KIPS Trans. Comp. and Comm. Sys.
Vol.11, No.12 pp.445~452
ISSN: 2287-5891 (Print), ISSN 2734-049X (Online)

https://doi.org/10.3745/KTCCS.2022.11.12.445

Imbalanced Data Improvement Techniques Based on SMOTE and Light GBM

Young-Jin Han† · In-Whee Joe††

ABSTRACT

Class distribution of unbalanced data is an important part of the digital world and is a significant part of cybersecurity. Abnormal activity of unbalanced data should be found and problems solved. Although a system capable of tracking patterns in all transactions is needed, machine learning with disproportionate data, which typically has abnormal patterns, can ignore and degrade performance for minority layers, and predictive models can be inaccurately biased. In this paper, we predict target variables and improve accuracy by combining estimates using Synthetic Minority Oversampling Technique (SMOTE) and Light GBM algorithms as an approach to address unbalanced datasets. Experimental results were compared with logistic regression, decision tree, KNN, Random Forest, and XGBoost algorithms. The performance was similar in accuracy and reproduction rate, but in precision, two algorithms performed at Random Forest 80.76% and Light GBM 97.16%, and in F1-score, Random Forest 84.67% and Light GBM 91.96%. As a result of this experiment, it was confirmed that Light GBM's performance was similar without deviation or improved by up to 16% compared to five algorithms.

Keywords : Machine Learning, Scaling, SMOTE, Light GBM, Imbalanced Classification

SMOTE와 Light GBM 기반의 불균형 데이터 개선 기법

한 영 진† · 조 인 휘††

요 약

디지털 세상에서 불균형 데이터에 대한 클래스 분포는 중요한 부분이며 사이버 보안에 큰 의미를 차지한다. 불균형 데이터의 비정상적인 활동을 찾고 문제를 해결해야 한다. 모든 트랜잭션의 패턴을 추적할 수 있는 시스템이 필요하지만, 일반적으로 패턴이 비정상인 불균형 데이터로 기계학습을 하면 소수 계층에 대한 성능은 무시되고 저하되며 예측 모델은 부정확하게 편향될 수 있다. 본 논문에서는 불균형 데이터 세트를 해결하기 위한 접근 방식으로 Synthetic Minority Oversampling Technique(SMOTE)와 Light GBM 알고리즘을 이용하여 추정치를 결합하여 대상 변수를 예측하고 정확도를 향상시켰다. 실험 결과는 Logistic Regression, Decision Tree, KNN, Random Forest, XGBoost 알고리즘과 비교하였다. 정확도, 재현율에서는 성능이 모두 비슷했으나 정밀도에서는 2개의 알고리즘 Random Forest 80.76%, Light GBM 97.16% 성능이 나왔고, F1-score에서는 Random Forest 84.67%,, Light GBM 91.96% 성능이 나왔다. 이 실험 결과로 Light GBM은 성능이 5개의 알고리즘과 비교하여 편차없이 비슷하거나 최대 16% 향상됨을 접근 방식으로 확인할 수 있었다.

키워드 : 기계학습, 스케일링, SMOTE, 라이트 GBM, 불균형 분류

1. 서 론

다양하고 거대한 데이터에서 수집하고 가공해서 학습을 시켜 결과치를 내는 건 많은 시간과 연구를 하게 한다. 불균형 데이터 상태로 학습 데이터에서 높은 성능을 보였다 해서 테스트 데이터에서 예측 성능이 더 낮아질 수 있으며, 과적합 문제가 발생하여 올바른 예측하기란 어렵다.[1] 이러한 문제는 금융사고[2], 희귀 질병 식별[3] 등에 이상 감지가 중요한 데이터에 다양하게 나타난다.

불균형 데이터를 처리할 경우 성능이 저하되는 것을 데이터 샘플링 기법으로 소수 샘플 개수를 조정하여 다수의 샘플에 균형 있게 데이터 집합으로 하는 기법으로 언더 샘플링(under sampling), 오버 샘플링(over sampling) 으로 분류된다. 언더 샘플링은 샘플을 균형 있게 하기 위해 다수 샘플들을 제거하는 방식이다. 여기에는 easyEnsemble[4] 등의 기법이 있으나 데이터를 제거하기 때문에 정보 손실의 문제점이 있다. 오버 샘플링은 언더 샘플링과는 반대이다. 소수의 샘플을 다수 샘플에 맞춰 생성하는 방식이다. 데이터 복제를 하기 때문에 데이터 과적합(overfitting)이 발생될 수 있으나 정보 손실은 피할 수 있다. 오버 샘플링은 SMOTE[5],

† 정 회 원 : 이화여자대학교 소프트웨어학부 외래교수
†† 정 회 원 : 한양대학교 컴퓨터소프트웨어학부 교수
Manuscript Received : July 5, 2022
Accepted : August 3, 2022
* Corresponding Author : In-Whee Joe(iwjoe@hanyang.ac.kr)

ADASYN[6] 등이 있다.

