The Korean Journal of Applied Statistics (2019) DOI: https://doi.org/10.5351/KJAS.2019.32.3.349
32(3), 349–374

Comparison of resampling methods for dealing with imbalanced data in binary classification problem

Geun U Park[a] · Inkyung Jung[a,1]

[a]Division of Biostatistics, Department of Biomedical Systems Informatics, Yonsei University College of Medicine

(Received February 15, 2019; Revised April 2, 2019; Accepted April 2, 2019)

Abstract

A class imbalance problem arises when one class outnumbers the other class by a large proportion in binary data. Studies such as transforming the learning data have been conducted to solve this imbalance problem. In this study, we compared resampling methods among methods to deal with an imbalance in the classification problem. We sought to find a way to more effectively detect the minority class in the data. Through simulation, a total of 20 methods of over-sampling, under-sampling, and combined method of over- and under-sampling were compared. The logistic regression, support vector machine, and random forest models, which are commonly used in classification problems, were used as classifiers. The simulation results showed that the random under sampling (RUS) method had the highest sensitivity with an accuracy over 0.5. The next most sensitive method was an over-sampling adaptive synthetic sampling approach. This revealed that the RUS method was suitable for finding minority class values. The results of applying to some real data sets were similar to those of the simulation.

Keywords: imbalanced-learn, imbalanced binary data, under-sampling, over-sampling

1. 서론

이분형 자료의 분류 및 예측을 위한 많은 모형이 있다. 분류문제에서 자료의 불균형 정도가 심한 경우 분류를 적절히 하지 못하는 "계급 불균형" (Prati 등, 2009) 문제가 있다. 실제 상황에서는 계급 간의 분포가 균등하다고 가정하기 어려운 경우가 많다. 대표적인 예로 희귀질병의 유무, 신용카드 사용자 사기 여부 등이 있다.

계급 불균형 문제를 해결하기 위해 사용되는 대표적인 방법은 분류모형을 불균형 자료를 예측하기 유리한 형태로 변형하거나 학습시키는 자료를 변형시켜 균형을 맞춘 다음 모형을 적용시키는 것이다. 이 이외에도 다른 방법이 있지만 이 두 가지 전략을 기본으로 한다 (He와 Garcia, 2009). 학습 자료를 변형시키는 것은 성능은 다소 떨어지지만 이미 알려진 모형들을 그대로 적용할 수 있다는 장점이 있다. 실제로 자료를 변형시키는 방법이 더 대중적으로 쓰이고 있다 (Haixiang 등, 2017). 본 논문에서는 실용적으로 접근 가능한 자료를 변형시키는 여러 방법에 대해 비교 연구한다.

[1]Corresponding author: Division of Biostatistics, Department of Biomedical Systems Informatics, Yonsei University College of Medicine, 50-1 Yonsei-ro, Seodaemun-gu, Seoul 03722, Korea. E-mail: ijung@yuhs.ac

원 자료를 이용해 새로운 학습 자료를 이용하는 것은 원 자료의 값을 없애거나 가상의 값을 만드는 방식이다. 이를 표본재추출(resampling) 방법이라 한다. 여러 가지 표본재추출 방법을 크게 세 가지로 구분하면 오버샘플링(over-sampling), 언더샘플링(under-sampling), 그리고 둘을 혼합한 방법이다. 본 논문에서는 파이썬 패키지인 imblearn으로 구현되어 있는 표본재추출 방법을 사용한다 (Lemaître 등, 2017). 모의실험을 통하여 오버샘플링 3가지, 언더샘플링 10가지, 혼합 방법 2가지에 대해 이분형 자료의 분류문제에서 계급 불균형을 해결하는 데 유용한 방법이 무엇인지 평가해보고자 한다.

2절에서는 본 논문에서 비교할 표본재추출 방법에 대해 간략히 소개한다. 3절에서는 여러 가지 상황의 모의실험 과정과 결과를 제시하고, 4절에서는 실제 자료에 적용한 결과를 비교한다. 마지막 5절에서 본 연구의 결과를 요약하고 결론 및 고찰을 제시한다.

2. 표본재추출 방법

오버샘플링은 원 자료를 중복하여 집어넣는 random over-sampling (ROS)과 Chawla 등 (2002)이 제안한 synthetic minority over-sampling technique (SMOTE)에서 파생된 방법으로 나눌 수 있다. 언더샘플링은 Hart (1968)가 제안한 condensed nearest neighbor (CNN) rule에서 파생된 방법들로 7가지가 있다. 그 외에 random under-sampling (RUS) 방법과 near miss (NM) 방법 (Mani와 Zhang, 2003), instance hardness threshold (IHT) 방법 (Smith 등, 2014) 등이 있다.

각 재추출방법을 시각적으로 표현하기 위해 가상의 자료를 만들고 각 방법을 적용한 후의 결과를 Figures 2.1–2.4에 나타내었다. 아래와 같이 표준정규분포를 따르는 두 개의 독립변수와 표준정규분포를 따르는 오차의 합으로 10,000개의 가상 데이터를 만들었다. 종속변수는 η값을 희소계급값이 1% 비율이 되도록 두 개의 범주로 나누었다.

$$\eta = X_1 + X_2 + \epsilon, \quad X_1, X_2, \epsilon \sim N(0,1),$$
$$Y = \begin{cases} 0, & \text{if } \eta_{(0.01)} < \eta, \\ 1, & \text{if o.w.}, \end{cases}$$

Figure 2.1은 오버샘플링 방법, Figure 2.2는 CNN 기반 언더샘플링 방법, Figure 2.3은 그 외 언더샘플링 방법, Figure 2.4는 혼합 방법이 적용되었을 때 원 자료에서 어떻게 변형되었는지를 나타낸다. 주황색이 희소 계급으로 원 자료에서는 1%에 해당한다.

