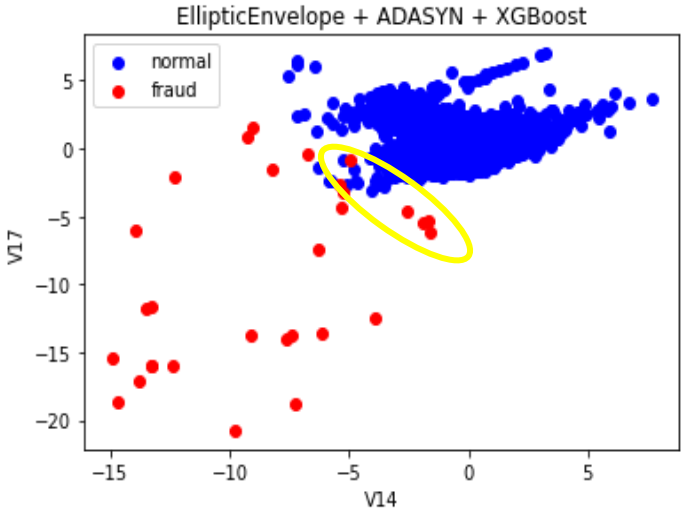
두 가지의 개별 모델 set보다

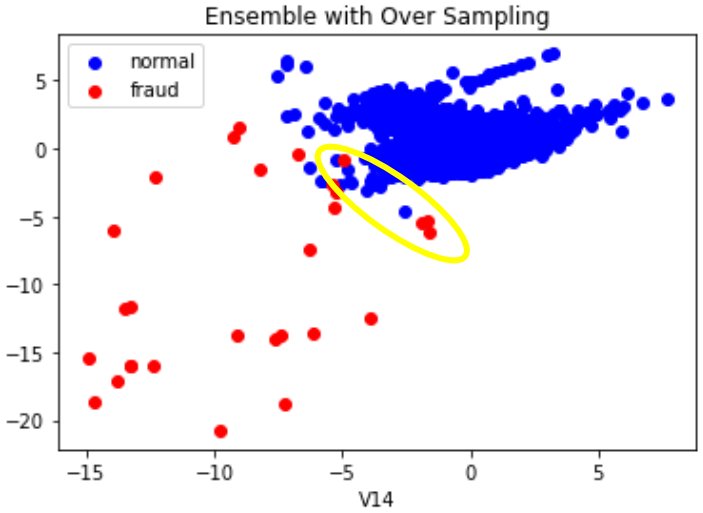
fraud data와 normal data 구별 능력 향상

Supervised Model









앙상블 결과

EDA → SMOTE → LGBM

EE → Adasyn → XGBoost

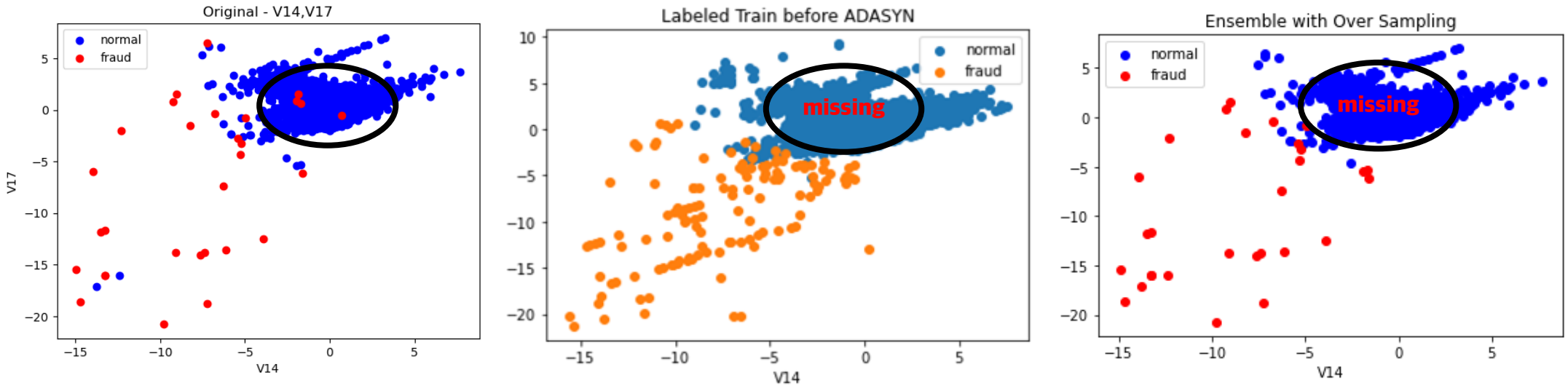
Test 결과

Public: 0.9305 (공동 17위, 상위 20%)