본 논문은 성능과 속도에 최적화되어 있는 모델 중에서 Light GBM[7] 모델을 이용하여 Kaggle의 credit card fraud 데이터를 사용하여 각 클래스들이 차이의 폭이 큰 상태로 모델을 학습하면, 패턴 분류를 다수의 범주로 많이 하게 됨으로 모델 성능에 영향을 끼치게 되는 불균형 데이터(imbalanced data) 문제를 해결할 수 있는 SMOTE로 성능을 높여 기존 알고리즘보다 성능 향상을 하고자 한다.

Light GBM은 Gradient Boosting 트리 알고리즘[8]에서 효과적인 오픈 소스이다. Gradient Boosting은 단순한 모델 세트의 추정치를 결합하여 정확하게 대상 변수를 예측하는 학습 알고리즘이다. 다양한 데이터 관계, 유형 및 분산을 처리하기 때문이며 결과에 개선된 최적화 및 수정할 수 있는 다수의 Hyper Parameter 때문이며, 이러한 유연성 때문에 Light GBM은 관련 문제에 있어 안정적인 선정이 된다고 본다.

본 논문의 구조는 다음과 같다. 2장에서는 부스팅과 데이터 이상치 탐지, LightGBM의 개념과 성능 비교에 대한 연구를 기술하고, 3장에서는 실제 불균형 데이터를 사용하여 비교하는 실험을 진행한다. 4장에서는 결론에 대해 요약하고, 끝으로 향후 연구 방향을 제시한다.

2. 본 론

2.1 부스팅(Boosting) 개요

랜덤 포레스트나 배깅과 같은 기법과 부스팅이 다른 건 기존에 있는 예측기를 합한다는 것이다. 랜덤 포레스트는 병렬로 결정 트리를 다양하게 동시에 작업한다면 부스팅은 점차적으로 디시전 트리를 상승시킨 뒤에 통합하는 작업을 거친다는 것이다.

부스팅의 방향은 크게 두 가지가 있다. 중요 데이터에 weight를 주는 adaboost 방식과 Fig. 1의 그림과 같이 부스팅 의사 결정 트리로 GBDT(Gradient Boosting Decision Tree)[9]와 같이 딥러닝의 loss function 정답과 오답의 차이를 재훈련하여 Gradient를 이용해서 모델을 개선해서 여러 개의 트리를 만드는 방식이다. XGboost나 Light GBM이 여기에 속한다.

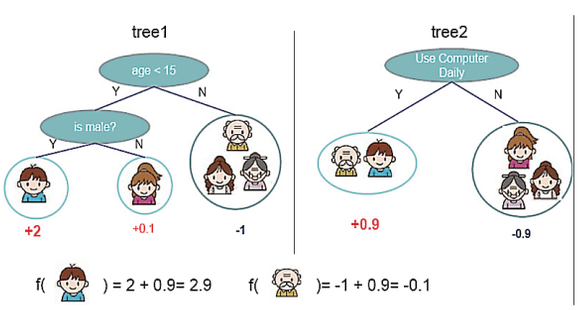

Fig. 1. Example of the Model(GBDT)

Gradient는 정답에 대한 경사라고 볼 수 있다. 정답이 1인데 0이라고 한다면 오답인 0을 보정하기 위해 모델 파라미터를 조절하는 과정은 loss를 줄이는 경사진 곳으로 이동하는 것과 같다. 차이를 한 번에 하는 건 다른 결괏값으로 나올 수 있으니 learning rate으로 조절한다.

트리 분기를 구성한다면 최대한 분기를 했을 때 각각의 분기 데이터들이 이질적이여만 잘 분기했다고 할 수 있다. 예를 들어 고객의 성별을 구별할 때 머리 길이라는 힌트가 있다면 분기문을 만들면 서로 이질적으로 구별할 수 있다.

2.2 데이터 이상치 탐지

일반적인 데이터 패턴과 다르게 이상한 패턴을 가지고 있는 데이터를 이상치(outlier)라 한다. 일반적으로 샘플링한 데이터가 패턴이 관측된 범위에서 아주 작은 값 또는 큰 값의 데이터는 모델에서 의사결정을 하는데 큰 영향을 주기에 이상치 탐지와 처리는 데이터 패턴을 보아야 한다.

Fig. 2는 IQR(Inter Qunatile Range)로 사분위 값을 이용하며, boxplot으로 볼 수 있다. low extreme보다 적은 값과 upper extreme보다 큰 값은 이상치이다.[10]

Fig. 3은 이상치 탐지를 위해 정규 분포에 대한 확률밀도함수[11] (PDF; Probability Density Function)를 따르는 확률변수 x는 평균(μ)이 0이고, 표준편차(σ) 분산이 1인 표준정규분포의 확률밀도함수를 나타낸다. 이는 평균인 0을 중심으로 좌우 대칭을 이뤄 모델의 가중치를 범위 내에 둘 수 있다.