2.1. 오버샘플링

희소계급의 데이터값을 그대로 중복하여 모형적합에 사용하는 방법인 ROS 방법과 희소계급에서 두 개의 점을 이어 일직선상에 임의의 값을 추가하는 방법이 있다. 모든 방법들은 희소계급값을 다수계급값과 같아질 때까지 오버샘플링 하여 1 : 1 비율을 맞춘다.

2.1.1. Synthetic minority over-sampling technique (SMOTE) SMOTE는 불균형 문제를 해결하기 위한 방법으로 Chawla 등 (2002)에 의해 개발된 방법으로 알고리즘은 Algorithm 1과 같다.

실제 imblearn의 내부의 함수를 적용하면 원래 소수계급에 속하는 값만을 가지고 오버샘플링 한다. imblearn의 SMOTE는 1가지 기본 옵션과 3가지 변형 옵션을 적용할 수 있다. 본 논문에서는 기본옵션인 'regular'와 'borderline1', 'borderline2'를 사용한다. 'regular' 옵션은 SMOTE이고, 'borderline1', 'borderline2' 옵션은 Borderline SMOTE이다. K-nearest neighbor (KNN)를 적용할 때의 K의 값은 원 논문에서 제안한 값인 5개의 이웃을 사용하였다.

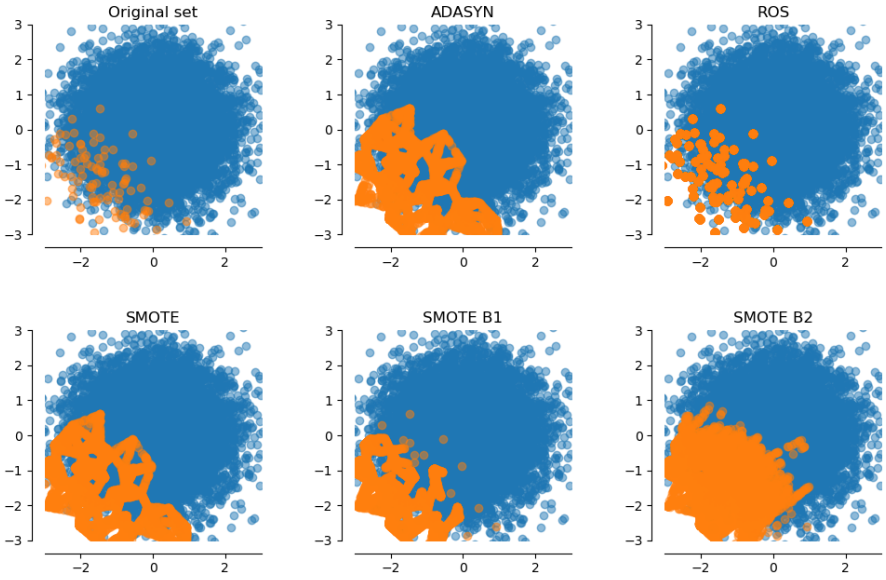

Figure 2.1. Changes in data set after applying various over-sampling methods.

Algorithm 1

1: 소수계급에 속하는 자료들을 집단 T로 한다.
2: 집단 T에서 하나의 표본을 뽑아 K-nearest neighbor 중 하나를 선택한다.
3: 2에서 뽑혀진 표본 쌍의 잇는 가운데 임의의 점을 찍는다.
4: 3에서 뽑은 표본을 집단 T에 넣고 집단 T가 다수계급의 숫자와 같아질 때까지 반복한다.

2.1.2. Borderline SMOTE Han 등 (2005)이 제안한 방법으로 SMOTE를 확장한 방법이다. 경계값에 있는 값만을 이용하여 SMOTE 오버샘플링 한다. 소수계급에 속하는 자료 주변의 KNN의 계급을 보고 경계인지를 판단한다. 일정 비율(원 논문에서는 0.5) 이상이 다수계급값인 경우 경계면에 있다고 판단하고 이 경우의 값들만을 가지고 SMOTE를 적용한다. 이때 KNN이 모두 다수계급값인 경우에는 잡음으로 판단하여 SMOTE를 적용시키는 집단에서 제외시킨다. KNN을 적용할 때 이웃의 개수는 Han 등 (2005)이 제안한 값인 5로 한다. borderline1은 기본 SMOTE와 마찬가지로 희소계급에 속하는 값들만 가지고 오버샘플링 한다. borderline2는 오버샘플링 되는 점을 결정하게 되는 두 점을 선택할 때, 한 점은 희소계급값의 경계값 이지만 다른 한 점을 선택할 때는 다수계급값 에서도 선택할 수 있다.

본 논문의 세팅에서는 희소계급값이 적으므로 borderline1은 SMOTE와 마찬가지로 희소계급을 이은 선들이 보인다 (Figure 2.1).

2.1.3. Adaptive synthetic sampling approach (ADASYN) He 등 (2008)이 제안한 방법으로 SMOTE를 발전시킨 방법이다. 알고리즘은 다음 Algorithm 2와 같다. borderline SMOTE와 유사하게 경계값에 가중치를 둔다. KNN를 적용시킬 때 이웃의 개수는 5를 사용하였다.

