경영전문석사학위 논문

불균형 데이터 분류분석을 위한 Gaussian Mixture Model과 CTGAN을 이용한 오버샘플링 기법에 관한 연구

2023년 2월

서울과학종합대학원대학교

양 문 일

경영전문석사학위 논문

불균형 데이터 분류분석을 위한 Gaussian Mixture Model과 CTGAN을 이용한 오버샘플링 기법에 관한 연구

2023년 2월

서울과학종합대학원대학교

양 문 일

불균형 데이터 분류분석을 위한 Gaussian Mixture Model과 CTGAN을 이용한 오버샘플링 기법에 관한 연구

지도교수 신 호 상

이 논문을 경영학 석사 학위논문으로 제출함
2023년 2월
서울과학종합대학원대학교
양 문 일

양문일의 석사 학위논문을 인준함
2023년 1월

위 원 장 김 보 영 (인)

위 원 임 효 숙 (인)

위 원 신 호 상 (인)

초 록

불균형 데이터란 라벨(Label)의 비율이 현저히 차이 나는 데이터를 뜻하며 산업 전 분야에 걸쳐 발견되고 있다. 불균형 데이터는 다수 범주 데이터가 소수 범주 데이터에 비해 과도하게 분포하고 있고, 이러한 현상은 분류 경계(Decision Boundary) 설정에 장애가 된다. 그 때문에 기계학습 분류 알고리즘의 성능을 저하하는 요인으로 작용한다. 이를 해결하기 위해 소수, 다수 데이터의 분포 차이를 완화하는 다양한 기법들이 제안되었다. 그중 오버샘플링 기법은 소수 범주의 데이터를 증폭시켜 데이터 불균형을 해소한다. 데이터를 증폭시키는 방법은 SMOTE(Synthetic Minority Oversampling Technique) 계열과 GAN(Generative Adversarial Networks) 계열이 있다. SMOTE 계열은 소수 범주 데이터와 근접 데이터를 KNN(K-Nearest Neighbor) 알고리즘을 활용하여 추출한 후, 보간하여 가상 데이터를 생성하는 방법이다. GAN 계열은 가상 데이터를 생성하는 생성자(Generator), 실제 데이터와 생성 데이터를 구별하는 판별자(Discrimonator), 두 인공신경망이 상호경쟁하며 훈련하는 데이터 증강기법(Data Augmentation)이다.

본 논문에서는 현존하는 오버샘플링 기법의 단점을 극복하기 위해

군집화 알고리즘인 가우시안 혼합 모델(Gaussian Mixture Model)과 CTGAN(Conditional Tabular GAN)을 결합한 오버샘플링 기법 G-CTGAN을 제안한다. 가우시안 혼합 모델은 전체 집단 분포에 하위 집단의 분포들이 다수의 가우시안 분포 형태로 존재한다고 가정하고, 이를 확률적으로 나타내는 모델이다. 개별 데이터마다 EM 알고리즘(Expectation-Maximization algorithm; EM algorithm)을 적용하여 하위 집단의 가우시안 분포에 속할 확률을 계산하여 할당한다. 그 다음 최대가능도 추정법(Maximum Likelihood Estimation, MLE)으로 개별 데이터들이 소속된 가우시안 분포의 모수를 추정하여 분포별로 군집을 형성한다. 이와 같이 유사한 분포로 형성된 서로 다른 군집에 CTGAN을 적용하여 SMOTE 적용시 발생하는 이상치에 따른 과적합 문제를 해소한 데이터 증강 기법이다. 우수성 증명을 위해 실제 불균형 데이터인 경기도 지방세 악성체납자 데이터를 실험에 사용한다. 데이터를 훈련 데이터와 시험 데이터로 나누고, 기존의 오버샘플링 기법과 본 논문에서 제안한 기법을 활용하여 훈련 데이터의 소수 범주 데이터를 증폭시킨 후 불균형을 완화한다. 완화된 훈련 데이터에 랜덤 포레스트(Random Forest), Light GBM, 다층 신경망(Multi-Layer Perceptron), TabNet 알고리즘을 이용하여 분류 모델을 생성하고, 시험 데이터를 이용하여 분류모델의 성능을 비교하는 방식으로 실험을 진행하였다.

AUC, $F1-Score$, 정밀도(Precision), 재현율(Recall)을 사용하여 성능을 비교하고, 가우시안 혼합 모델과 CTGAN을 결합한 기법을 사용하였을 때 분류모델의 성능이 기존 오버샘플링 기법보다 개선되었음을 확인하였다.

목 차

표 목 차

그 림 목 차

제 I 장 서 론

제 1 절 연구배경

데이터 불균형은 종속변수의 구성비가 균등하지 않아 소수(minority) 범주와 다수(majority) 범주 관측치의 비율이 현저하게 차이 나는 것을 말한다. 이러한 현상은 다양한 산업군에 존재하며, 주로 관측치의 비율이 낮은 소수 범주를 표적으로 분류 문제를 정의하는 경우가 많다. 대표적인 사례로 사기행위 적발 (Fawcett & Provost, 1997), Oil spill detection(Kubat et al., 1998), Scene classification (Yan et al., 2003) 등이 있다.

데이터 불균형 현상으로 발생하는 문제점은 다음과 같다. 첫째, 분류모델의 성능 지표에 대한 적합성 문제이다(강필성 & 조성준, 2006). 보편적으로 분류 성능 측정을 위해 활용되는 지표는 정확도(Accuracy)이며 전체 분석 대상 중 정분류된 표본의 비율로 측정된다. 그러나 데이터가 불균형한 상황에서 정확도를 성능 지표로 사용한다면 분류모델의 신뢰도에 문제가 생긴다. 예를 들어, 다수 범주와 소수 범주의 비율이 99:1인 데이터를 사용하여 이진 분류모델을 생성한다고 가정할 때, 해당 모델이 모든 데이터를 다수 범주로 분류하면 정확도가 99%가 된다. 모든 다수 범주 데이터가 정분류되었기 때문이다. 반면 소수 범주 데이터의 정분류율은 0%이다. 모두 다수 범주로 잘못 분류되었기 때문이다. 결국 분류모델의 정확도는 99%로 높은 수치지만 소수 범주 데이터를 전혀 분류하지 못해 실용성이 없는 모델을

개발하게 된다. 이러한 문제를 해결하기 위해 다수 범주와 소수 범주의 정분류율을 동시에 고려한 ROC(Receiver Operating Characteristic) 곡선과 곡선 아래 면적 값인 AUC (Area Under the ROC Curve) 지표가 정확도를 대체하여 사용되고 있다(Fawcett, 2006). 둘째, 데이터 불균형 현상은 분류모델의 성능을 저하하는 요인으로 작용한다. 불균형이 심한 경우 소수 범주 데이터에 비해 다수 범주 데이터가 과도하게 분포한다. 그 결과 소수 범주 데이터의 영역을 다수 범주 데이터가 침범하게 되어 명확한 분류 경계(Decision Boundary) 설정이 어려워지고 분류성능이 저하된다. 이러한 문제를 해결하기 위해 샘플링 방식에 따른 두 가지 기법이 제안되었다. 첫째, 언더샘플링 기법은 고안된 알고리즘에 의해 소수 범주 데이터 수만큼 다수 범주 데이터를 샘플링하여 데이터 간 분포차이를 완화한다. 다수 범주 데이터를 축소하기 때문에 계산량이 줄어 적용 시간이 짧다는 장점이 있지만 소수 범주의 비율이 너무 작은 경우 정보 손실량이 많다는 단점이 있다. 둘째, 오버샘플링 기법은 고안된 알고리즘을 적용하여 다수 범주 데이터 수만큼 소수 범주 데이터를 증폭시킨다. 정보 손실이 없다는 장점이 있지만 데이터가 증폭되어 계산량이 많고, 증폭된 소수 범주 인공데이터의 형태에 따라 분류성능이 결정된다는 단점이 있다.

오버샘플링 기법 중 대표적으로 SMOTE(Synthetic Minority Oversampling Technique)가 있다. 소수 범주 데이터와 근접 데이터를 KNN(K-Nearest Neighbor) 알고리즘을 활용하여 추출한 후, 가중치를 통해 보간하여 무작위로 가상 데이터를 생성하는 방법이다(Chawla et al., 2002; Altman 1992). 소수 범주의

인공데이터를 생성함으로써 데이터 불균형을 해소하여 분류모델의 성능을 향상했다. 그러나 생성된 인공데이터에 의한 과적합 문제와 이상치에 민감하다는 단점이 있다. 이러한 단점을 보완하기 위해 SMOTE 기법을 기초로 하여 파생된 He et al.(2008)가 제안한 ADASYN(Adaptive Synthetic Sampling), Zhang & Yang(2018)이 제안한 G-SMOTE 등 있다. G-SMOTE는 가우시안 혼합 모델(Gaussian Mixture Model, GMM)(Reynolds, 2009)을 이용하여 소수 범주 데이터의 군집을 형성후 각각의 군집에 SMOTE를 적용하는 방식이다(He et al., 2008; Zhang & Yang, 2018). 그러나 제시된 기법들 모두 과적합과 이상치에 의한 성능저하 문제는 잔존하였다.

오버샘플링 기법 중 인공신경망을 이용하여 가상의 데이터를 생성하는 GAN(Generative Adversarial Networks)이 있다(Goodfellow et al., 2020). GAN은 인공데이터를 생성하는 생성자(Generator), 실제 데이터와 생성 데이터를 구별하는 판별자(Discrimonator)가 상호경쟁하여 훈련하는 데이터 증강기법(Data Augmentation)이다. Douzas & Bacao(2018)는 실험을 통해 SMOTE를 사용했을 때 보다 GAN을 사용했을 때 분류모델의 성능이 더 우수했음을 증명하였다(Douzas & Bacao, 2018). GAN은 이상치에 민감한 SMOTE와 비교하여 더 나은 성능을 보여주었다. 그러나 단점으로 유사한 데이터만 생성하게 되는 모드 붕괴현상(Mode Collapse)이 있다(Goodfellow et al., 2020). 그 때문에 GAN에서 파생된 다양한 기법이 제시되었고, 기존의 GAN에서 발전하여 정형데이터(Tabular Data)에 효과적으로 적용 가능한 CTGAN(Conditional Tabular GAN)이 제안되었다(Xu et al., 2019).

제 2 절 연구목적

본 논문에서는 SMOTE, GAN 기법의 단점을 극복하기 위해 가우시안 혼합 모델과 CTGAN을 결합한 G-CTGAN을 제안한다. 먼저 가우시안 혼합 모델을 적용하여 변수별 다중모드 분포(Multimodal distribution)를 해소한다. 그 다음 각각의 군집에 CTGAN을 적용하여 SMOTE의 이상치에 따른 성능저하 문제와 GAN의 모드 붕괴현상을 해소하여 보다 정교한 오버샘플링을 적용하고자 한다. 제안한 기법의 우수성 증명을 위해 실제 불균형 데이터인 경기도 악성 체납자 데이터를 활용하였다. 다수의 불량체납자(Negative) 중에서 관측 기간 내에 세금을 납부한 소수의 우량체납자(Positive)를 선별하는 분류모델을 개발하였다. 기존의 오버샘플링 기법을 적용한 분류모델과 G-CTGAN을 적용하여 생성한 분류모델의 성능을 비교하였다.

제 3 절 논문 구성

본 논문의 구성은 다음과 같다. 제1장에서는 데이터 불균형 현상으로 발생하는 문제점을 제시하면서 연구의 배경과 목적, 논문 구성에 대해 설명한다. 제2장에서는 현존하는 오버샘플링 기법들과 실험에 사용된 분류 알고리즘의 이론적 배경에 대해 소개한다. 제3장에서는 본 연구에서 제안하는 G-CTGAN 오버샘플링 기법에 대한 소개와 분류성능 우수성 검증을 위한 성능 평가에 사용된 지표에 대해 소개한다. 제4장에서는 실험에 사용된 경기도

악성 체납자 데이터의 설명과 실험 설계, 샘플링을 통한 분류모델 성능을 비교한다. 마지막 제5장에서는 본 논문에서 제안하는 오버샘플링 기법이 기여하는 바와 후속 연구 진행방향을 제시하며 결론과 고찰을 서술한다.