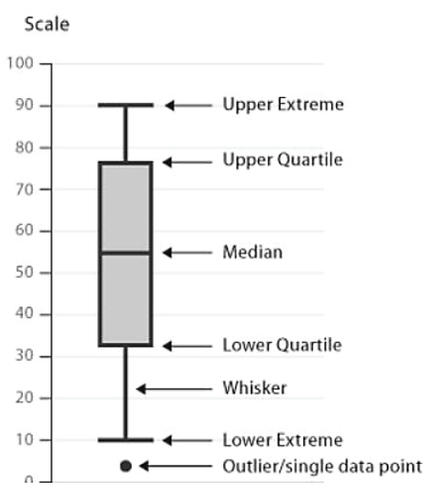

Fig. 2. Different Parts of a Boxplot

$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$$

Fig. 3. PDF for a Normal Distribution

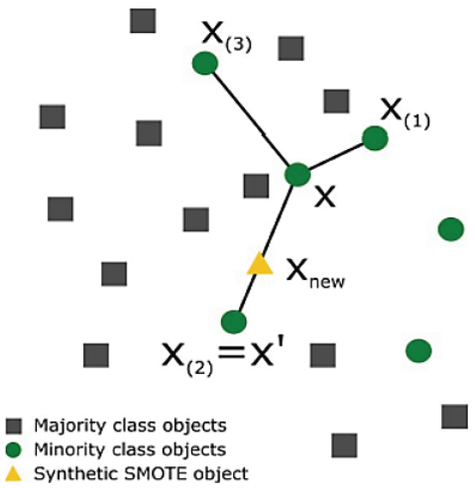

Fig. 4. Generation of X_{new} using SMOTE

SMOTE(Synthetic Minority Oversampling Technique)는 데이터 세트의 사례 수를 균형 있게 늘릴 수 있는 통계 기법이다. 이 모듈이 새 인스턴스를 기존 소수 사례 생성해서 작동하며 SMOTE는 다수 사례 수에서 변경하지 않고 구현한다. SMOTE는 입력으로 전체 데이터 세트를 사용하고 백분율은 소수 사례에서만 늘린다. SMOTE 모듈을 불균형 데이터 세트에 연결한다. 데이터 세트가 불균형해지는 건 데이터를 수집하기 어렵거나 범주를 채우기 드물기 때문이다. 대표값이 부족한 클래스를 분석할 때 SMOTE를 사용한다[12].

새 인스턴스는 기존 소수 사례의 복사본이다. 알고리즘이 대상 클래스와 인접한 항목의 공간에서 샘플을 사용하고 항목의 기능과 결합하는 새 예제를 생성하므로 클래스에 기능을 사용할 수 있게 증가하며 샘플이 일반적이게 된다.

Fig. 4는 소수 클래스에 속하는 데이터 샘플과 가까운 소수 클래스의 데이터 샘플을 KNN 알고리즘으로 찾은 후, 보간을 이용하여 합성 샘플을 새롭게 생성하는 방식이다. K 개의 최근접 이웃 샘플들 중 임의의 샘플을 랜덤하게 선택하고 (xzi) 새 객체 $Xnew = ax + (1 - a)x'$를 합성한다. 여기서 a는 [0, 1]에 대한 균등 분포의 랜덤 변수이다[13].

$$xnew = x\ I+\ \lambda\ \times\ (xz\ I-\ xi),\ \lambda\ \in\ [0,\ 1]$$

N 개의 합성 샘플들을 샘플 포인트 xi 와 xzi 사이에서 랜덤하게 생성한다.

2.3 Light GBM

Light GBM은 그라데이션 부스팅(Gradient Boosting) 프레임워크이다. Fig. 5의 Tree 기반 학습으로서 기존의 Tree 기반 알고리즘은 Tree가 수직으로 확장되나 Light GbM은 Tree가 수평적으로 확장된다. 다시 말하면 Light GBM은 leaf-wise이고, 타 알고리즘은 level-wise 이다. 확장하기 위

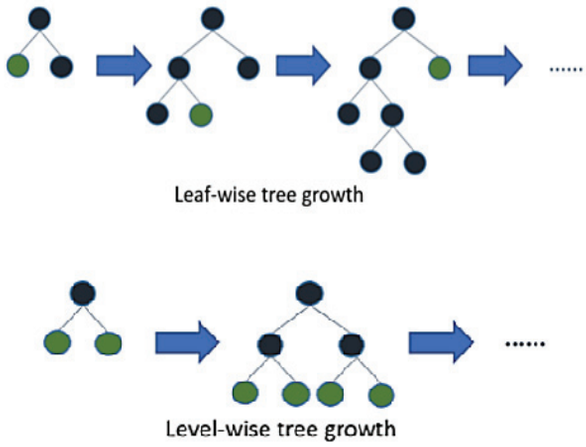

Fig. 5. Lefe-wise Tree & Level-wise Tree Diagram

해서는 max delta loss의 leaf를 선택한다. 동일한 범주에서 leaf를 확장한다면 leaf-wise(리프 트리 분할)은 level-wise (균형 트리 분할) 보다 손실을 많이 줄일 수 있다. Tree 기반 알고리즘의 수평적 확장으로 Light GBM의 leaf-wise와 타 알고리즘의 level-wise 구현을 나타낸다.