2.1.4. Random over-sampling (ROS) ROS 방법은 소수계급의 표본 수를 늘리기 위해 소수계급의 표본을 무작위로 선택해 반복 추출하는 방법이다 (Prati 등, 2009). 소수계급의 표본 수가 증가하

Algorithm 2

1: 불균형 정도 측정.
2: 소수계급에 속하는 값의 K-nearest neighbor 중 다수계급에 속하는 값의 비율을 구하고 이를 r_i라 한다.
3: 모든 소수계급의 값에대해 r_i를 구해 r_i값을 표준화 한다 ($\hat{r}_i = r_i / \sum_i r_i$).
4: 균형을 맞추기 오버샘플 해야하는 데이터의 개수를 $\hat{r}_i$에 곱하여 g_i를 구한다 ($g_i = \hat{r}_i \times$ resample N).
5: 각 g_i에 x_i를 대응시키고 x_i에 대응되는 K-nearest neighbor 중 소수계급에 속하는 값에서 임의로 하나의 값을 뽑는다.
6: 5에서 뽑혀진 값과 x_i 가운데 임의의 점을 뽑는다.
7: 5의 소수계급에 속한 모든 x_i에 대해 5, 6번을 g_i만큼 반복한다.

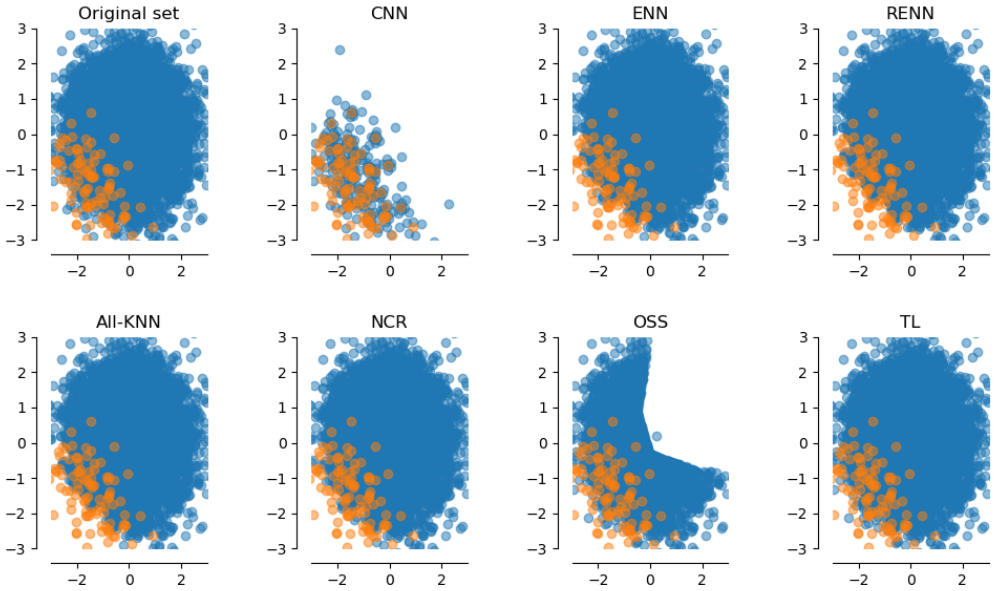

Figure 2.2. Changes in data set after applying various CNN-based under-sampling methods.

는 만큼 자료의 크기는 커지지만, 단순히 소수계급의 표본을 반복하는 것이기 때문에 정보의 양이 늘어나는 것은 아니다. 따라서 정보 손실은 없으나 같은 표본의 중복으로 인해 과적합(over-fitting) 문제가 발생할 수 있고, 자료의 크기가 커지므로 분류 알고리즘을 적용할 때 학습시간이 증가한다.

2.2. CNN 기반 언더샘플링

Figure 2.2를 보면 CNN을 제외한 다른 모든 방법들이 다수계급값을 소수계급값과 숫자를 맞출 때까지 없애지 못한다는 것을 알 수 있다.

2.2.1. Condensed nearest neighbor (CNN) Hart (1968)가 제안한 방법이다. 원 논문에서는 nearest neighbor 알고리즘에 비해 메모리 사용량을 줄인다는 장점이 있다고 한다. CNN의 목적은

최소한의 부분집합을 찾는 것이다. 하지만 이 알고리즘은 최소한의 부분집합을 찾지는 못한다고 후속연구에서 밝혀졌다 (Gates, 1972). 후속논문에 의하면 표본의 개수가 커지면 시간이 많이 걸린다고 한다. 알고리즘은 다음 Algorithm 3과 같다.

Algorithm 3

1: 표본을 하나 뽑고 집합 S에 포함시킨다.
2: 다음 표본을 뽑고 집합 S에 nearest neighbor rule로 분류한다.
3: 만약 분류가 틀렸으면 그 표본을 집합 S에 포함시킨다.
4: 집합 S에 포함되어있지 않은 모든 값의 분류가 집합 S로 가능할 때까지 2, 3번을 반복한다.
결과로 나온 집합 S를 return 한다.

imblearn 패키지의 함수에서 설정해줄 수 있는 것은 처음 집합 S를 만들 때 추출하는 표본의 개수와 nearest neighbor rule을 적용할 때 사용하는 이웃의 개수이다. 함수에서 기본값은 둘 다 1이고 그대로 1을 사용하였다.