제 Ⅱ 장 선행 연구 고찰

제 1 절 오버샘플링 방법론

(1) SMOTE(Synthetic Minority Oversampling Technique)

SMOTE(Synthetic Minority Oversampling Technique)는 대표적인 오버샘플링 기법으로 Chawla et al.(2002)에 의해 제안되었다. 해당 기법의 적용과정은 세 가지 단계로 정의할 수 있다. 첫 번째, 임의의 소수 범주 관측치를 선택 후 KNN(K-Nearest Neighbor) 알고리즘을 활용하여 해당 관측치에서 가장 가까운 k개의 이웃 관측치들을 선별한다. 두 번째, k개의 이웃 관측치 중 임의로 m개의 관측치를 선택하여 첫 번째 단계에서 임의로 선택된 관측치와 선형 관계를 형성한다. 세 번째, 형성된 선형 관계에 가중치를 곱하여 해당 위치에 새로운 인공데이터를 생성한다. 가중치는 0과 1사이에서 발생시킨 난수이다. 해당 과정을 첫 번째 단계에서 선택한 관측치를 제외하고 반복하여 오버샘플링을 진행한다(Chawla et al., 2002; Altman 1992). <그림 1>은 SMOTE 알고리즘 슈도 코드(Pseudo Code)이다.

Algorithm *SMOTE* (T, N, k)

Input: Number of minority class samples T; Amount of SMOTE $N\%$; Number of nearest neighbors k

Output: $(N/100) * T$ synthetic minority class samples

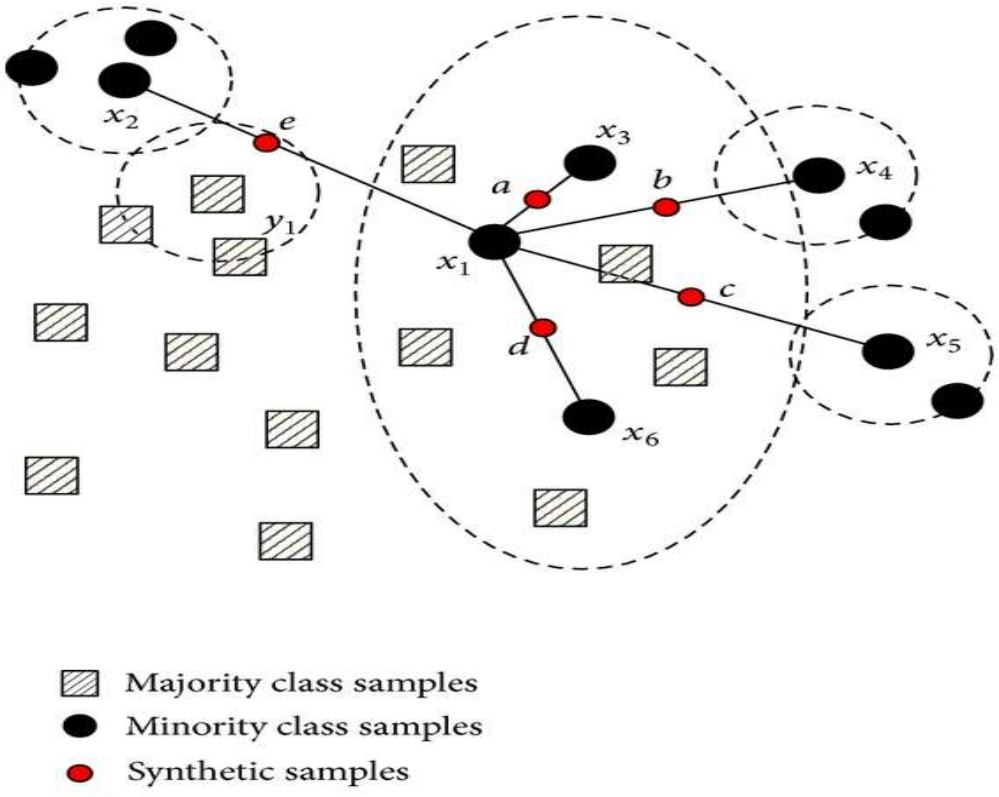
(∗ *IF N is less than 100%,randomize the minority class samples as only a random percent of them will be SMOTEd.*∗)
if $N < 100$
  then Randomize the T minority class samples
    $T = (N/100) * T$
    $N = 100$
endif
$N = (int)(N/100)$ (∗ *The amount of SMOTE is assumed to be in integral multiples of 100.* ∗)
k = Number of nearest neighbors
$numattrs$ = Number of attributes
$Sample[\,][\,]$: array for original minority class samples
$newindex$: keeps a count of number of synthetic samples generated, initialized to 0
$Synthetic[\,][\,]$: array for synthetic samples
for $i \leftarrow 1$ **to** T
    Compute *K* nearest neighbors for i, and save the indices in the *nnarray*
    Populate(*N*, *i*, *nnarray*)
endfor

Populate(*N*, *i*, *nnarray*) (∗ *Function to generate the synthetic samples.* ∗)
while $N \neq 0$
    Choose a random number between 1 and *k*, call it *nn*. This step chooses one of the *k* nearest neighbors of *i*.
    for $attr \leftarrow 1$ **to** *numattrs*
      $Compute: dif = Sample[nnarray[nn]][attr] - Sample[i][attr]$
      $Compute: gap = random\ number\ between\ 0\ and\ 1$
      $Synthetic[newindex][attr] = Sample[i][attr] + gap * dif$
    endfor
    newindex++
    $N = N - 1$
endwhile
return (∗ *End of Populate.* ∗)
$End\ of\ Pseudo - Code.$

<그림 1> SMOTE 알고리즘 슈도 코드

자료: (Chawla et al., 2002)

<그림 2>는 소수 범주 관측치 x에서 $k = 5$, $m = 5$로 정의된 SMOTE를 이용하여 인공데이터를 생성하는 과정이다. 임의의 소수 범주 관측치 x_1을 선택한 후, 5개의 이웃 관측치 x_2, x_3, x_4, x_5, x_6와 선형 관계를 형성하여 인공데이터 a, b, c, d, e가 생성되었다.



<그림 2> SMOTE($k = 5, m = 5$)

(2) ADASYN(Adaptive Synthetic Sampling)

SMOTE는 인공데이터를 임의의 개수로 생성하기 때문에 과적합(Overfitting) 현상을 발생시킬 수 있다. 이를 보완한 ADASYN이 제안되었다(He et al., 2008). ADASYN은 소수 범주 관측치에 따라 특정 규칙을 적용하여 생성할 인공데이터의 개수를 정한다. ADASYN은 다음의 단계로 적용된다. 첫 번째, 소수 범주 관측치에서 생성해야 할 인공데이터의 수 G를 정의한다. G는 다수 범주 관측치 m_l에서 소수 범주 관측치 m_s를 뺀 값에 0과 1 사이의 범위를 갖는 β 값을 곱하여 산출한다. β 값에 따라 오버샘플링 후 데이터의 라벨 범주의 비율이 결정되며, 산출식은 $G = (m_l - m_s) \cdot \beta$ 이다. 두 번째, KNN 알고리즘을 기반으로 i 번째 소수 범주 관측치에서 가장 가깝게 이웃하는 K개의 다수 범주 관측치 Δ_i개를 찾아 수식 Δ_i / K를 적용하여 r_i를 정의한다. r_i는 소수 범주 관측치 주변에 분포하는 다수 범주 관측치의 비율이다. 세 번째, 산출된 G와 i번째 소수 범주 관측치의 r_i의 곱으로 소수 범주 관측치 주변 다수 범주 관측의 비율에 따른 인공데이터 생성 개수 g_i를 결정한다.

<그림 3>은 ADASYN 적용 과정을 다수 범주 관측치 10,000개, 소수 범주 관측치 10개, 오버샘플링 후 생성하고자 하는 데이터 라벨 범주의 비율은 1:1, $k = 5$인 상황을 가정하여 작성하였다. 위와 같이 가정된 상황에 따라 $\beta = 1, G = 9900$로 정의되고, K와 각 소수 범주 관측치의 주변 다수 범주 관측치 수에 따라 생성될 인공데이터의 비율이 정해진다. 정해진 비율에 따라 생성될 인공데이터의 개수가 최종적으로 결정된다.

Algorithm ADASYN $(K = 5, \beta = 1)$

< Probability >

: $majority\ class \rightarrow Major;\ minority\ class \rightarrow Minor$

$\boldsymbol{G = (m_l - m_s) \cdot \beta}$

m_l: $The\ number\ of\ Major.sample$
m_s: $The\ number\ of\ Minor.sample$
β: $coefficient\ whose\ range\ is\ zero\ to\ unity$
G: $The\ number\ of\ sampled\ minority\ class, i.e.\ How\ many\ generate\ Minor$

$\boldsymbol{r_i = \Delta_i\ /\ K}$

$K = K\ nearest\ neighbors\ based\ on\ Euclidean\ distance$
Δ_i= $The\ number\ of\ Major.Located\ near\ i_{st}\ Minor.$

$\boldsymbol{g_i = \hat{r_i} \times G}$

$\hat{r_i}$: $normalized\ ratio$
g_i: $The\ number\ of\ Minor.Generated\ near\ i_{st}\ Minor.$

< Example >

$m_l = 10000, m_s = 10$

$If\ \beta\ =\ 1, G\ =\ 9900$

$k = 5$

$r_1 = 2/5\ , r_2 = 1/5, \dots , r_9 = 0, r_{10} = 1/5$

$\hat{r_1} = \dfrac{r_1}{r_1 + \cdots + r_{10}} = 0.1, \dots , \hat{r_{10}} = \dfrac{r_{10}}{r_1 + \cdots + r_{10}} = 0.05$

$g_1 = 999, \dots , g_{10} = 499$

<그림 3> ADASYN(K = 5, β = 1)

자료: (He et al., 2008)

(3) G-SMOTE(A GMM-based SMOTE)

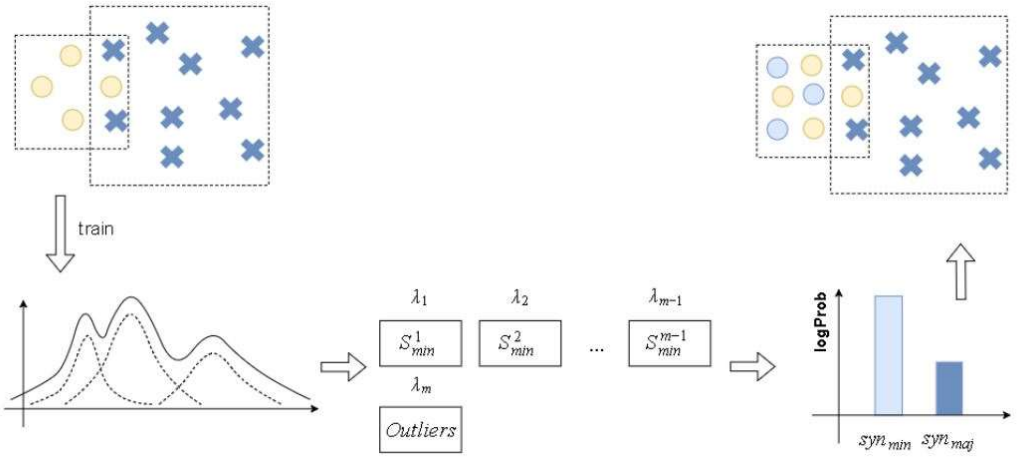
G-SMOTE는 가우시안 혼합 모델과 SMOTE가 결합된 오버샘플링 기법이다. G-SMOTE는 다음의 단계로 적용된다. 첫 번째, 소수 범주 관측치에 가우시안 혼합 모델을 적용하여 군집화를 진행한다. EM 알고리즘(Expectation-Maximization algorithm; EM algorithm)(Dempster et al., 1977)을 적용한 소수 범주 관측치에 대한 m 개의 정규분포를 <그림 4>과 같은 수식으로 정의할 수 있다(Zhang & Yang, 2018).

$$f(x \mid \mu, \Sigma) = \sum_{k=1}^{m} c_k \frac{1}{\sqrt{2\pi|\Sigma_k|}} \exp\left[(x-\mu_k)^T \Sigma^{-1} (x-\mu_k)\right]$$

<그림 4> 정규분포 m 개

자료: (Zhang & Yang, 2018)

k 번째 군집에 대한 가우시안 혼합 모델의 가중치는 c_k, 평균 벡터는 μ_k, 공분산은 $\sum_k$ 라고 할 수 있다. 두 번째, 각각의 군집에 포함된 소수 범주 관측치들의 분포와 수를 통해 생성할 인공데이터의 비율을 결정한다. 마지막으로 SMOTE를 이용하여 인공데이터를 생성한다. <그림 5>는 G-SMOTE의 전체 과정을 도식화한 것이다.