Private: 0.9081 (공동 77위, 상위 10%)

Supervised Model

한계점



서로 다른 개별 모델을 앙상블 ➔ 오류를 보완하여 일정 수준 이상의 f1 score는 달성 가능

하지만, train data를 임의로 라벨링 ➔ 잘못 라벨링 된 데이터는 detecting하는 데 한계가 존재

라벨링 후에 랜덤한 오차를 부여하는 **Adasyn**의 이점이 더 클 수 있음

Supervised Model

SMOTE 대신 Adasyn을 써서 앙상블 해보자 !

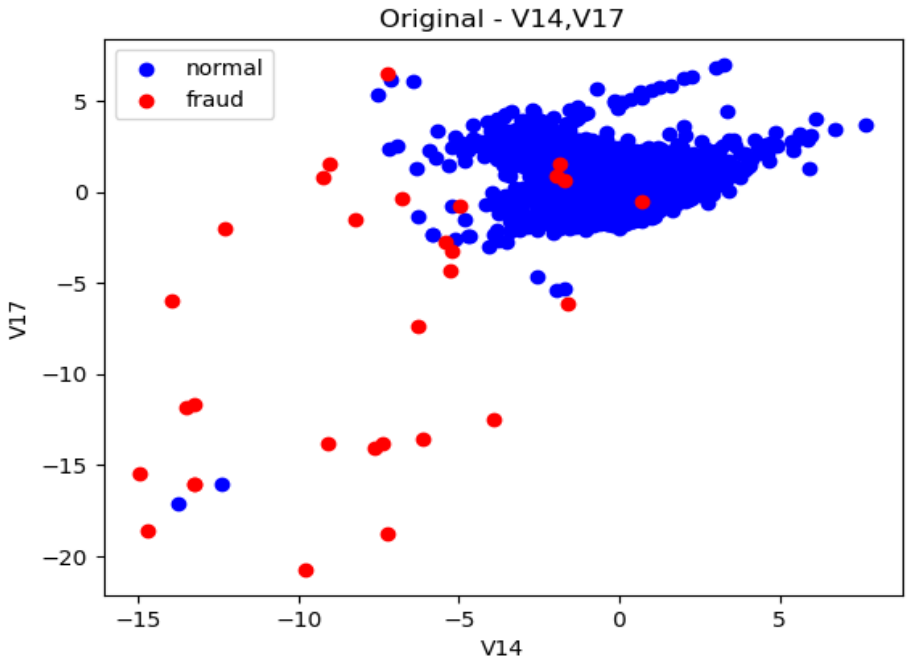
- None
- SMOTE
- **Adasyn**

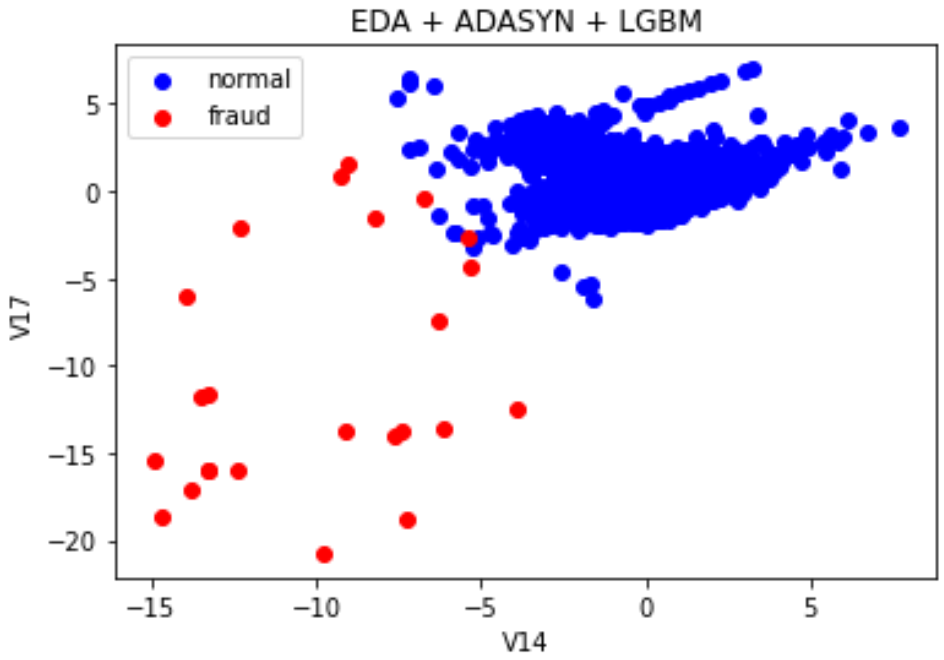
- **EDA를 통한 labeling**
- **Elliptic Envelope를 통한 labeling**
- KNN을 통한 labeling