본래 이 방법의 목적은 서로 다른 두 계급의 경계가 명확할 때 경계부분의 데이터 값만 남겨 경계를 구하는데 드는 계산을 줄이고자 하는 것이 목적이다. 하지만 Figure 2.2의 상황에서는 경계값이 많이 겹치는 데이터로 오히려 분류가 쉽지 않은 상황이 만들어졌다.

2.2.2. Edited nearest neighbors (ENN) Wilson (1972)이 제안한 CNN의 변형이다. CNN과 달리 집합 S에 포함된 값이 틀린 분류를 하면 S에서 제외시킬 수 있다. CNN과 달리 K-NN을 적용할 때 기본값으로 3을 사용한다. 이로 인해 분류가 잘못된 경우에는 그 값을 S에서 제외시킨다.

2.2.3. Repeated ENN (RENN)과 All K-NN 두 가지 방법 모두 CNN의 변형 알고리즘이다. RENN은 ENN을 집합 S가 수렴할 때까지 반복한다. 이로 인해 최소한의 값을 가지는 S을 찾을 수 있다 (Tomek, 1976a).

ALL-KNN은 분류기의 K-NN을 적용할 때 $K = 1, 3, 5$인 모든 경우에 정확한 분류를 하였을 때 집합 S에 포함시키는 분류기이다 (Tomek, 1976b). 다소 보수적인 방법이라고 할 수 있다.

2.2.4. Tomek link (TL) Tomek (1976a, 1976b)이 제안한 방법으로 CNN의 변형이라고 할 수 있다. 전체 자료에서 서로 다른 계급을 갖는 두 개의 표본(X_i, X_j)이 다음을 만족할 때 Tomek Link라 정의한다. $d(X_i, X_j)$를 X_i와 X_j의 거리라고 하자. 이 때 $d(X_i, X_k) < d(X_i, X_j)$ 혹은 $d(X_j, X_k) < d(X_i, X_j)$를 만족하는 X_k가 존재하지 않으면 (X_i, X_j)는 Tomek Link이다. Tomek Link인 값들은 양쪽 계급의 경계에 있는 값이거나 잡음이다. 이 X_i, X_j 중 다수계급값을 제거하여 언더샘플링 한다.

2.2.5. Neighborhood cleaning rule (NCR) Laurikkala (2001)가 제안한 ENN을 변형한 방법이다. 표본을 3-NN으로 분류하여 잘못 분류하면 삭제하는 방법이다. 만약 3-NN으로 분류된 값이 다수계급이면 분류된 값을 지우고, 3-NN으로 분류된 값이 소수계급이면 분류된 값은 그대로 두고 3-NN중 다수계급 값을 지운다.

2.2.6. One sided selection (OSS) 다수계급값의 데이터를 CNN을 사용하는 과정 중 Tomek Link를 이용하여 잡음을 제거한다.

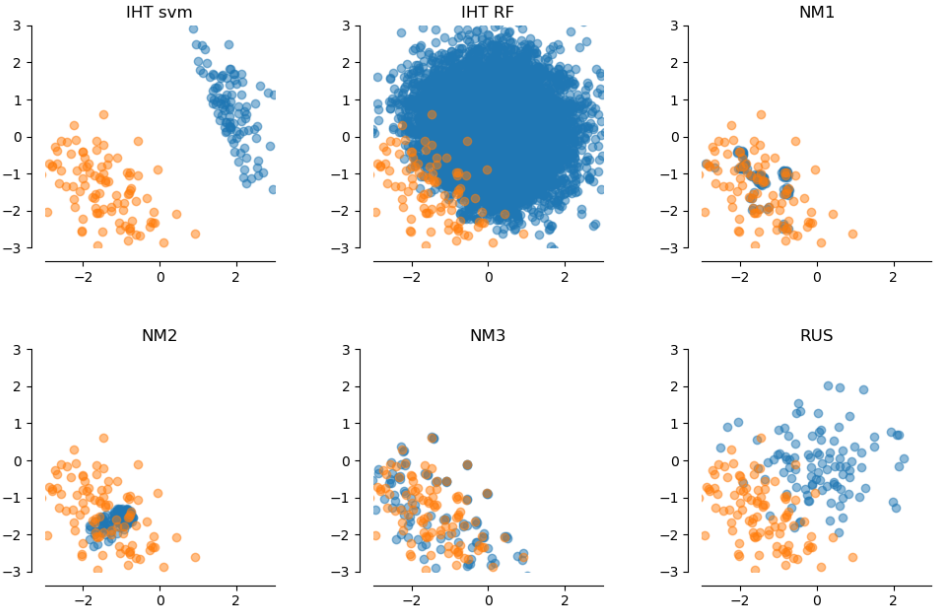

Figure 2.3. Changes in data set after applying various under-sampling methods.

2.3. 언더샘플링

2.3.1. Instance hardness threshold (IHT) 다른 방법들과 달리 거리를 정의하여 분류하는 방법이 아니라 분류모형에 기반하여 언더샘플링 하는 방법이다. 이상치와 같이 모형의 분류가 어려워지는 확률을 계산하여 없애는 것이다. 분류 알고리즘을 결정하고 각 데이터 값에 대한 오분류 확률을 계산하여 오분류 확률이 높은 값을 없애는 방법이다.