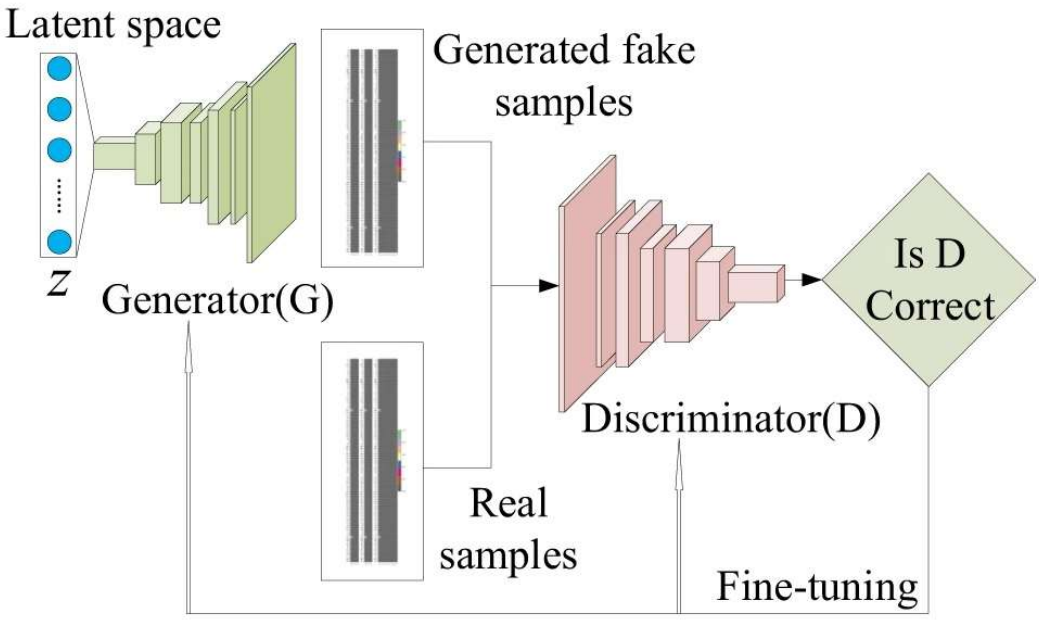
<그림 5> G-SMOTE

자료: (Zhang & Yang, 2018)

(4) GAN(Generative Adversarial Networks)

적대적 생성망이라 불리는 GAN(Generative Adversarial Networks)은 가상 데이터를 생성하는 생성자(Generator)와 실제 데이터와 생성 데이터를 구별하는 판별자(Discrimonator), 두 인공신경망이 경쟁적으로 대립하며 학습하는 데이터 증강기법(Data Augmentation)이다. 생성자의 목표는 실제 데이터를 학습하여 실제와 유사한 인공데이터를 생성하는 것이다. 생성자는 랜덤 벡터 Z를 입력 받아 인공데이터를 출력하는 함수라고 정의할 수 있다. 랜덤 벡터 Z가 존재하는 공간을 잠재 공간(Latent Space)이라 하며, 여기서 Z는 균등 분포(Uniform Distribution)나 정규 분포(Normal Distribution)에서 무작위로 추출된 임의의 값으로 존재한다. 이러한 임의의 분포를 생성하고자 하는

데이터의 분포에 매핑(Mapping)하는 것이 생성자의 역할이다. 판별자는 생성자가 만든 인공데이터와 실제 데이터를 판별하도록 학습한다. 두 인공신경망이 이와 같은 학습을 반복하여 실제 데이터와 유사한 분포를 가진 인공데이터가 생성된다(Goodfellow et al., 2020). <그림 6>은 GAN의 신경망 구조이다.



<그림 6> GAN 신경망 구조

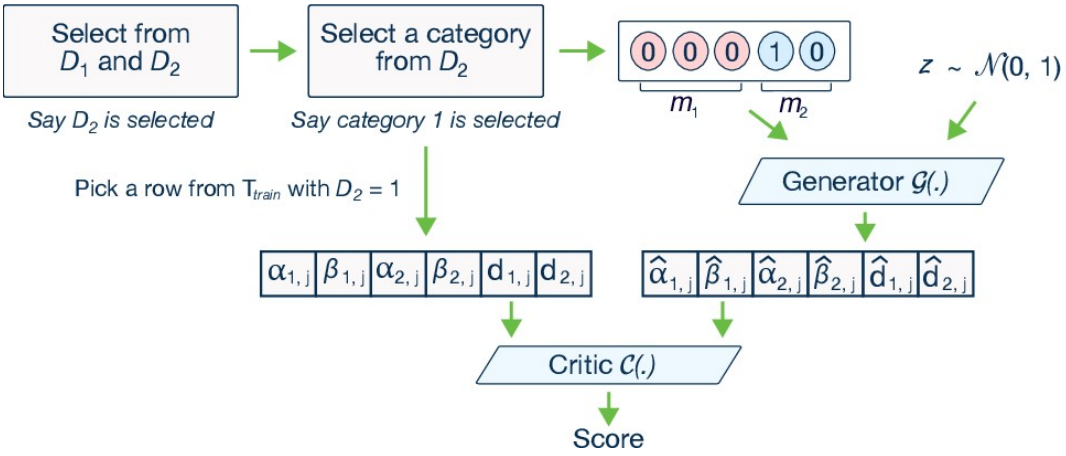
자료: (Goodfellow et al., 2020)

(5) CTGAN(Conditional Tabular GAN)

GAN은 이미지와 영상과 같은 비정형 데이터 분석을 위해 제안되었기에

다음과 같은 한계가 있다. 적용가능한 데이터의 형태 측면에서 보면 가우시안 분포가 아닌 데이터, 불균형 정도가 심한 범주형 데이터, 원 핫 인코딩(one-hot encoding)에 의해 발생된 희소(Sparse) 행렬 등 다양한 형태의 정형 데이터를 증폭하는데 제약이 있다(황철현, 2022). 또한 생성자의 편향된 학습으로 인해 다양한 인공데이터를 생성하지 못하는 모드 붕괴현상(Mode Collapse)도 존재한다(Bau et al., 2019).

이러한 단점을 해소하기 위해 정형 데이터에서도 활용가능한 CTGAN(Conditional Tabular GAN)이 제안되었다. CTGAN은 GAN 기반 아키텍처를 사용하였다. <그림 7>은 CTGAN의 신경망 구조이다.



<그림 7> CTGAN 신경망 구조

자료: (Xu et al., 2019)

CTGAN이 GAN과 차별되는 장점은 크게 두 가지이다. 첫 번째는 비가우시안(Non-Gaussian) 형태를 띄는 연속형 변수를 변동 가우스

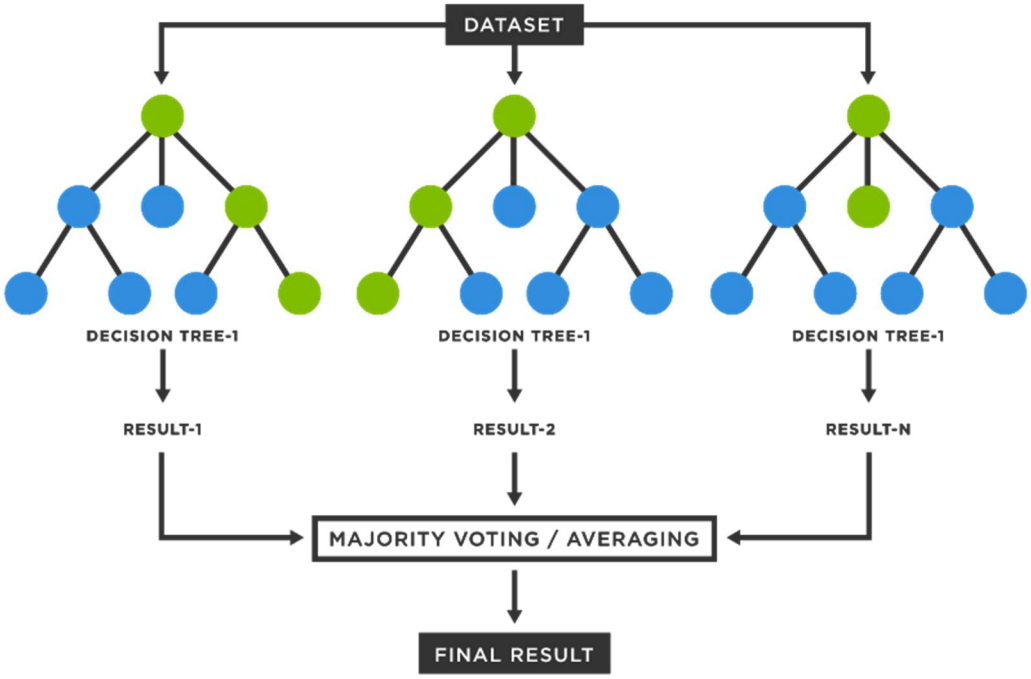
혼합모델(Variational Gaussian Mixture Model)을 단위 분포로 사용하여 각 변수의 분포 수를 추정하는 것이다. 그 때문에 가우시안 분포가 아닌 데이터의 인공데이터를 보다 정교하게 생성할 수 있다. 두 번째는 범주형 변수의 불균형 처리를 위해 조건부 벡터(Conditional Vector)를 사용한다는 점이다. GAN의 생성자는 범주형 변수중 빈도 수가 큰 데이터를 높은 확률로 생성한다. 이를 방지하고자 범주형 변수 D_1와 D_2를 (0, 0, 0), (1, 0) 인 조건부 벡터로 원 핫 인코딩한다. 그 후 범주의 로그 빈도에 따라 인공데이터 표본이 추출되어 희소 행렬 형태의 범주형 변수가 균등하게 추출되도록 한다(Xu et al., 2019).

제 2 절 분류 알고리즘

(1) 랜덤 포레스트(Random Forest)

의사결정나무(Decision Tree)(Quinlan, 1986)는 학습 데이터에 따라 분류 성능의 변동폭이 크다는 단점이 있다. 그 때문에 일반화하여 활용하기에 어려움이 있다. 이러한 단점을 극복하고자 랜덤 포레스트(Random Forest)가 제안되었다(Breiman, 2001). 랜덤 포레스트는 여러 개의 약분류기(Weak learner)를 결합하면 단일 분류기(Single learner)보다 더 좋은 분류 성능을 낼 수 있다는 가정 하에 개발된 의사결정나무 기반의 학습 알고리즘이다. 단일 분류기는 전체 데이터를 학습한 하나의 의사결정나무를 뜻하고, 약분류기는

전체 데이터중 임의의 표본을 학습한 여러 개의 의사결정나무를 뜻한다. 즉, 랜덤 포레스트는 다수의 서로 다른 의사결정나무를 임의적으로 학습하는 앙상블(Ensemble) 학습 방식인 것이다. <그림 8>은 랜덤 포레스트 알고리즘을 도식화한 것이다.