- **Xgboost**
- **LGBM**
- Decisiontree
- Adaboost

Supervised Model

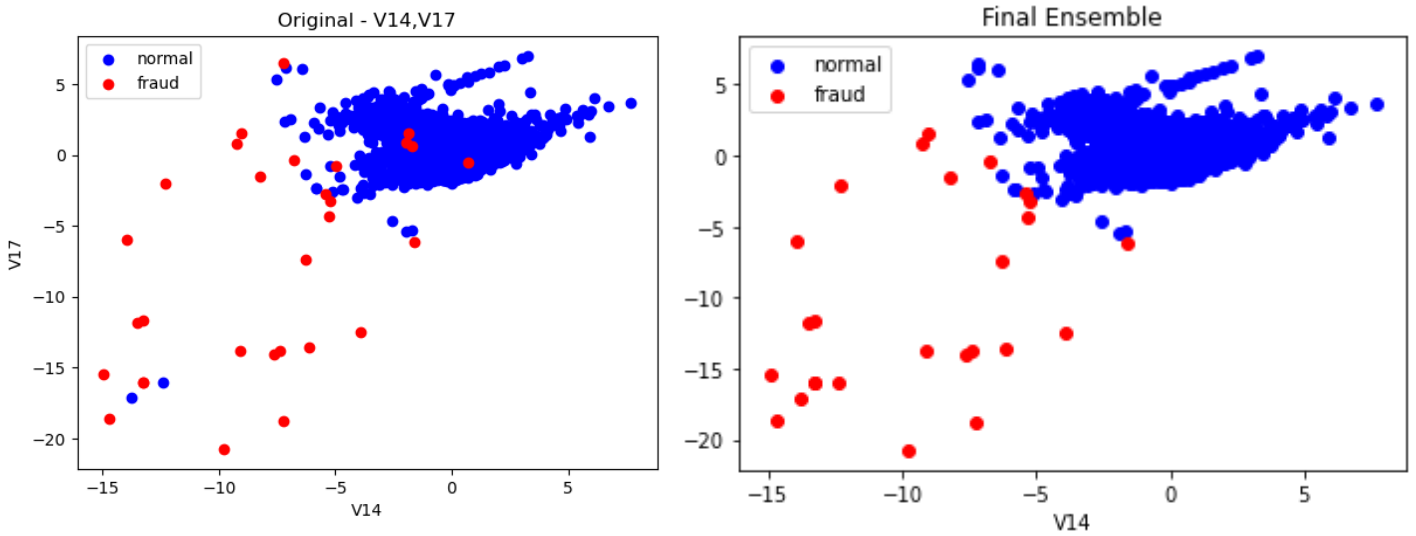
EDA → Adasyn → LGBM



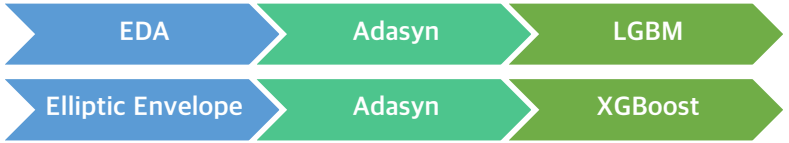


Valid f1 score : 0.9073

Supervised Model

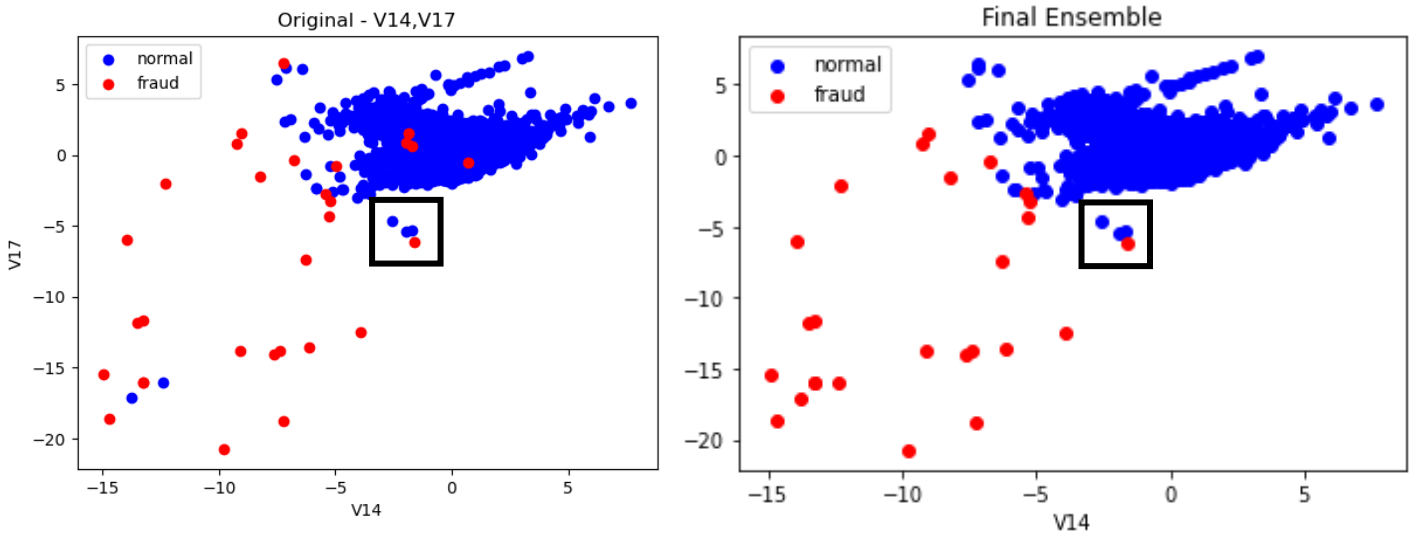


앙상블 결과

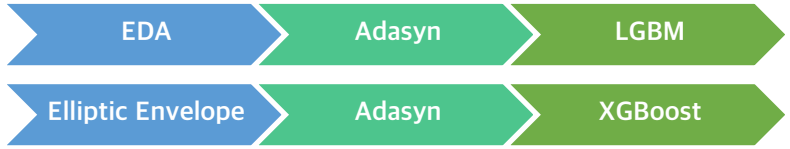


Adasyn을 사용한 두개의 단일 set를 앙상블

Supervised Model



앙상블 결과

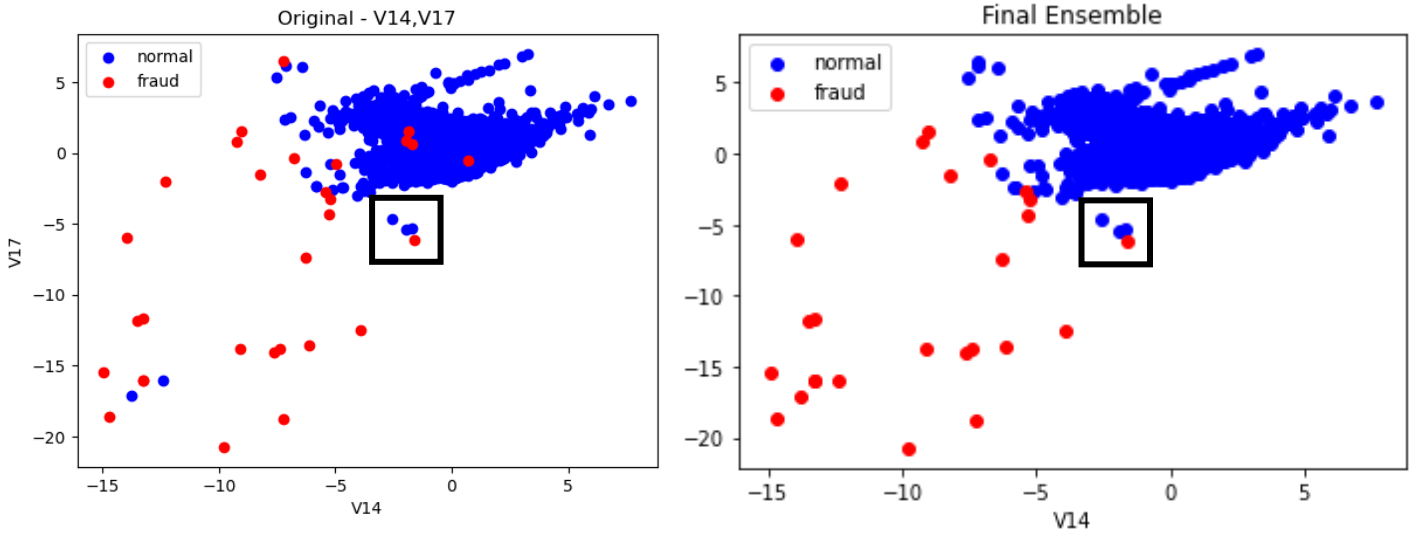


Adasyn을 사용한 두개의 단일 set를 앙상블

- fraud data 탐지 Good
- fraud / normal data 구분도 Good

➔ 지속적으로 문제가 되었던 부분 해결

Supervised Model



앙상블 결과