이 방법을 사용하기 위해서는 필연적으로 분류 알고리즘을 선택하여야 한다. 이 방법을 제안한 논문 (Smith 등, 2014)에서는 다양한 알고리즘을 모두 사용하여 평균을 사용하는 방법으로 소개되어 있다. imblearn 패키지에서는 하나의 분류 알고리즘을 선택하여 분류할 수 있도록 되어있고, 기본값은 랜덤 포레스트(random forest classifier)로 되어있다. 파이썬 패키지인 scikit-learn에 있는 classifier 중 선택하여 사용할 수 있다. scikit-learn의 랜덤 포레스트의 나무 개수의 기본값이 10이다. 분석에서는 100으로 수정해 사용하였다. 다른 하나로는 linear-SVM을 사용하여 IHT를 적용시켰다. cross validation은 기본값인 5를 그대로 사용하였다. Figure 2.3을 보면 IHT RF는 CNN계열의 CNN을 제외한 다른 방법과 같이 다수계급값을 희소계급값의 숫자와 같아질 때까지 없애지 못하는 것을 확인할 수 있다.

2.3.2. Near miss (NM) NM은 세 가지 방법이 있다. 첫 번째는 몇 개의 소수집단에 가까운 다수집단만을 선택하는 방법이다. NM 방법에서 '몇 개'는 보통 3개로 정의된다. 다수집단의 각 값들에서 가장 가까운 3개의 소수집단값의 거리평균을 구한다. 이 거리평균이 작은 것을 선택한다. 두 번째 방법은 모든 소수집단에 가까운 다수집단만을 선택하는 것이다. 각 다수집단의 값에서 가장 먼 3개의 소수집단값의 거리평균을 구한다. 이 거리평균이 작은 것을 선택한다. 세 번째 방법은 각 소수집단값의 가장 가까운 다수집단을 몇 개 선택하는 것이다. 기본값은 첫 번째 방법으로 되어 있는데 이 방법을 제안한 논문 (Mani와 Zhang, 2003)에 의하면 두 번째 방법이 가장 좋다고 한다. 이 논문에서는 3가지 방법을 모두 사용한다.

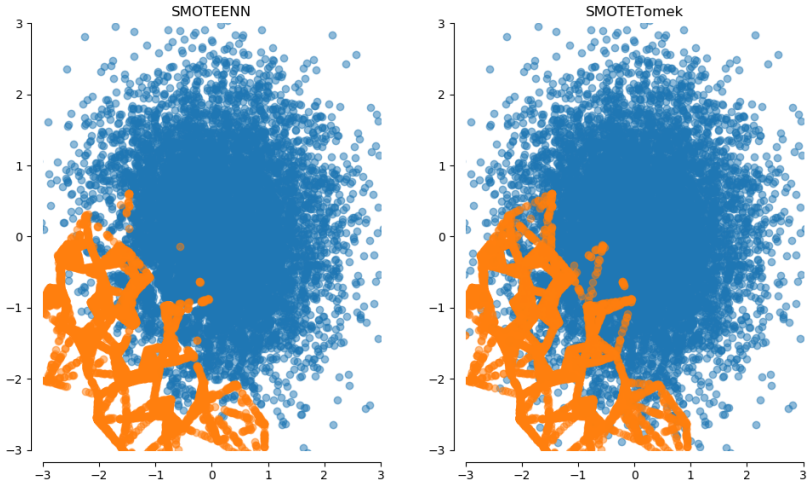

Figure 2.4. Changes in data set after applying two combined methods.

Table 2.1. Misclassification table

		Predicted class	
		Positive	Negative
Actual class	Positive	True Positives (TP)	False Negatives (FN)
	Negative	False Positives (FP)	True Negatives (TN)

2.3.3. Random under sampler (RUS) 다수계급의 값에서 소수계급의 개수만큼 임의로 표본추출하는 방법이다 (Prati 등, 2009). 따라서 다수계급의 표본 수가 감소하는 만큼 자료의 크기는 작아지고, 그만큼의 정보 손실이 발생한다. 하지만 랜덤오버샘플링 방법에 비해 자료의 크기가 작은 만큼 크기가 큰 데이터를 다룰 때 학습 과정에 소요되는 시간을 단축할 수 있다. 다른 모든 방법들이 거리에 기반한 선택 알고리즘인 반면에 임의추출이므로 거리에 대한 가정을 필요로 하지 않는다.

2.4. 혼합 방법

2.4.1. SMOTE ENN SMOTE를 적용하여 오버샘플링 한 이후 ENN을 이용하여 언더샘플링 하는 방법이다 (Batista 등, 2004). 이와 같이 결합하는 방식을 다른 방법들에도 적용할 수 있다.

2.4.2. SMOTE TL SMOTE를 적용하여 오버샘플링을 한 이후 TL 방법을 이용하여 자료를 정제하는 방법이다. TL 방법은 자료를 정제하기 위해 사용하는 것이므로 Tomek Link인 값들을 전부 삭제한다.

2.5. 모형 평가지표

이진 분류모형에서 분류 모형의 예측 성능을 평가하는 평가 척도는 오분류표(misclassification table)로부터 계산한다 (Table 2.1). 혼동행렬(confusion matrix)이라는 용어로도 표현한다.

- 정확도(accuracy)

  예측한 값이 정확히 맞은 비율을 의미하며 다음과 같이 계산한다.