<그림 8> 랜덤 포레스트(Random Forest) 도식화

자료: (Breiman, 2001)

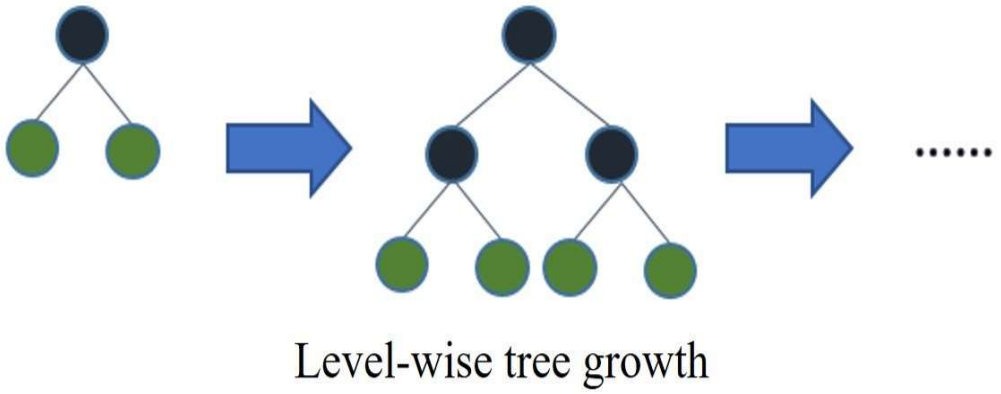
랜덤 포레스트는 크게 세 단계로 학습을 진행한다. 첫 번째 단계에서는 다수의 약분류기를 학습하기 위해 훈련 데이터에서 무작위 복원 추출을 통해 훈련 데이터 표본을 추출한다. 이러한 표본 추출 과정을

붓스트랩(Bootstrap)이라 한다. 두 번째 단계에서는 추출된 표본의 변수를 무작위로 선정하여 다수의 약분류기를 일괄적으로 생성한다. 마지막 단계에서는 생성된 약분류기들을 선형 결합하여 최종 분류기를 만든다. 선형 결합 방식은 다수결(Voting) 원칙으로 결정한다. 이러한 앙상블 과정을 베깅(Bagging, Bootstrap AGGregatING)이라 한다(Lee et al., 2020).

(2) Light GBM

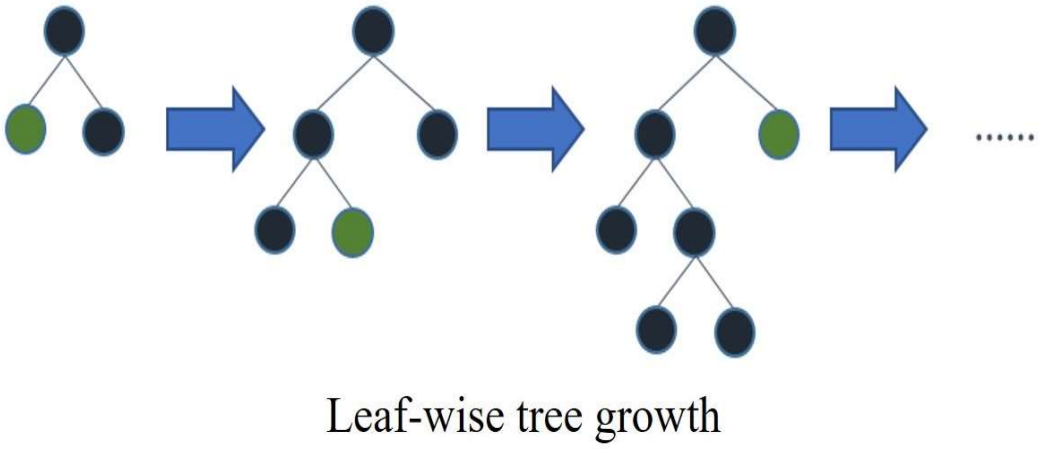
Light GBM은 그래디언트 부스팅(Gradient Boosting) 기반의 모델로 GOSS(Gradient-based One-Side Sampling)와 EFB(Exclusive Feature Bundling)를 적용해 기존 부스팅 모델의 단점인 연산속도를 개선한 모델이다. 랜덤 포레스트와는 다르게 부스팅(Boosting) 앙상블 학습방식을 사용한다. 임의의 변수를 선택하여 다양한 약분류기를 만들어내는 베깅과는 다르게 약분류기의 오차(Error)를 순차적으로 보완하여 최종 분류기를 만든다. 기존의 의사결정나무 기반 알고리즘들은 나무의 구조가 수평적으로 확장되는 균형 트리 분할(Level Wise) 방식을 사용했지만 Light GBM은 리프 중심 트리 분할(Leaf Wise) 방식으로 나무의 구조가 수직적으로 확장된다. 리프 중심 트리 분할 방식은 loss 변화가 가장 큰 노드에서 분할하는 수직 성장 방식이다. 비대칭적인 트리 구조를 생성하지만 loss 변화를 기준으로 분할하기 때문에 예측 오류를 최소화한다. 또한 균형 트리 분할 방식은 트리를 균형적으로 만들기 위한 추가 연산이 필요하지만 리프 중심 트리 분할 방식은 해당 과정이 없기에 빠르게 정답에 도달할 수 있다(Ke et al., 2017). 균형 트리

분할은 <그림 9>, 리프 중심 트리 분할은 <그림 10>에 도식화하였다.



<그림 9> 균형 트리 분할(Level Wise)

자료: (Ke et al., 2017)



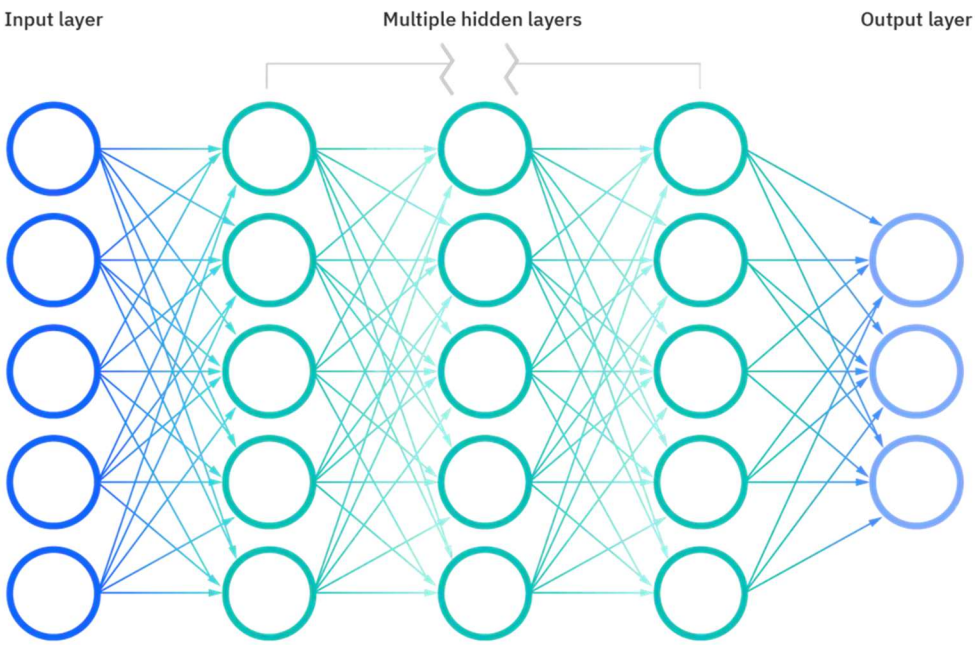
<그림 10> 리프 중심 트리 분할(Leaf Wise)

자료: (Ke et al., 2017)

(3) 다층 신경망(Multi-Layer Perceptron)

인공신경망(Artificial Neural Network)은 사람의 뇌 속 신경세포(Neuron)의 작동 원리를 본떠 컴퓨팅 시스템으로 구현한 모형으로 기계학습에 속하는 여러 알고리즘 중 하나이다. 생물학적 신경세포는 수상돌기, 축삭돌기로 구성되어있으며 다수의 신경세포가 수상돌기와 축삭돌기로 이어져 있다. 수상돌기와 축삭돌기 사이 접합 부분을 시냅스(Synapse)라 하며, 시냅스를 통해 전기적 신호의 형태로 정보가 전달된다.

인공신경망은 이를 입력노드(Input Node), 가중치(Weight), 출력노드(Output Node)의 형태로 모방하였다. 입력노드를 통해 학습 데이터가 인입되고 가중치를 적용하여 출력노드로 예측값을 출력하는 구조의 형태로 구현되었다. 입력노드들의 집합을 입력층(Input Layer), 출력노드들의 집합을 출력층(Output Layer)이라고 하며 이러한 입력층과 출력층 사이의 새로운 노드들의 집합을 은닉층(Hidden Layer)이라 한다. 은닉층의 개수에 따라 은닉층이 없는 경우는 단층 신경망(Single-Layer Perceptron)이라 하고, 1개 이상의 은닉층이 포함된 인공신경망을 다층 신경망(Multi-Layer Perceptron)이라 한다. 다층 신경망을 기반으로 은닉층을 늘린 모형을 심층 인공신경망(Deep neural network)이라 하며 이러한 학습방식을 심층학습(Deep learning)이라 한다(Hinton et al., 2006). <그림 11>은 심층 인공신경망의 기본적인 구조도이다.



<그림 11> 심층 인공신경망(Deep neural network) 기본 구조도

자료: (Hinton et al., 2006)

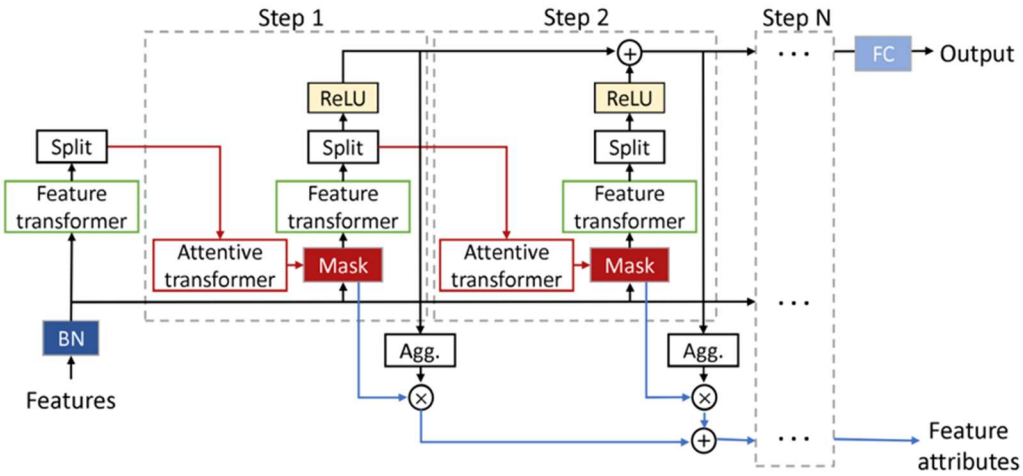
(4) TabNet

정형 데이터 분류 모델을 개발하기 위해 대부분의 경우 의사결정나무 기반의 앙상블 모델이 선호되었다. 해당 모델들은 노드의 추적을 통해 모델의 해석과 변수 중요도(Feature attributes)를 파악할 수 있다는 장점이 있기 때문이다.

TabNet(Attentive Interpretable Tabular Learning)은 이러한 의사결정나무 기반 모델의 장점인 모델 해석 기능을 가진 심층신경망 기반의 종단간 학습(end-to-

end learning) 모델이다. 기존 의사결정나무의 특징인 분류 경계(Decision Boundary)를 설정하여 예측값을 도출하는 특징과 경사 하강법(Gradient descent)을 기반으로 최적화를 통해 학습한다는 특징이 있다(Arik et al., 2021).