Light GBM은 민감하고 작은 데이터에 대해서 과적합하는 방법으로 효과가 있다. 복잡한 것은 파라미터 튜닝이다. Light GBM은 100개 이상의 매개 변수를 커버하기 때문이다. 파라미터는 최적의 값을 결정하기 중요하며 모델 정확도를 향상시키기 위해 파라미터 튜닝은 필요하다. Tree 모델은 num_leaves 의 값은 2 ^ 최대 깊이(max_depth) 값 보다 같거나 적어야 한다. 이것보다 많은 값은 과적합을 유발할 수 있다. 값을 크게 세팅하는 건 Tree가 깊게 확장하는 건 막을 수 있지만 언더 피팅이 발생할 수도 있다. 명확하게 Tree 깊이를 제한하기 위해 값을 max_depth 에 설정할 수 있다.

Light GBM의 장점은 학습하는데 걸리는 시간이 적고, 메모리 사용량이 적은 편이며, 카테고리형 피처들의 자동 변환과 최적 분할을 할 수 있다. 단점은 균형을 맞추기 위한 시간이 필요하며, 적은 데이터에 사용할 경우 과적합 가능성이 크다.

2.4 성능 비교

GBDT(Gradient Boosting Decision Tree)에는 Gradient가 있고 weight는 없다. 따라서 데이터 수를 줄여서 계산할 때, Gradient가 적은 것만 랜덤하게 drop 하므로 one-side sampling이라 한다.

Fig. 6의 Gradient는 잔차를 이용한다. 작은 Gradient의 drop된 부분이 정확도가 낮아지는 건 데이터 분포의 왜곡된 상태에서 훈련하기 때문인데, 작은 Gradient의 값들을 버린 샘플만큼 수를 맞춰주려면 1 - a / b를 곱해야 한다. (a는 탑 n 개의 비율, b는 샘플링 n 비율) 큰 Gradient는 절대값으로 랭크를 시키고 작은 Gradient는 작은 비율만 선택한다. 분기

$$\tilde{V}_j(d) = \frac{1}{n}\left(\frac{(\sum_{x_i \in A_l} g_i + \frac{1-a}{b}\sum_{x_i \in B_l} g_i)^2}{n_l^j(d)} + \frac{(\sum_{x_i \in A_r} g_i + \frac{1-a}{b}\sum_{x_i \in B_r} g_i)^2}{n_r^j(d)}\right),$$

Fig. 6. Algorithm that Gives an Variance Calculation Weight

를 계산할 때는 분산을 계산하게 되는데 적은 데이터의 Gradient는 샘플링이 되었으므로 1 - a / b 를 반영해 주어 적어진 비율만큼 다시 분산을 계산하고 큰 쪽으로 정보 분기를 한다. 알고리즘은 모델로 일단 예측하고, 실제 값과의 error로 loss를 계산한다. loss대로 정렬한 후, 상위 N 개를 뽑아 topSet에 저장한다. 전체 데이터 셋을 100이라 했을 때 그중에 30개를 선택했다면 a는 0.3이고, 남은 70개 중 10개만 랜덤 샘플링하여 b는 0.1이다. 전체 데이터 대비 30%의 Big Gradient + 전체 데이터 대비 10%의 Small Gradient로 다시 샘플링 된다. 줄어든 Small Gradien에 대해 weight를 (1-a/b) 적용한 후 데이터를 "전체 데이터 셋 + LOSS + WEIGHT" 를 예측기를 데이터 셋으로 만들어 추가를 전체 예측기 셋에 한다. 정확도에서 데이터가 많았을 때 수학적으로 해치지 않음을 증명해 내고 있다. 실질적으로 보통의 competition에 Light GBM은 배깅과 맞닿아서 랜덤 샘플링하는 방법으로 다양성을 높여 generalization이 우수한 정확도를 보인다.

Table 1은 Light GBM의 계산 속도의 벤치마킹이고 Table 2는 LIght GBM의 정확도의 벤치마킹이다. xgb_his (히스토그램 기반 알고리즘), xgb_exa(사전 정렬 알고리즘), lgb_baseline과 비교한 것이다. 이 벤치마킹은 마이크로소프트의 Light GBM에 대한 정확도이다. 알고리즘에 대해 실험한 결과는 공개되어 이용 가능한 데이터 셋 5가지이며, 그중에 마이크로소프트 Learning to Rank(LETOR)의 데이터 셋은 WEB 검색 질의는 30,000개를 내포하고 있다. 사용된 변수는 대부분이 수치형 밀집 변수이며, Allstate Insurance Claim과 Flight Delay 데이터 셋은 One-Hot Encoding 된 변수를 내포하고 있다. KDD CUP 2010과 KDD CUP 2012에서 추출한 것은 마지막 두 데이터 셋이다. 이런 데이터 셋은 큰 크기이면서 희소 변수 및 밀집 모두 포함하므로 현실의 분석 과제를 대신할 수 있다고 본다[14].

계산 속도는 Light GBM이 처리속도가 월등하다.