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{TN} + \text{FN}}.$$

- 민감도(sensitivity)

  실제 양성인 자료 중 양성으로 예측 한 값의 비율을 의미한다. 재현율(recall)로도 표현한다.

$$\text{sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}.$$

- 양성예측도(positive predictive value; PPV)

  양성으로 예측한 값 중 실제 양성인 자료의 비율을 의미한다. 정밀도(precision)로 표현하기도 한다.

$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}}.$$

- F-Measure

  민감도와 양성예측도에 가중치를 둔 조화평균이다. 수식은 다음과 같다.

$$\begin{aligned} F\text{-measure} &= \frac{1}{\alpha \frac{1}{\text{recall}} + (1-\alpha)\frac{1}{\text{precision}}} \\ &= \frac{1}{\alpha \frac{1}{\text{sensitivity}} + (1-\alpha)\frac{1}{\text{PPV}}} \end{aligned}$$

  $\alpha = 0.5$를 사용하여 가중치를 동일하게 두어 계산한 값을 F1-점수라고 한다.

- Area under curve (AUC)

  위양성율(false positive rate)을 X축으로 민감도를 Y축 하는 그래프를 receiver operating characteristic (ROC) 그래프라고 한다. 이 그래프를 구하여 그 아래의 면적을 구한 것이 ROC 곡선의 AUC이다. 분류모형의 성능을 나타내는데 흔히 사용하는 지표로 1에 가까울수록 성능이 좋다고 할 수 있다.

2.6. 분류 모형

표본재추출을 완료한 이후 실제 분류를 위해서는 분류모형을 만들어야 한다. 본 논문에서는 대표적인 통계분류모형을 사용하여 표본재추출 작업 이후 분류를 하였다.

2.6.1. 로지스틱 회귀모형 이분형 자료의 분류를 위해 자주 사용되는 대표적인 방법이다. 변수선택 과정을 따로 거치지 않게 하기위해 LASSO 로지스틱 회귀모형을 적용하였다. Tuning parameter는 cross validation을 통해 값을 선택하였다.

2.6.2. Support vector machine (SVM) SVM에서는 kernel함수를 찾는 과정을 넣기도 하지만 본 논문에서는 kernel이 필요한 만큼 복잡한 데이터 상황을 가정하지 않기 때문에 kernel은 항등함수만을 사용하였다.

2.6.3. 랜덤 포레스트(random forest) 랜덤 포레스트는 의사결정나무 학습법(decision tree learning)을 이용한 방법이다. 본 연구에서는 나무의 개수를 100개로 설정하여 분석하였다.

3. 모의실험

세 가지 분류모형을 이용하여 불균형 자료의 분류를 할 때, 불균형 정도에 따른 각 재추출 방법들의 분류 성능 개선을 네 가지 상황의 모의실험을 통해 비교하였다. 본 논문에서는 표본재추출 방법들을 다음과 같이 약어를 사용하여 표현하였다.

- ADASYN: 오버샘플링 Adaptive Synthetic Sampling Approach 이용
- ROS: 오버샘플링 Random Over Sampler 이용
- SMOTE: 오버샘플링 Synthetic Minority Over-sampling Technique 이용
- SMOTE B1: 오버샘플링 Borderline SMOTE 1번째 변형 이용
- SMOTE B2: 오버샘플링 Borderline SMOTE 2번째 변형 이용
- CNN: 언더샘플링 Condensed Nearest Neighbour 이용
- ENN: 언더샘플링 Edited Nearest Neighbours 이용
- RENN: 언더샘플링 Repeated Edited Nearest Neighbours 이용
- AllKNN: 언더샘플링 All K-NN 이용
- IHT RF: 언더샘플링 Instance Hardness Threshold에 Random Forest 분류기 사용
- IHT SVM: 언더샘플링 Instance Hardness Threshold에 linear-SVM 사용
- NM1: 언더샘플링 Near Miss 1번째 변형 이용
- NM2: 언더샘플링 Near Miss 2번째 변형 이용
- NM3: 언더샘플링 Near Miss 3번째 변형 이용
- NCR: 언더샘플링 Neighbourhood Cleaning Rule 이용
- OSS: 언더샘플링 One Sided Selection 이용
- RUS: 언더샘플링 Random Under Sampler 이용
- TL: 언더샘플링 Tomek Links 이용
- SMOTE ENN: 혼합샘플링 SMOTE 적용 후 ENN 이용
- SMOTE TL: 혼합샘플링 STMOE 적용 후 TL 이용

모의실험을 통하여 자료를 생성하고 각 방법을 적용한 후 AUC, F1-점수, 정확도, 민감도를 이용하여 표본재추출 방법의 성능을 비교·평가하였다.

3.1. 모의실험 설계

3.1.1. 모의실험 1 본 모의실험에서는 선행논문 (Moon, 2018)과의 비교를 위하여 해당논문의 자료를 그대로 이용하였다. 표본의 크기는 2,000으로 하여 9개의 독립변수로 종속변수를 생성하였다.

$$X_1, X_2, X_3 \sim N(0,1)$$
$$X_4, X_5, X_6 \sim \text{Exp}(1)$$
$$X_7, X_8, X_9 \sim \text{Unif}(0,1)$$

$$\eta = \frac{\exp(X_1 - X_2) + X_4^3}{X_7} + \epsilon, \quad \epsilon \sim N(0,1),$$
$$Y = \begin{cases} 0, & \text{if } \eta_{\left(\frac{n}{a_1}\right)} < \eta < \eta_{\left(\frac{n}{a_2}\right)}, \\ 1, & \text{if o.w.}, \end{cases}$$

where

$$a_1 = \left(\frac{1-\mathrm{IR}}{1+\mathrm{IR}}\right)/2 \times 100, \qquad a_2 = \left(\frac{1+\mathrm{IR}}{1+\mathrm{IR}}\right)/2 \times 100.$$