<그림 12>은 TabNet 알고리즘의 전체적인 인코더(Encoder) 구조이다. TabNet의 의사결정 단계는 특징변환기(Feature transformer)와 입력변수변환기(attentive transformer)로 구성되어 있다. 인입 데이터는 해당 변환기들을 거쳐 마스크(Mask)를 출력하고, 이 마스크를 활용하여 입력변수선택(Feature masking)이 이루어진다. 이 과정은 인코더마다 반복되어 변수 선택(Feature selection)을 하게 되고, 최종적으로 예측값과 변수 중요도를 산출할 수 있게 된다.



<그림 12> TabNet 인코더(Encoder) 구조도

자료: (Arik et al., 2021)

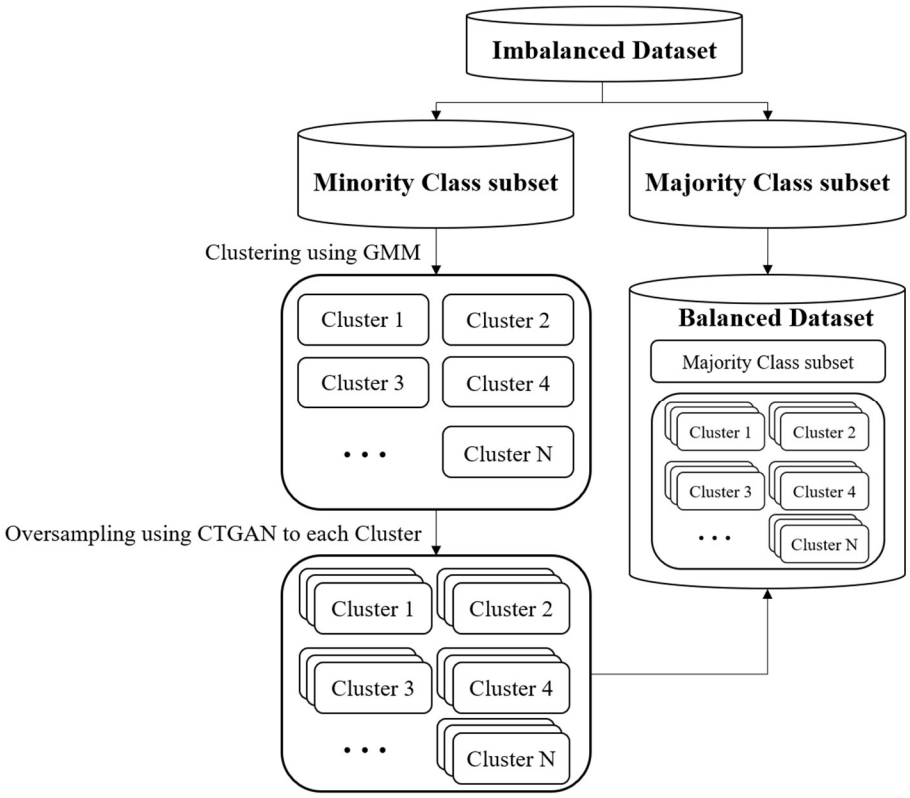
제 Ⅲ 장 GMM과 CTGAN을 이용한 복합 오버샘플링

제 1 절 G-CTGAN

본 논문에서는 데이터 불균형을 해소하고 현존하는 오버샘플링 기법들의 단점을 극복하기 위해 가우시안 혼합 모델과 CTGAN을 결합한 G-CTGAN을 제안한다. G-CTGAN의 첫 번째 단계에서는 훈련 데이터를 소수 범주 데이터와 다수 범주 데이터로 분할한다. 두 번째 단계에서는 분할된 소수 범주 데이터에 가우시안 혼합 모델을 적용하여 BIC(Bayesian Information Criterion)가 최저인 군집 수를 찾는다(Steele & Raftery, 2010). 세 번째 단계에서는 BIC가 최저인 군집 수만큼 소수 범주 데이터의 군집을 형성한 후 군집마다 CTGAN을 적용하여 인공데이터를 생성한다. 마지막 단계에서는 인공적으로 생성된 소수 범주 데이터를 결합하여 범주의 비율이 동일한 훈련 데이터를 생성한다. 이러한 과정을 통해 이상치에 의해 발생하는 데이터 분포의 왜곡과 범주의 불균형이 해소된 훈련 데이터를 생성할 수 있다. <그림 13>은 G-CTGAN의 알고리즘을 설명한 것이다.

G-CTGAN이 다른 오버샘플링 기법과 비교하여 높은 안정성을 갖는 이유는 다음 두 가지로 설명 가능하다. 첫째, 소수 범주 데이터의 독립변수에 존재하는 다중모드 분포를 가우시안 혼합 모델을 이용하여 해소한다. 가우시안 혼합 모델에 의해 형성된 군집의 독립변수는 하나의 데이터 분포를

형성하고 있을 가능성이 높기에 보다 정교한 오버샘플링이 가능하다. 둘째, 오버샘플링 적용 시 SMOTE 기법 대신 CTGAN을 사용함으로써 이상치에 의한 데이터 분포 왜곡을 방지한다. SMOTE 기법은 보간법을 이용하여 인공데이터를 무작위로 생성한다. 그 때문에 소수 범주 데이터 내 이상치가 존재할 경우, 이상치 데이터에 의해 무작위로 생성된 인공데이터는 소수 범주 데이터의 분포를 반영하지 못할 가능성이 크다. SMOTE 기법 대신 CTGAN을 적용함으로써 이상치와 유사한 인공데이터의 생성을 최소화하고, 소수 범주 데이터의 분포와 유사한 인공데이터를 생성할 수 있다.



<그림 13> G-CTGAN 알고리즘 구조도

제 2 절 성능 평가

분류모델의 성능 평가를 위해 다양한 지표들이 존재한다. 지표들을 정의하기 위해 소수 범주에 속하는 관측치를 Positive, 다수 범주에 속하는 관측치를 Negative라고 하면 혼동 행렬(Confusion matrix)을 도출할 수 있다. 혼동 행렬은 모델에 의해 예측된 범주와 실제 범주 간의 관계를 나타낸다. <표 1>은 혼동 행렬이다.

<표 1> 혼동 행렬(Confusion matrix)

		Prediction	
		Positive	**Negative**
Actual	**Positive**	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

보편적으로 활용되는 평가지표로는 정확도가 있다. 전체 관측치에서 정분류된 건의 비율로 계산되며, 산출식은 (TP + TN) / (TP + FN + TN + FP)이다. 그러나 확인된 바와 같이 데이터 불균형 하에서는 다수 범주 데이터의 정분류율에 따라 정확도가 결정된다. 이는 소수 범주를 분류하지 못하더라도 정확도는 높게 측정되기 때문에 적합한 평가지표가 될 수 없다(강필성 & 조성준, 2006).

소수 범주의 정분류율을 고려한 정밀도(Precision)와 재현율(Recall)

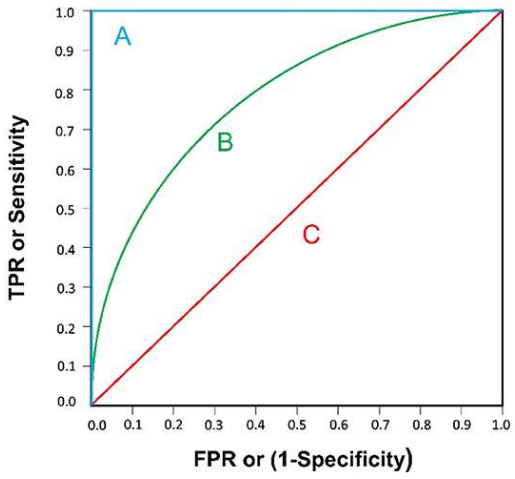
평가지표가 있다. 정밀도는 분류모델이 Positive라고 예측한 관측치들 중 Positive로 정분류한 관측치의 비율을 나타내며, 산출식은 $TP\,/\,(TP\,+\,FP)$ 이다. 재현율은 전체 Positive 관측치 중 정분류한 관측치의 비율을 나타내며, 산출식은 $TP\,/\,(TP\,+\,FN)$이다. 두 지표를 상호보완적으로 사용하기 위해 $F_\beta - Score$ 가 고안되었다. 산출식은 $(1+\beta^2)\cdot\frac{Precision\cdot Recall}{(\beta^2\cdot Precision)+Recall)}$ 이다(Rijsbergen, 1979). 그 중 보편적으로 사용되는 지표는 베타 값이 1이자 정밀도와 재현율의 조화평균인 $F_1 - Score$이다. 산출식은 $2\cdot\frac{Precision\cdot Recall}{Precision+Recall}$이다.

그 외 ROC(Receiver Operating Characteristic) 곡선과 이에 기반한 곡선 아래 면적 값인 AUC (Area Under the ROC Curve) 지표가 있다. ROC 곡선은 분류모델의 예측 결과를 민감도(Sensitivity)와 1-특이도(Specificity)로 나타낸 곡선이다. 민감도와 특이도의 산출식은 각각 $TP\,/\,(TP\,+\,FN)$, $TN\,/\,(TN\,+\,FP)$ 이며 X 축은 1-특이도, Y 축은 민감도이다. AUC는 이러한 ROC 곡선의 면적을 나타내며 AUC가 0.5인 경우 랜덤 추정 모델, 1인 경우 분류율이 100%인 완벽한 모델을 나타낸다. 일반적으로 0.5보다 크고 1보다 작은 값을 가지며 1에 근접할수록 분류성능이 높은 모델로 평가된다(Fawcett, 2006).

<표 2>는 언급된 지표들의 수식이다. <그림 14>는 ROC 곡선과 AUC를 나타내었으며 A는 완벽한 모델, B는 일반적인 모델, C는 랜덤 추정 모델이다 (Zhang et al., 2022).

<표 2> 성능 지표 & 수식

Measure name	Formula
Accuracy	$\frac{TP + TN}{TP + FN + TN + FP}$
Precision	$\frac{TP}{TP + FP}$
Recall	$\frac{TP}{TP + FN}$
$F_\beta - Score$	$\frac{(1 + \beta^2) \cdot (Precision \cdot Recall)}{(\beta^2 \cdot Precision + Recall)}$
$F_1 - Score$	$\frac{2 \cdot (Precision \cdot Recall)}{Precision + Recall}$
Sensitivity	$\frac{TP}{TP + FN}$
Specificity	$\frac{TN}{TN + FP}$



<그림 14> ROC 곡선의 세 가지 예

자료: (Zhang et al., 2022)

제 Ⅳ 장 실험 및 결과

제 1 절 데이터 설명

본 논문에서 제안하는 G-CTGAN 기법의 우수성 검증을 위해 라벨의 비율이 불균형한 경기도 지방세 악성 체납자 데이터를 사용하여 실험을 진행하였다. 2021년 1월을 기준 시점으로 관측된 경기도 지방세 체납자 중 체납일수가 1년 이상인 악성 체납자 452,783명을 분석 대상으로 선정하였다. 종속변수(우량체납자, 불량체납자)의 정의를 위해 분석 대상들을 2021년 2월부터 7월까지 6개월간 월별 체납자 명단에서 관측하였다. 우량체납자는 6개월 관측 기간(2~7월)에 관측 횟수가 0번, 즉 1월 체납자 명단에 등록된 이후 수납하여 6개월간 지방세를 체납한 이력이 없는 대상으로 정의한다. 불량체납자는 6개월 관측 기간(2~7월)에 관측 횟수가 6번, 즉 1월 체납자 명단에 등록된 이후 6개월간 체납상태인 대상으로 정의한다. 또한 정교한 분류모델 개발을 위해 우량, 불량체납자로 정의할 수 없는 대상들은 불분명 집단(Gray Zone)으로 분류하여 실험 데이터에서 제외한다(전희주, 2008). 이와 같이 불분명 집단을 고려하는 이유는 우량체납자, 불량체납자 집단만이 가지는 데이터 특성을 분류모델에 반영하기 위함이다. 분석에서 제외된 불분명 집단은 22.3%인 100,841명이고 실험에 사용된 건수는 우량체납자 23,956명, 불량체납자 327,986명으로 총 351,942명이다. <표 3>은 전체

데이터에서 우량, 불량, 불분명 집단의 건수 및 구성비 현황이다.