EDA → Adasyn → LGBM

Elliptic Envelope → Adasyn → XGBoost

Valid f1 score	0.9285
Test f1 score	Public 0.9219 (237위, 상위 27%)
	Private 0.9110 (8위, 상위 1%)
	(Data Leakage 위반 팀을 제외하면 무려 4위!!)

Conclusion

Anomaly Detection
(Hard Voting Ensemble with
AE, EE, IF)

Test 결과
Public: 0.9277 (공동 192위, 상위 22%)
Private: 0.9095 (공동 44위, 상위 6%)

Supervised Learning
(Soft Voting Ensemble with
EDA labeling+SMOTE+LGBM
EE labeling+ADASYN+XGBoost)

Test 결과
Public: 0.9305 (공동 17위, 상위 20%)
Private: 0.9081 (공동 77위, 상위 10%)

Supervised Learning
(Soft Voting Ensemble with
EDA labeling+ADASYN+LGBM
EE labeling+ADASYN+XGBoost)

Test 결과
Public: 0.9219 (237위, 상위 27%)
Private: 0.9110 (8위, 상위 1%)

Conclusion

- ✓ Unlabeled & Imbalanced 데이터의 경우 Anomaly Detection 방법을 이용하는 것이 적절하다!

- ✓ 그러나 보다 높은, 안정된 성능을 위해서는 합리적인 기준에 따라 라벨링하여
  SMOTE나 ADASYN과 같은 Over Sampling 기법을 활용하는 것이 도움이 된다.

- ✓ 특히, 다양한 기준으로 over sampling한 데이터들을 일종의 bootstrap 데이터처럼 생각하여
  앙상블한다면 더욱 좋은 결과가 나올 수 있다.

- ✓ 덧붙여, SMOTE를 활용하면 특수 케이스들을 더 잘 맞출 수 있는 것 같지만(public 최고 점수),
  ADASYN 방식을 활용하면 예상치 못한 데이터에 대해서도 비교적 robust한 것으로 보인다.(private 최고 점수)

End of Presentation