Table 1. Benchmarking the Calculation Speed

	xgb_exa	xgv_his	lgb_baseline	Light GBM
Allstate	10.85	2.63	6.07	0.28
Flight Delay	5.94	1.05	1.39	0.22
LETOR	5.55	0.63	0.49	0.31
KDD10	108.27	OOM	39.85	2.85
KDD12	191.99	OOM	168.26	12.67

Table 2. Accuracy Benchmarking

	xgb_exa	xgv_his	lgb_base line	Light GBM
Allstate	0.6070	0.6089	0.6093	0.6093 ± 9e-5
Flight Delay	0.7601	0.7840	0.7847	0.7846 ± 4e-5
LETOR	0.4977	0.4982	0.5277	0.5275 ± 5e-4
KDD10	0.7796	OOM	0.78735	0.78732 ± 1e-4
KDD12	0.7029	OOM	0.7049	0.7051 ± 5e-5

정확도는 xgb_exa가 높다.

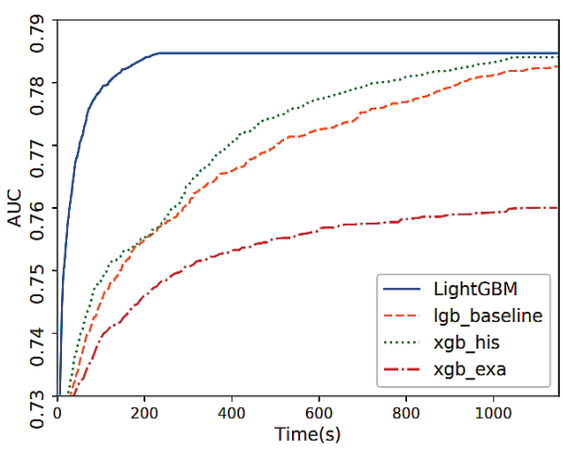

Fig. 7. Visualizing Benchmarking

Fig. 7은 Light GBM, lgb_baseline, xgb_his(히스토그램 기반 알고리즘), xgb_exa(사전 정렬 알고리즘)를 벤치마킹하여 시각화한 것이다.

3. 실 험

3.1 실험 방법

실험은 Kaggle에서 제공하는 credit card fraud를 이용하였다.

Table 3의 데이터 셋은 2013년 9월 유럽 카드 소지자의 신용 카드 거래 내용이 들어 있으며 총 거래 284,,807 건에서 492 건이 이틀 동안 사기가 발생한 거래를 보여준다. 이 데이터 세트는 불균형적이며 모든 거래의 0.173%를 차지한다. Class 변수는 사기행위면 1, 정상이면 0을 사용한다.

Table 3. Experimental Data Set

	Data	Percentage(%)	Class
Total	284,807	100%	All
Non-Fraud	284,315	99.827%	0
Fraud	492	0.173%	1

데이터의 구성은 28만여 개의 row 데이터와 31개의 feature를 가지고 있다. 고객의 정보가 포함된 데이터(V1~V28)는 28개 feature는 숨겨져 있으므로 Time, Amount, Class 3가지의 feature로 구현한다.

Fig. 8에서 보는 것과 같이 신용카드의 대부분은 정상 데이터이고 사기 건수는 몇 개 없기 때문에 불균형 데이터로서 균형이 이뤄지지 않은 상태이다. 실험은 3가지의 방법으로 진행하고자 한다. 첫째 원래의 상태에서 머신러닝 모델 검출과 둘째, 데이터를 전처리하여 중복이나 누락 없이 모델을 구축하고 데이터 내에서 이상치 검출을 IQR 방법으로 검출하여 제거한다. 셋째, 불균형 데이터 접근 방식으로 SMOTE으로 머신러닝으로 성능 측정을 하여 Logistic Regression [15], Decision Tree[16], KNN[17], Random Forest[18], XGBoost 알고리즘[19] 등과 비교하여 Light GBM의 알고리즘의 leaf-wise로 수직적인 확장된 알고리즘이 더 많은 loss를 줄일 수 있다는 걸 구현한다.

원본 데이터를 전처리 하기 전 6가지의 알고리즘으로 신용카드 사기를 검출한다. Table 4에서 Light GBM의 경우 정확도는 0.99, 정밀도는 0.95, 재현율은 0.82, F1-score는 0.88, AUC는 0.91 정도의 결과가 나오며, 다른 알고리즘보다 성능이 비슷하거나 우수하다.

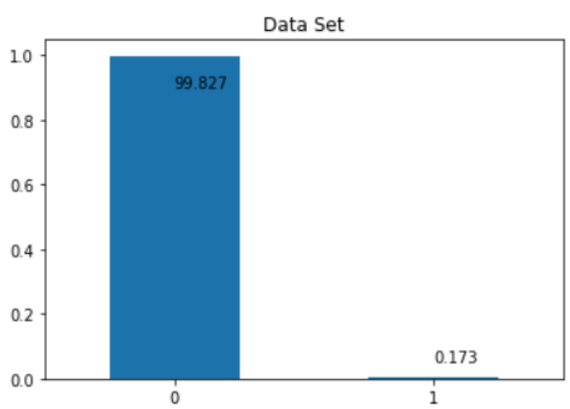

Fig. 8. Data Set Structure

Table 4. Comparison of ML Algorithms before Preprocessing

	Accuracy	Precision	Recall	F1-score	AUC
Decision Tree	0.9994	0.8962	0.7851	0.8370	0.8924
KNN	0.9983	0.8888	0.0661	0.1230	0.5330
Logistic Regression	0.9989	0.6936	0.6363	0.6637	0.8179
Random Forest	0.9993	0.9230	0.6942	0.7924	0.8470
XGBoost	0.9995	0.9339	0.8181	0.8722	0.9090
Light GBM	0.9996	0.9523	0.8264	0.8849	0.9131

Fig. 9는 데이터 정규화와 데이터 이상치 탐지를 한다. 신용카드 사기 여부에 연관성 비교를 고객의 정보가 담긴 V1~V28 중에서 V14와 v17이 음의 연관성을 보인다. 이상치 제거는 IQR을 이용하여 최대값과 최소값을 구하여 제거한다.