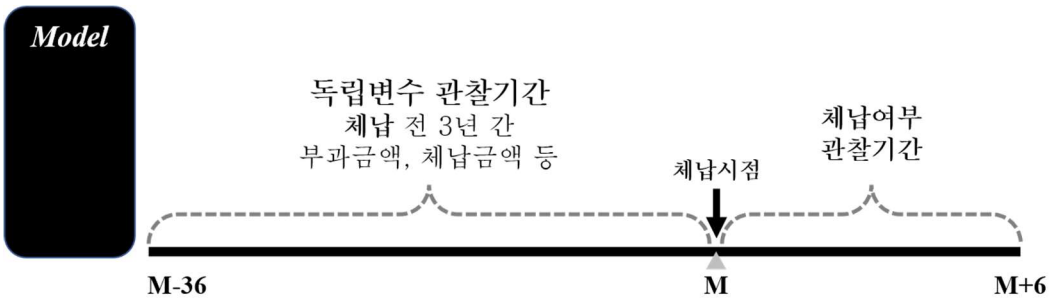
<표 3> 우량체납자, 불량체납자, 불분명 집단 현황

Label	Count	Ratio
Minority	23,956	5.3%
Majority	327,986	72.4%
Gray zone	100,841	22.3%
Total	**452,783**	**100.0%**

우량체납자와 불량체납자를 구분하는 분류모델을 생성하기 위한 독립변수는 경기도 지방자치단체에서 수집하는 Local 항목 11개와 개인신용평가회사(Credit Bureau)의 Credit 항목 9개를 사용하였으며, 변수명은 마스킹 처리하였다. Local 항목은 최초 체납 관측 시점으로부터 과거 3년간 발생한 세목별 부과금액, 체납금액 등이다. Credit 항목은 체납자 명단 관측시점에 추출된 개인신용정보이다. 독립변수 모두 연속형 변수이며 <표 4>는 우량체납자, 불량체납자로 구성된 351,942건에 대한 독립변수의 기초통계량, <그림 15>은 독립변수, 종속변수를 정의하기 위한 변수 관측 기간을 도식화한 것이다.

<표 4> 독립변수 기초통계량

Category	No	Feature	Count	Mean	Min	Max
Local	1	L1	351942	1029692.91	0	1121712800
	2	L2	351942	136630.42	0	497441150
	3	L3	351942	40617.86	0	141866850
	4	L4	351942	-0.01	-1	818.07202
	5	L5	351942	59056.30	0	7100080
	6	L6	351942	-0.05	-1	179.71
	7	L7	351942	29348.69	0	143690240
	8	L8	351942	2573923.83	54545	788555833.33
	9	L9	351942	102174.48	0	2807707360
	10	L10	351942	5732044.46	0	58322213354
	11	L11	351942	31537802.28	0	92554000000
Credit	12	C1	351942	3381.79	-9310	1866348
	13	C2	351942	2086.17	0	3023853
	14	C3	351942	40205.45	0	896776373
	15	C4	351942	102.76	0	750000
	16	C5	351942	572.01	0	493372
	17	C6	351942	5676.25	0	6591680
	18	C7	351942	20158.48	0	9590221
	19	C8	351942	2533.09	0	9158405
	20	C9	351942	3764.82	0	2741124



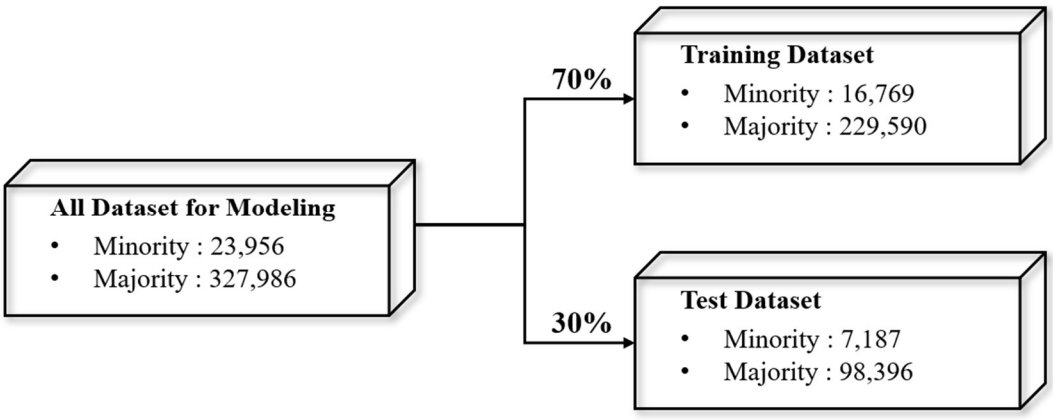
<그림 15> 변수 관측 기간

제 2 절 실험 설계

실험을 위해 불분명 집단이 제거된 데이터를 7:3의 비율로 훈련 데이터, 시험 데이터로 분할하였다. <표 5>는 실험에 사용된 데이터의 종속변수 비율, <그림 16>은 훈련 데이터와 시험 데이터를 나타낸다.

<표 5> 우량체납자, 불량체납자, 종속변수 비율 현황

Label	Count	Ratio
Minority	23,956	6.8%
Majority	327,986	93.2%
Total	**351,942**	**100.0%**



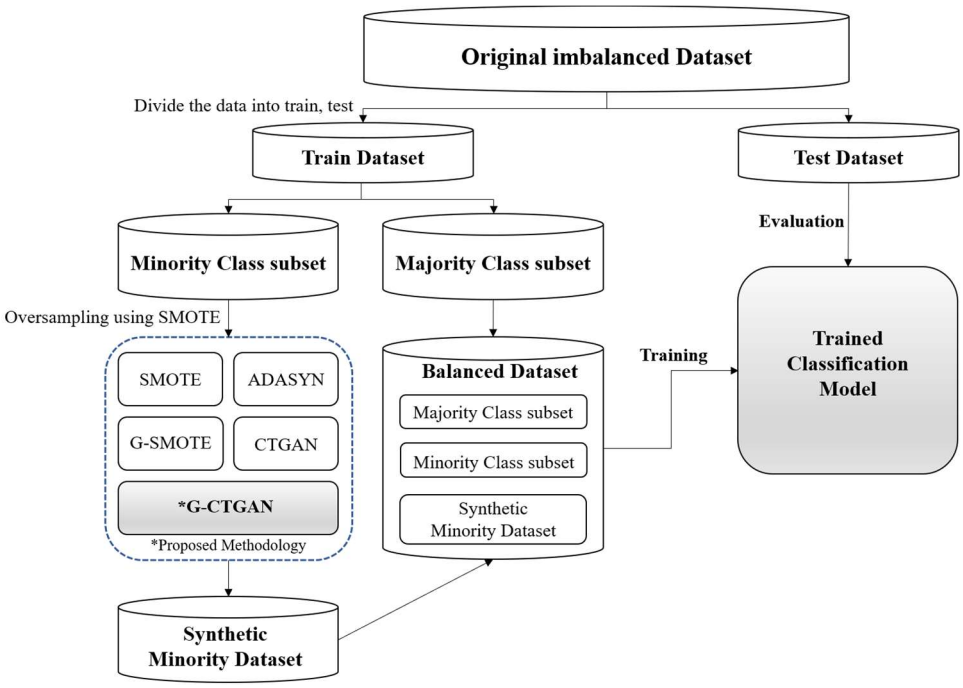
<그림 16> 훈련 데이터 & 시험 데이터

현존하는 오버샘플링 기법과 G-CTGAN을 비교하기 위해 다음과 같이 실험을 설계하였다. 실험을 위해 훈련 데이터 원본과 SMOTE, ADASYN, G-SMOTE, CTGAN, G-CTGAN 방법론을 훈련 데이터에 적용하여 라벨 비율의 균형을 맞춘 새로운 훈련 데이터 5개, 총 6개의 데이터를 정의한다. SMOTE 계열의 경우 k 값은 일반적으로 사용되는 $k = 3$ 을 사용하였다(Chawla et al., 2002). CTGAN의 매개변수는 파이썬 패키지 ctgan 0.6.0 (Xu et al., 2019)에서 제공하는 기본값을 사용하였다. G-CTGAN은 소수 범주 데이터에 가우시안 혼합 모델을 적용하여 BIC 수치가 최저(Steele & Raftery, 2010)인 5개의 군집으로 나누고, 각각의 군집에 CTGAN을 적용하여 3만 건의 데이터를 생성하였다. 생성된 인공데이터와 훈련 데이터 원본을 병합하여 라벨 비율이 5:5가 되도록 설정하였다. <표 6>은 오버샘플링이 미적용된 훈련 데이터 원본과 생성된 인공데이터들의 요약정보이다.

<표 6> 훈련 데이터 원본 & 생성된 인공데이터 현황

No	Oversampling	Total	Label description		% of Minority
			Minority	Majority	
1	None	246,359	16,769	229,590	6.8%
2	SMOTE	459,180	229,590	229,590	50.0%
3	ADASYN	464,244	234,654	229,590	50.5%
4	G-SMOTE	459,180	229,590	229,590	50.0%
5	CTGAN	446,359	216,769	229,590	48.6%
6	G-CTGAN	396,359	166,769	229,590	42.1%

분류 알고리즘은 랜덤 포레스트, Light GBM, 다층 신경망, TabNet을 사용하고, 오버샘플링 기법으로 새롭게 정의된 훈련 데이터에 최적으로 설정된 초모수(Hyper-parameter)를 탐색하여 기계학습을 진행하였다. 이후 학습된 모델을 시험 데이터에 적용하여 성능 평가를 위한 척도 AUC, $F_1 - Score$, Precision, Recall을 사용하여 분류 정확도를 비교하였다. <그림 17>은 실험과정을 도식화한 것이다.



<그림 17> 실험과정 도식화

제 3 절 실험 결과

본 절은 앞서 정의한 실험 사례들의 분류 알고리즘 성능 결과를 비교한다. <표 7>은 분류 결과에 따른 AUC, $F_1 - Score$, Precision, Recall 수치를 나타낸다. 분류 알고리즘별 가장 성능이 높은 지표는 bold체로 표기한다.