독립변수 9개 중 4개(X_1, X_2, X_4, X_7)가 종속변수와 실제 연관이 있다. 불균형 정도는 imbalance rate (IR)를 통해서 조정하였고 IR의 범위는 9, 19, 29, 49, 99, 199로 설정하였다. 각 IR값에 대해 소수계급이 차지하는 비율을 백분율로 보면 $\{10\%, 5\%, 3\%, 2\%, 1\%, 0.5\%\}$이다.

생성된 데이터 중 모형의 학습에 사용되는 훈련자료(training set)와 평가에 사용되는 평가자료(test set)의 비율은 8 : 2로 하였다. 각 실험은 500번 반복하였고 분류 성능 평가 기준값은 500번 반복의 평균값을 이용하였다. 모의실험 자료를 생성하는 과정은 통계소프트웨어 R을 이용하여 진행하였다. 표본재추출 방법과 통계 모형을 적용하는 것은 파이썬 패키지인 scikit-learn을 이용하였다. 이는 네 모의실험 상황에 모두 동일하다.

3.1.2. 모의실험 2 모의실험 1에서 자료를 약간 수정하였다. 훈련자료의 숫자는 비슷하게 유지하면서 평가자료의 소수계급숫자를 늘리기 위해 자료를 2배로 늘리고 훈련과 평가자료 비율을 5 : 5로 하였다. 베르누이 분포를 따르는 독립변수를 3개 추가하였고, 이에 영향을 받도록 수식을 약간 변형하였다. 수식에서 나온 결과의 양 극단값을 케이스로 두었는데 이는 본 논문에서 사용하는 linear-SVM이나 로지스틱회귀분석에 많이 불리하다고 생각되어 한쪽 극단값만을 케이스로 분류하였다. 표본의 크기는 4,000으로 하였다.

$$\begin{aligned} X_1, X_2, X_3 &\sim N(0,1) \\ X_4, X_5, X_6 &\sim \mathrm{Exp}(1) \\ X_7, X_8, X_9 &\sim \mathrm{Unif}(0,1) \\ X_{10} &\sim B(1, 0.05) \\ X_{11} &\sim B(1, 0.01) \\ X_{12} &\sim B(1, 0.005) \end{aligned}$$

$$\eta = \frac{\exp(X_1 - X_2) + X_4^3 - X_{12}}{X_7 + X_{10}} + \epsilon, \quad \epsilon \sim N(0,1),$$
$$Y = \begin{cases} 0, & \text{if } \eta_{\left(\frac{n}{a_1}\right)} < \eta, \\ 1, & \text{if o.w.}, \end{cases} \quad \text{where } a_1 = \left(1 - \frac{\mathrm{IR}}{1+\mathrm{IR}}\right) \times 100.$$

3.1.3. 모의실험 3 이전 모의실험의 자료 생성과정이 다소 복잡하여 비교적 단순한 모형을 가정하고 모의실험을 진행했다. 독립변수는 연속형 변수와 범주형 변수 각 3개씩 총 6개 생성하였고, 변수들 간에 0.3의 상관계수를 두었다. 4개의 독립변수가 종속변수 생성에 관여하며 실제 선형적인 관계가 있

도록 설정하였다. 표본의 크기는 8,000으로 하였다.

$$\begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ \tilde{X}_4 \\ \tilde{X}_5 \\ \tilde{X}_6 \end{bmatrix} \sim \text{MVN}\left(\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \rho & \rho & \rho & \rho & \rho \\ \rho & 1 & \rho & \rho & \rho & \rho \\ \rho & \rho & 1 & \rho & \rho & \rho \\ \rho & \rho & \rho & 1 & \rho & \rho \\ \rho & \rho & \rho & \rho & 1 & \rho \\ \rho & \rho & \rho & \rho & \rho & 1 \end{bmatrix} \right), \quad \text{where} \quad \rho = 0.3$$

$$X_4 = \begin{cases} 0, & \text{if} \quad \tilde{X}_4 < \Phi^{-1}(0.3), \\ 1, & \text{if} \quad \text{o.w.}, \end{cases}$$

$$X_5 = \begin{cases} 0, & \text{if} \quad \tilde{X}_5 < \Phi^{-1}(0.2), \\ 1, & \text{if} \quad \text{o.w.}, \end{cases}$$

$$X_6 = \begin{cases} 0, & \text{if} \quad \tilde{X}_6 < \Phi^{-1}(0.15), \\ 1, & \text{if} \quad \text{o.w.}, \end{cases}$$

$$\eta = \frac{X_1}{2} + \frac{X_2}{4} + X_4 - X_6 + \epsilon, \quad \epsilon \sim N\left(0, 2^2\right)$$

$$Y = \begin{cases} 0, & \text{if} \quad \eta_{\left(\frac{n}{a_1}\right)} < \eta, \\ 1, & \text{if} \quad \text{o.w.}, \end{cases}, \quad \text{where} \quad a_1 = \left(1 - \frac{\text{IR}}{1+\text{IR}}\right) \times 100.$$