Fig. 10은 IQR을 계산하고 그 이상치를 탐지 제거한 후 Q1 지점 데이터를 가져온다. Q3을 구해서 Q3에서 Q1을 감산하면 IQR를 구할 수 있다. 여기에 1.5를 곱셈한 뒤 Q1에서 가감해 주고(최소값) Q3에 더한다. (최대값) 마지막으로 최대값보다 큰 데이터 또는 최소값보다 적은 데이터 index를 drop 한다. 이상치 데이터가 13,737개가 있다.

이상치 데이터를 제거한 뒤의 성능 결과는 Table 5이다.

Table 4와 Table 5의 알고리즘 성능을 분석한 결과 성능은 Class에 대한 균형을 이루었기 때문에 수치는 향상되었다.

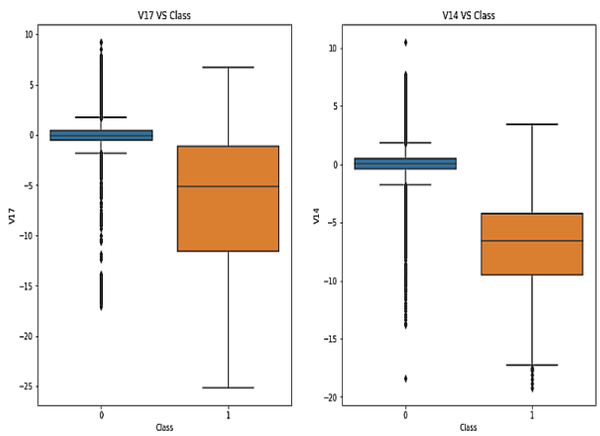

Fig. 9. The distribution of Class vs V17 and Class vs V14

```
data_copy = df.copy()
data_copy = remove_outlier(data_copy, 'V14')

13737
(269989, 31)
```

Fig. 10. Distribution of Class (V17 & V14)

Table 5. IQR ML Algorithm Comparison

	Accuracy	Precision	Recall	F1-score	AUC
Decision Tree	0.9997	0.9904	0.8813	0.9327	0.9406
KNN	0.9996	0.9795	0.8135	0.8888	0.9067
Logistic Regression	0.9996	0.9898	0.8305	0.9032	0.9152
Random Forest	0.9997	1.0	0.8728	0.9321	0.9364
XGBoost	0.9997	1.0	0.8813	0.9369	0.9406
Light GBM	0.9997	0.9900	0.8474	0.9132	0.9237

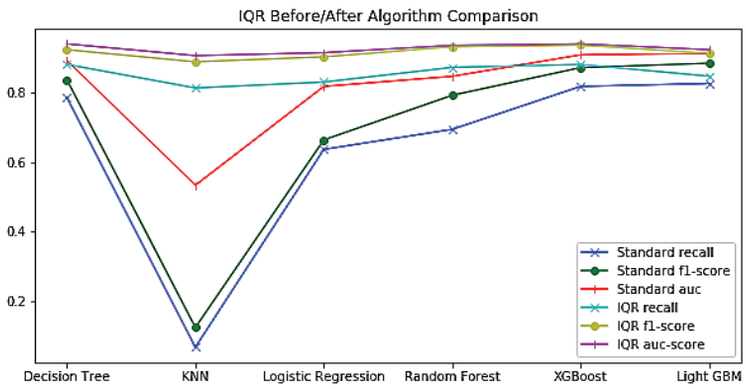

Fig. 11. IQR Before/After Algorithm Comparison

Fig. 11은 정규화 전과 IQR의 재현율과 F1-score, AUC에 대한 시각화이다. 예측 성능을 높이기 위해 데이터 샘플과 가장 가까운 소수 범주의 데이터 샘플들을 보간(interpolation)하여 새로운 합성(synthetic) 샘플을 생성하여 Light GBM 알고리즘으로 과적합이 없는 방법으로 정확성 향상과 높은 유연성 처리 과정을 기존 알고리즘과 분류 보고서를 이용하여 실험 결과를 예측하고자 한다.

3.2 실험 결과

SMOTE의 방식은 클래스의 표본을 데이터에서 개수가 적은 것 중에서 새로운 샘플에 임의의 값을 추가하여 데이터에 추가하는 오버 샘플링 방식이며 Table 6은 SMOTE 머신러닝 알고리즘 개별 성능 결과이다.