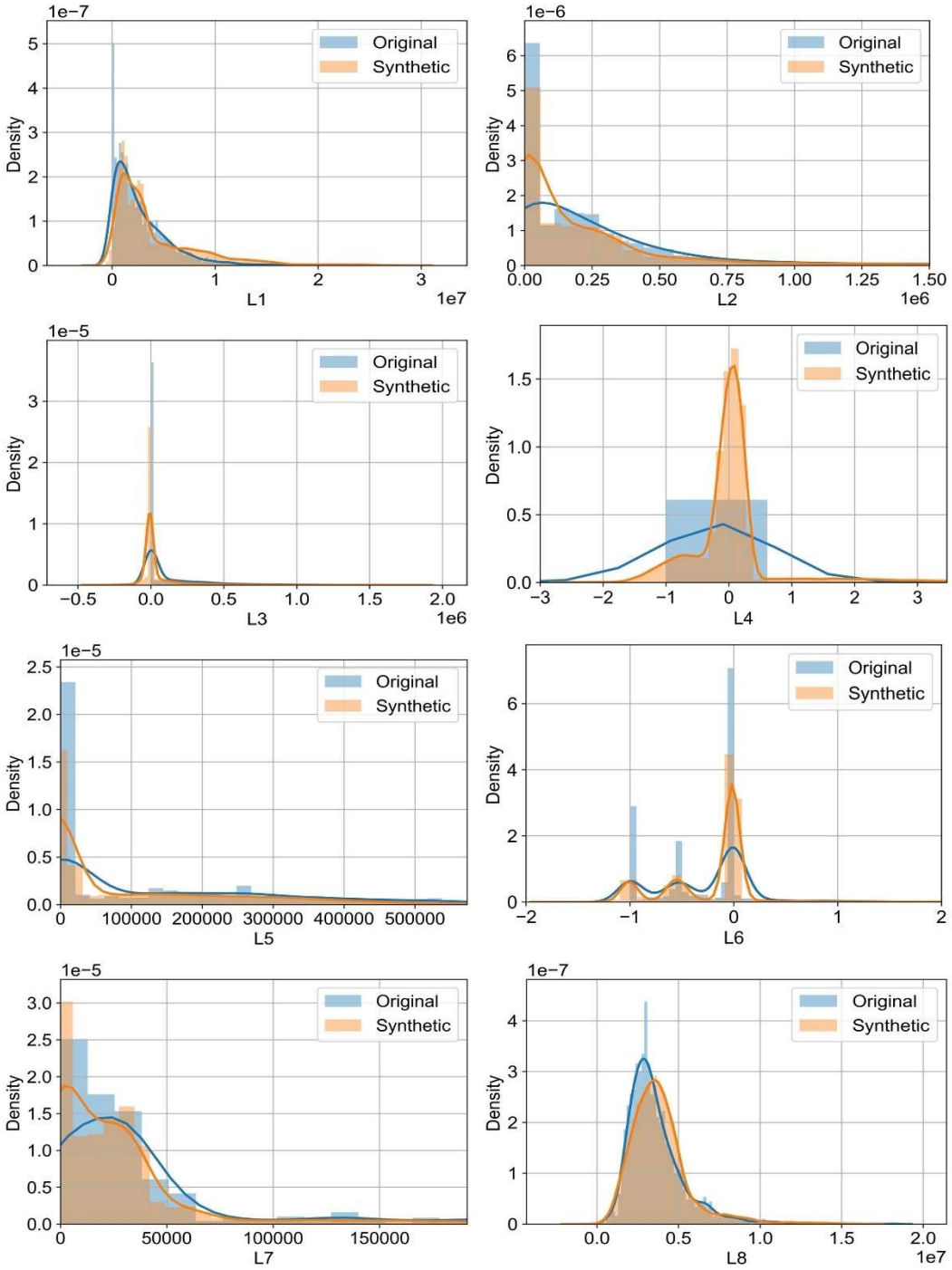
분석 결과 랜덤 포레스트 모델은 G-CTGAN을 적용할 때 AUC, $F_1 - Score$, Precision, Recall 수치가 각각 0.7622, 0.3261, 0.2792, 0.3920으로 가장 높다. Light GBM 모델은 G-CTGAN을 적용할 때 AUC, $F_1 - Score$, Precision 수치가 각각 0.7464, 0.2958, 0.2585로 가장 높고, Recall은 훈련 데이터 원본을 사용할 때 0.3956으로 가장 높다. 다층 신경망 모델은 G-SMOTE를 적용할 때 AUC, $F_1 - Score$, Precision 수치가 각각 0.7391, 0.2995, 0.3022로 가장 높고, Recall은 훈련 데이터 원본을 사용할 때 0.4192로 가장 높다. TabNet 모델은 G-SMOTE를 적용할 때 AUC, $F_1 - Score$, Precision 수치가 각각 0.7330, 0.2986, 0.2464로 가장 높고, CTGAN을 사용할 때 0.8489로 가장 높다.

총 24개의 실험사례를 비교분석 하였고, 종합적으로 랜덤 포레스트 모델과 Light GBM 모델에 제안한 방법론을 적용했을 때 미적용 사례보다 분류 성능이 개선되었음을 확인하였다. 랜덤 포레스트의 경우 오버샘플링 기법 미적용 사례보다 G-CTGAN을 적용했을 때 AUC는 0.0292, F1-Score는 0.0251 증가하였고, Light GBM의 경우 AUC는 0.0434, $F_1 - Score$는 0.0053 증가하였다. 상대적으로 다층 신경망 모델과 TabNet 모델의 분류성능은 G-SMOTE 기법을 적용하였을 때 개선되었음을 확인할 수 있었다.

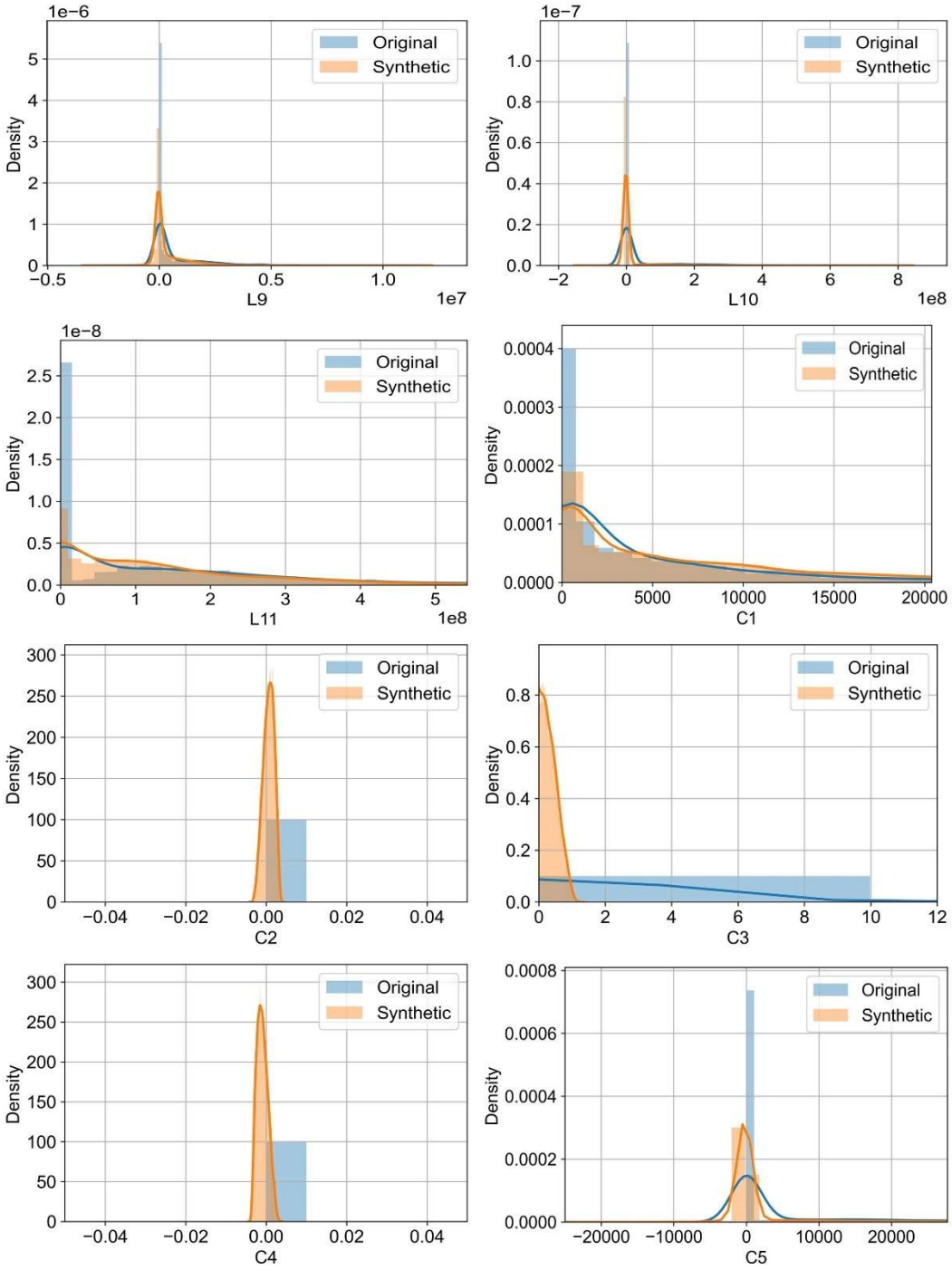
<표 7> 알고리즘별 분류 성능 현황

Classifier	No	Oversampling	AUC	F1-Score	Precision	Recall
RF	1	None	0.7330	0.3010	0.2694	0.3409
	2	SMOTE	0.7075	0.2488	0.1958	0.3413
	3	ADASYN	0.7020	0.2425	0.1816	0.3651
	4	G-SMOTE	0.7309	0.2966	0.2667	0.3339
	5	CTGAN	0.7315	0.2975	0.2485	0.3707
	6	G-CTGAN	**0.7622**	**0.3261**	**0.2792**	**0.3920**
LGBM	7	None	0.7269	0.2905	0.2296	**0.3956**
	8	SMOTE	0.7030	0.2710	0.2322	0.3253
	9	ADASYN	0.6936	0.2574	0.2179	0.3143
	10	G-SMOTE	0.7170	0.2545	0.2382	0.2731
	11	CTGAN	0.7321	0.2887	0.2482	0.3451
	12	G-CTGAN	**0.7464**	**0.2958**	**0.2585**	0.3458
MLP	13	None	0.7178	0.2696	0.1987	**0.4192**
	14	SMOTE	0.6892	0.2610	0.2167	0.3280
	15	ADASYN	0.6786	0.2518	0.2214	0.2918
	16	G-SMOTE	**0.7391**	**0.2995**	**0.3022**	0.2969
	17	CTGAN	0.6354	0.1995	0.1360	0.3744
	18	G-CTGAN	0.7030	0.2635	0.2278	0.3124
TabNet	19	None	0.7008	0.2430	0.1781	0.3824
	20	SMOTE	0.6345	0.2016	0.1833	0.2240
	21	ADASYN	0.6635	0.2469	0.1982	0.3274
	22	G-SMOTE	**0.7330**	**0.2986**	**0.2464**	0.3789
	23	CTGAN	0.4283	0.1247	0.0673	**0.8489**
	24	G-CTGAN	0.5447	0.2012	0.1708	0.2449

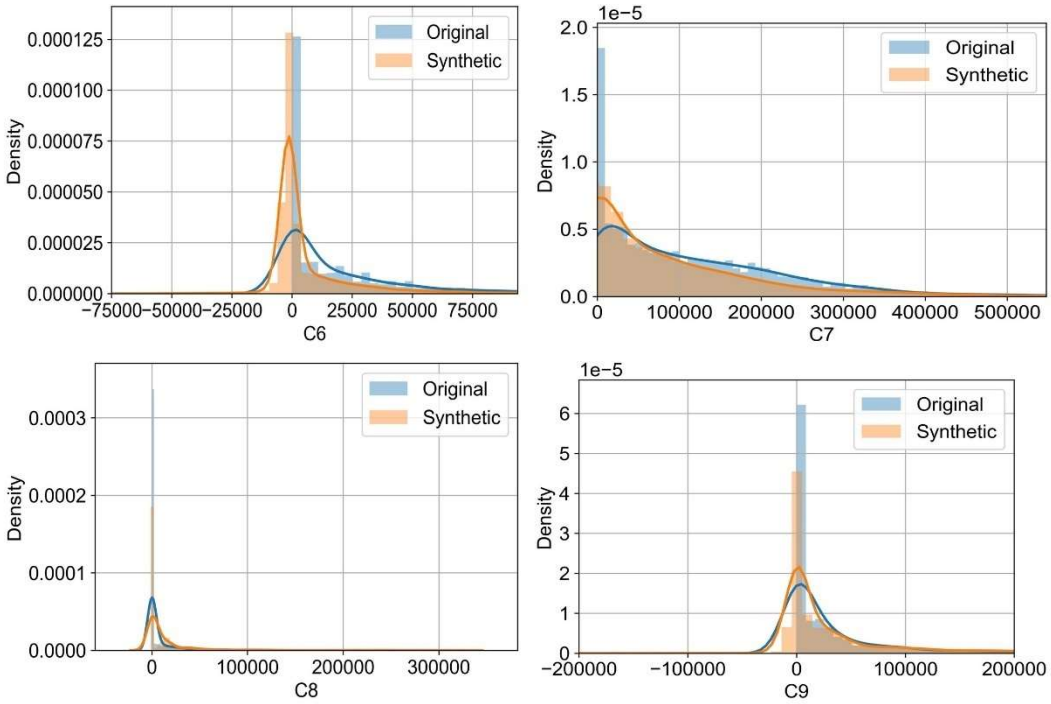
G-CTGAN을 적용하여 생성된 소수 범주 데이터 내 5개의 군집 중 가장 데이터 건수가 많은 1개의 군집에서 독립변수 20개의 데이터 분포를 확인하였다. 원본 데이터와 인공데이터로 나누어 분포를 비교하였고, <그림 18>, <그림 19>, <그림 20>에 나타내었다. 다음의 그림들을 살펴보았을 때 <그림 18>의 L4, <그림 19>은 C2, C3, C4, <그림 20>의 C6 변수를 제외하고 데이터의 분포가 유사함을 확인할 수 있었다.



<그림 18> 원본 데이터 & 인공 데이터 분포 비교(1)



<그림 19> 원본 데이터 & 인공 데이터 분포 비교(2)



<그림 20> 원본 데이터 & 인공 데이터 분포 비교(3)

제 V 장 결론 및 고찰

데이터 불균형 현상으로 인해 발생하는 분류모델의 성능저하 문제를 해결하기 위해 다양한 방법들이 연구되어왔다. 그중 샘플링 수준의 접근 방법으로 SMOTE 기법과 GAN 기법이 제안되었지만 과적합과 이상치에 의한 성능저하 등 단점들이 존재하였다. 본 논문에서는 이러한 단점을 보완하기 위해 G-CTGAN 기법을 제안하였다. 경기도 악성 체납자 데이터를 통해 분석 결과를 비교하였을 때, 랜덤 포레스트 모델과 Light GBM 모델에서 우수한 분류 성능을 나타내었다. 해당 모델을 활용하여 다수의 악성체납자중 소수의 우량체납자를 선별한다면 지방자치단체 체납담당자의 업무효율증대와 징수율 향상에 기여할 수 있다(행정안전부, 2021).

본 논문의 연구 결과가 시사하는 바는 다음과 같다. 첫 번째로 군집분석을 통해 나눠진 각각의 군집에 오버샘플링을 적용한 훈련 데이터를 사용한 경우 더 높은 분류성능을 보였다는 것이다. 실험 결과를 보면 랜덤 포레스트, Light GBM 모델에서 본 논문에서 제안한 G-CTGAN 오버샘플링 기법을 사용한 경우 분류 성능, 즉 AUC가 가장 높았다. 또한 다층 신경망, TabNet 모델에서는 G-SMOTE를 사용하였을 때 AUC가 가장 높았다. G-CTGAN, G-SMOTE 모두 가우시안 혼합 모델을 적용하여 군집을 나누어 오버샘플링을 적용한다는 공통점이 있다. 이는 군집분석을 통해 전체 집단 분포에 존재하는 하위 집단의 분포를 추정하고, 유사한 확률밀도함수(Probability Density Function, PDF)를 가지는 하위 집단에 오버샘플링을 적용하는 것이 더욱 정교한 인공

데이터를 생성한다는 것을 유추할 수 있다. 두 번째, G-CTGAN을 실무에 용이하게 적용할 수 있다 점이다. 가우시안 혼합 모델과 CTGAN 오버샘플링은 모델러의 특별한 개입 없이도 개발된 파이썬 패키지를 활용하여 빠르게 양질의 인공 데이터를 생성할 수 있다. 그 때문에 실무자는 다양한 실험을 빠르게 수행할 수 있고, 분석 목표에 맞는 최적의 예측 알고리즘 개발이 가능하다.

본 논문의 후속으로 다음과 같은 연구를 추가로 진행할 수 있다. 본 논문에서 제안한 G-CTGAN을 적용할 때, 가우시안 혼합 모델을 이용하여 군집을 형성한 뒤 CTGAN을 이용하여 각 군집별로 생성할 인공데이터의 수를 임의로 설정하였다. Shin et al.(2021)에서 제안한 바와 같이 유전알고리즘을 이용하여 최적의 오버샘플링 비율을 탐색하고 효과적으로 샘플링하여 연구를 진행할 수 있을 것이다. 또한 본 논문에서 사용된 실험 데이터의 변수 특성이 모두 연속형 변수이기 때문에 범주형 변수에 대해서는 다루지 않았다. CTGAN은 연속형 변수 외에도 범주형 변수의 처리도 가능하기 때문에 김석준 & 이종석(2021)과 같이 두 변수가 혼합된 데이터를 사용하여 연구를 진행할 수 있을 것이다. 마지막, G-CTGAN이 적용 가능한 영역으로 범주가 3개 이상인 다중 분류 문제를 예로 들 수 있다. 경기도 악성 체납자 데이터의 종속변수를 정의할 때 현업의 요구사항, 도메인 지식 등을 바탕으로 분석 대상에서 제외하였던 불분명 집단에 대한 새로운 정의가 가능하다. 이러한 점에서 우량체납자, 불량체납자 외 새로운 범주의 체납자 유형이 추가되어 2개 이상의 범주로 조작적 정의가 필요할 수 있다. 때문에 후속 연구로서 해당 영역에 대한 연구를 수행하고자 한다.

참고문헌

강필성, and 조성준. (2006). 데이터 불균형 해결을 위한 Under-Sampling 기반 앙상블 SVMs. 대한산업공학회/한국경영과학회, 291-298.

김석준, and 이종석. (2021). 범부형과 연속형 변수가 혼합된 불균형 데이터 분류를 위한 CAT2VEC 과 Conditional Tabular GAN 의 활용. 대한산업공학회 추계학술대회 논문집, 799-812.

전희주. (2008). 고객 로열티 스코어 모델 개발. 응용통계연구, 21(2), 211-219.

행정안전부(2021), 지방세 체납데이터 지능형(AI)분석으로 맞춤형 징수활동 추친,https://www.mois.go.kr/frt/bbs/type010/commonSelectBoardArticle.do?bbsId=BBSMSTR_000000000008&nttId=88805, 검색일 2023. 1. 1.

황철현. (2022). 공공기술 사업화를 위한 CTGAN 기반 데이터 불균형 해소. 한국정보통신학회논문지, 26(1), 64-69.

Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician, 46(3), 175-185.

Arik, S. Ö., and Pfister, T. (2021, May). Tabnet: Attentive interpretable tabular learning. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 35, No. 8, pp. 6679-6687).

Bau, D., Zhu, J. Y., Wulff, J., Peebles, W., Strobelt, H., Zhou, B., and Torralba, A. (2019). Seeing what a gan cannot generate. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 4502-4511).

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, 321-357.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society: Series B (Methodological), 39(1), 1-22.

Douzas, G., and Bacao, F. (2018). Effective data generation for imbalanced learning using conditional generative adversarial networks. Expert Systems with applications, 91, 464-471.

Fawcett, T., and Provost, F. (1997). Adaptive fraud detection. Data mining and knowledge discovery, 1(3), 291-316.

Fawcett, T. (2006). An introduction to ROC analysis. Pattern recognition letters, 27(8), 861-874.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... and Bengio, Y. (2020). Generative adversarial networks. Communications of the ACM, 63(11), 139-144.

He, H., Bai, Y., Garcia, E. A., and Li, S. (2008, June). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence) (pp. 1322-1328). IEEE.

Hinton, G. E., Osindero, S., and Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. Neural computation, 18(7), 1527-1554.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... and Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural

information processing systems, 30.

Kubat, M., Holte, R. C., and Matwin, S. (1998). Machine learning for the detection of oil spills in satellite radar images. Machine learning, 30(2), 195-215.

Lee, T. H., Ullah, A., and Wang, R. (2020). Bootstrap aggregating and random forest. In Macroeconomic forecasting in the era of big data (pp. 389-429). Springer, Cham.

Quinlan, J. R. (1986). Induction of decision trees. Machine learning, 1(1), 81-106.

Reynolds, D. A. (2009). Gaussian mixture models. Encyclopedia of biometrics, 741(659-663).

Rijsbergen, C. V. (1979). Information retrieval 2nd ed Buttersworth. London [Google Scholar].

Shin, S. S., Cho, H. Y., and Kim, Y. H. (2021). Optimal Ratio of Data Oversampling Based on a Genetic Algorithm for Overcoming Data Imbalance. Journal of the Korea Convergence Society, 12(1), 49-55.

Steele, R. J., and Raftery, A. E. (2010). Performance of Bayesian model selection criteria for Gaussian mixture models. Frontiers of statistical decision making and bayesian analysis, 2, 113-130.

Xu, L., Skoularidou, M., Cuesta-Infante, A., and Veeramachaneni, K. (2019). Modeling tabular data using conditional gan. Advances in Neural Information Processing Systems, 32.

Yan, R., Liu, Y., Jin, R., and Hauptmann, A. (2003, April). On predicting rare classes with SVM ensembles in scene classification. In 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). (Vol. 3, pp. III-21). IEEE.

Zhang, C., Zhao, J., Zhu, Z., Li, Y., Li, K., Wang, Y., and Zheng, Y. (2022). Applications

of Artificial Intelligence in Myopia: Current and Future Directions. Frontiers in Medicine, 9.

Zhang, T., and Yang, X. (2018). G-SMOTE: A GMM-based synthetic minority oversampling technique for imbalanced learning. arXiv preprint arXiv:1810.10363.

Abstract

A Study on the Oversampling Technique using Gaussian Mixture Model and CTGAN for Classification Analysis of Imbalanced Data

Yang, Munil

Seoul School of Integrated Sciences and Technologies

Advisor: Shin, Ho Sang

Imbalanced data refers to data in which the proportion of labels is significantly different and has been found across all industries. Imbalanced data has an over-distribution of majority category data compared to minority category data, and this phenomenon hinders Decision Boundary setting.

For the reason, it acts as a factor that degrades the performance of the machine learning classification algorithm. To solve this problem, various techniques have been proposed to resolve the distribution difference between minority and majority data. Among them, oversampling techniques resolve data imbalance by amplifying data in a minority category. Methods for amplifying data include Synthetic Minority Oversampling Technique (SMOTE) and Generative Adversarial Networks (GAN). The SMOTE is a method of extracting minority category data and proximity data using KNN (K-Nearest Neighbor) algorithms and then interpolating them to generate virtual data. The GAN is a generator

that generates virtual data, a discriminator that distinguishes real data from generated data, and a data augmentation technique in which two artificial neural networks compete and train.

In this paper, to overcome the shortcomings of existing oversampling techniques, we propose the oversampling technique G-CTGAN that combines Gaussian Mix Model(GMM) which a clustering algorithm, and Conditional Tabular GAN (CTGAN). GMM assumes that the distributions of subgroups exist in multiple Gaussian distributions in the overall group distribution, and is a stochastic representation model. For each individual data, the probability of belonging to the Gaussian distribution of the subgroup is calculated and allocated by applying the EM algorithm. Then, the maximum likelihood estimation (MLE) method estimates the parameters of the Gaussian distribution to which individual data belong and forms clusters for each distribution. It is a data augmentation technique that solves the problem of overfitting due to outliers that occur when applying SMOTE by applying CTGAN to different clusters formed with similar distributions. To prove excellence, the actual imbalance data, Gyeonggi-do local tax delinquent data, is used for experiments. The data was divided into training data and test data, and use existing oversampling techniques and the technique proposed in this paper to amplify minority category data in the training data, then resolve the imbalance. The experiment was conducted by generating classification models using random forest, light GBM, multi-layer neural network, and TabNet classification algorithms in balanced training data and comparing the performance of classification models using test data.

Use AUC, F1-Score, Precision, and Recall for performance comparison and confirm that the performance of the classification model is improved over the existing oversampling technique when using a technique that combines Gaussian mixture model with CTGAN.

Key words: Imbalanced data, Classification, Oversampling, SMOTE, GAN, Gaussian Mixture Model, CTGAN, G-CTGAN

Student Number: 2125418005