Table 5의 이상치를 제거한 성능 결과 데이터와 Table 6의 SMOTE 성능 결과를 비교하면 Decision Tree, KNN, Logistic Regression, XGBoost 모델은 정밀도, F1-score 급감했고 Recall과 AUC는 향상되었다. Random Forest 모델과 Light GBM 모델은 정밀도가 약간 감소, 재현율과 AUC는 증가하였다.

Lightgbm은 “is_unbalance”와 “scale_pos_weight”의 매개 변수 튜닝을 통해 성능을 높일 수 있으나 SMOTE를 통해 데이터 셋으로 성능이 가장 좋은 건 오버 샘플링한 모델이란 것을 확인할 수 있다.

Table 6. Compare the Smote ML Algorithm

	Accuracy	Precision	Recall	F1-score	AUC
Decision Tree	0.9889	0.1276	0.9152	0.2240	0.9521
KNN	0.9985	0.5483	0.8644	0.6710	0.9315
Logistic Regression	0.9765	0.0642	0.9152	0.1200	0.9459
Random Forest	0.9994	0.8076	0.8898	0.8467	0.9447
XGBoost	0.9944	0.2281	0.9067	0.3645	0.9507
Light GBM	0.9997	0.9716	0.8728	0.9196	0.9364

Table 7. Recall ML Algorithm Comparison

Machine Learning Model		Recall
Decision Tree	Standard	0.7851
	IQR	0.8813
	SMOTE	0.9152
KNN	Standard	0.0661
	IQR	0.8135
	SMOTE	0.8644
Logistic Regression	Standard	0.6363
	IQR	0.8305
	SMOTE	0.9152
Random Forest	Standard	0.6942
	IQR	0.8728
	SMOTE	0.8898
XGBoost	Standard	0.8181
	IQR	0.8813
	SMOTE	0.9067
Light GBM	Standard	0.8264
	IQR	0.8474
	SMOTE	0.8728

credit card fraud 데이터에 대한 SMOTE를 이용하여 예측 알고리즘으로 분석한 결과 원래의 상태(Stardard)와 데이터 이상치 탐지 제거(IQR)와 불균형 데이터 처리(SMOTE)에서 성능이 개선되는 걸 Table 7의 재현율로 분석하였다. 각 모델에서 이상치 탐지 제거에서 불균형 데이터 처리를 한 성능은 Decision Tree 모델 3%, KNN 모델 5%, Logistic Regression 모델 8%, Random Forest 모델 1%, XGBoost 모델 2%, Light GBM 모델 3%의 향상이 되었다.

Fig. 12는 정규화 전과 이상치 탐지 제거 후 IQR 성능치와 SMOTE로 오버 샘플링한 모델의 시각화를 나타내어 보여준다.

이 연구의 Light GBM의 알고리즘은 성능이 대부분 높게 나왔으며, Light GBM과 XGBoost는 Logistic Regression, Decision Tree, KNN, Random Forest와 비교하여 분류

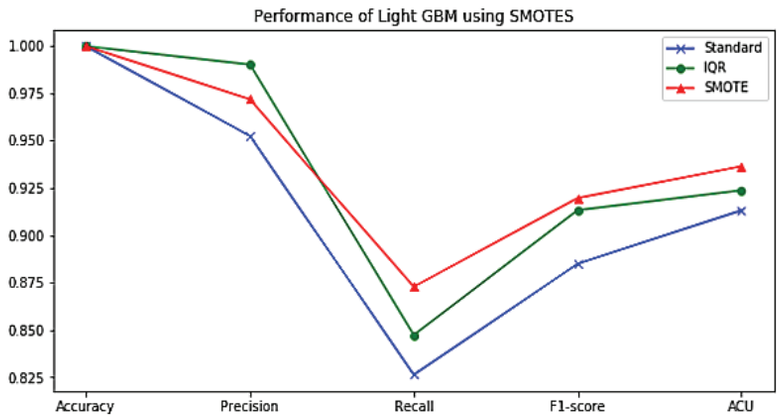

Fig. 12. IQR Before/After Algorithm Comparison

정확도가 높다. 데이터 클래스가 데이터 불균형이 기형일수록 Light GBM의 안정성과 정확성 면에서 적합하다는 것을 알 수 있었다.

이 실험 결과는 SMOTE를 이용하여 LIght GBM 모델이 데이터 불균형 해결에 6개의 다른 알고리즘보다 평균적으로 향상됨을 확인했다.

4. 결 론

본 연구에서는 Logistic Regression, Decision Tree, KNN, Random Fores, XGBoost Light GBM 모델을 동일한 표본으로 수행하였다. 각 모델들을 변수 간의 새로운 종속성을 찾아 분석을 수행하여 시각화하여 분류 문제는 어느 정도는 해결했다고 본다. LIght GBM은 데이터의 예측을 테스트하고 튜닝하여 최적의 성능을 발휘했으며, 높은 정확도의 데이터 품질은 처리 시간 단축에 기여한다고 본다.

향후 연구로는 Catboost의 Ordered Boosting 기법을 활용하여 Prediction Shift 문제를 해결하려 한다. GBM (Gradient Boosting Machine)은 새로운 Tree를 다음 스텝에 만들 때 모델의 사용된 데이터를 Gradient estimate를 다시 사용되기 때문에 과적합에 취약하다고 본다. .

이러한 문제를 기존의 tree 구조를 Internal node의 cut-off value 방식으로 선택 후 leaf value는 역순으로 구하고 ordered principle 개념을 적용해 tree 구조를 선택하는 과정을 해결하고자 한다.

References

[1] R. Leuning, E. Van Gorsel, W. J. Massman, and P. R. Isaac, "Reflections on the surface energy imbalance problem," *Agricultural and Forest Meteorology*, Vol.156, pp.65-74, 2012.

[2] M. Artís, M. Ayuso, and M. Guillén, "Detection of automobile insurance fraud with discrete choice models and misclassified claims," *Journal of Risk and Insurance*, Vol.69, No.3, pp.325-340, 2002.

[3] D. F. Stroup, G. D. Williamson, J. L. Herndon, and J. M. Karon, "Detection of aberrations in the occurrence of notifiable diseases surveillance data. Statistics in medicine," Vol.8, No.3, pp.323-329, 1989.

[4] T. Y. Liu, "Easyensemble and feature selection for imbalance data sets," In *2009 international Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing*, IEEE, pp.517-520, 2009.

[5] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, Vol.16, pp.321-357, 2002.

[6] S. Hasmita, F. Nhita, D. Saepudin, and A. Aditsania, "Chili commodity price forecasting in bandung regency using the adaptive synthetic sampling (ADASYN) and k-nearest neighbor (KNN) algorithms," In *2019 International Conference on Information and Communications Technology (ICOIACT)*, IEEE, pp.434-438, 2019.

[7] C. XGBoost, S. N. L. P. LightGBM, and B. Quinto, "Next-Generation Machine Learning with Spark," 2020.

[8] C. Sammut and G. I. Webb, (Eds.). "Encyclopedia of machine learning," Springer Science and Business Media, 2011.

[9] V. Lalchand, "Extracting more from boosted decision trees: A high energy physics case study," *arXiv preprint arXiv: 2001.06033*, 2020.

[10] D. Zwillinger, "CRC standard mathematical tables and formulas," Chapman and Hall/CRC, 2018.

[11] AP Statistics Review - "Density Curves and the Normal Distributions," Archived from the original on 2 April 2015. Retrieved 16 March 2015.

[12] L. Torgo, R. P. Ribeiro, B. Pfahringer, and P. Branco, Smote for regression, In *Portuguese Conference on Artificial Intelligence*, Springer, Berlin, Heidelberg, pp.378-389, 2013.
[13] N. Kozlovskaia and A. Zaytsev, "Deep ensembles for imbalanced classification," In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, IEEE, pp.908-913, 2017.
[14] D. Wang, Y. Zhang, and Y. Zhao, "LightGBM: An effective miRNA classification method in breast cancer patients," In *Proceedings of the 2017 International Conference on Computational Biology and Bioinformatics*, pp.7-11, 2017.
[15] D. W. Hosmer and S. Lemeshow, "Applied Logistic Regression," John Wiley & Sons. New York, 2000.
[16] R. Kohavi and R. Quinlan, "Decision tree discovery handbook of data mining and knowledge discovery," 2002.
[17] E. Mirkes, "KNN and Potential Energy (Applet)," University of Leicester, 2011.
[18] R. Zhu, D. Zeng, and M. R. Kosorok, "Reinforcement learning trees," *Journal of the American Statistical Association*, Vol.110, No.512, pp.1770-1784, 2015.
[19] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.785-794, 2016.

한 영 진

https://orcid.org/0000-0003-1014-166X
e-mail : sni94@hanyang.ac.kr
1989년 GENERAL BBS(전자게시판) HQ/Director
1991년 ~ 1993년 한국교육진흥원 전산개발실 Manager
1994년 ~ 1999년 SOFTCOM CEO
1999년 ~ 현 재 에스엔아이(주) CEO
2020년 한양대학교 컴퓨터공학(석사)
2022년 한양대학교 컴퓨터소프트웨어학과(박사수료)
2021년 ~ 현 재 한국외국어대학교 GBT학부 외래교수
2022년 ~ 현 재 이화여자대학교 소프트웨어학부 외래교수
2022년 ~ 현 재 가톨릭대학교 데이터사이언스학과 외래교수
관심분야 : 딥러닝, IoT, 정보보안, 이미지 프로세싱, 네트워크 기반 제어

조 인 휘

https://orcid.org/0000-0002-8435-0395
e-mail : iwjoe@hanyang.ac.kr
1983년 한양대학교 전자공학과(학사)
1995년 University of Arizona 컴퓨터공학(석사)
1998년 Georgia Tech 컴퓨터공학(박사)
1985년 ~ 1992년 (주)데이콤 종합연구소 선임연구원
1998년 ~ 2000년 Oak Ridge 국립연구소 연구원
2000년 ~ 2002년 Bellcore Lab(Telcordia) Scientist
2002년 ~ 현 재 한양대학교 컴퓨터소프트웨어학부 교수
관심분야 : 이동통신, IoT, 딥러닝, XAI, 임베디드 시스템, EV 배터리 및 시뮬레이션