3.1.4. 모의실험 4 모의실험 3에서 종속변수를 결정하는데 사용되는 오차의 편차를 줄인 모형이다. 또한 종속변수를 만드는데 사용되는 독립변수의 계수값을 크게 하여 상대적으로 오차의 영향이 적어지도록 하였다. 표본의 크기는 8,000이다.

$$\begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ \tilde{X}_4 \\ \tilde{X}_5 \\ \tilde{X}_6 \end{bmatrix} \sim \text{MVN}\left(\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \rho & \rho & \rho & \rho & \rho \\ \rho & 1 & \rho & \rho & \rho & \rho \\ \rho & \rho & 1 & \rho & \rho & \rho \\ \rho & \rho & \rho & 1 & \rho & \rho \\ \rho & \rho & \rho & \rho & 1 & \rho \\ \rho & \rho & \rho & \rho & \rho & 1 \end{bmatrix} \right), \quad \text{where} \quad \rho = 0.3$$

$$X_4 = \begin{cases} 0, & \text{if} \quad \tilde{X}_4 < \Phi^{-1}(0.3), \\ 1, & \text{if} \quad \text{o.w.}, \end{cases}$$

$$X_5 = \begin{cases} 0, & \text{if} \quad \tilde{X}_5 < \Phi^{-1}(0.2), \\ 1, & \text{if} \quad \text{o.w.}, \end{cases}$$

$$X_6 = \begin{cases} 0, & \text{if} \quad \tilde{X}_6 < \Phi^{-1}(0.15), \\ 1, & \text{if} \quad \text{o.w.}, \end{cases}$$

$$\eta = 1.1X_1 + 0.9X_2 + X_4 - X_6 + \epsilon, \quad \epsilon \sim N\left(0, 2^2\right),$$

$$Y = \begin{cases} 0, & \text{if} \quad \eta_{\left(\frac{n}{a_1}\right)} < \eta, \\ 1, & \text{if} \quad \text{o.w.}, \end{cases} \quad \text{where} \quad a_1 = \left(1 - \frac{\text{IR}}{1+\text{IR}}\right) \times 100.$$

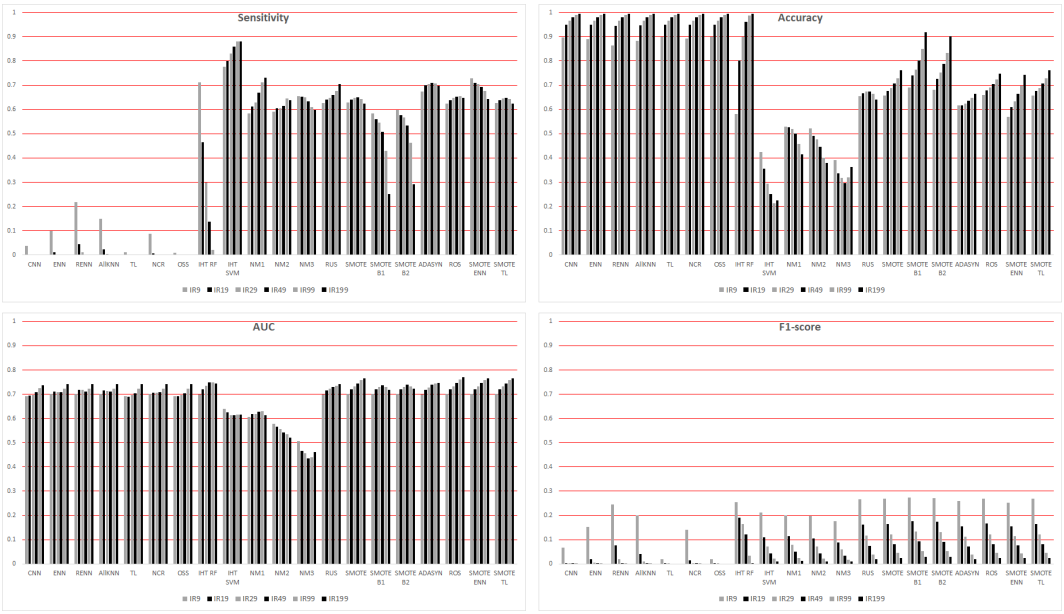

Figure 3.1. Sensitivity, accuracy, ACU, and F1-score of logistic regression for simulation 3.

3.2. 모의실험 결과

각 모의실험에서 비교하고자 하는 경우의 수는 3(분류기) ∗ 20(재추출 방법) ∗ 6(IR) = 360가지이다. 각 분류기에 대해 IR값에 따른 20개의 재추출 방법의 성능을 정리하고 분류기에 따른 차이가 있는지 살펴보았다. 계급불균형이 심한 자료에서는 희소 계급을 잘 예측할 수 있는 능력이 중요하다고 판단되기 때문에, 민감도, 정확도, AUC, F1-점수의 평가 기준 중에 민감도를 제일 중요하게 생각하였다. 모의실험 3의 결과를 Figures 3.1–3.3에 제시하였다. 다른 결과는 부록에 첨부하였다 (Figures A.1–A.9).

3.2.1. 모의실험 1 결과 로지스틱 회귀분석과 SVM은 집단을 둘로 나누는 분류방법이다. 모의실험 1은 둘로 나누어지지 않는 자료이므로 모든 분석 결과에서 AUC가 0.7 미만으로 낮게 나타났다. 로지스틱 회귀분석 결과, CNN계열과 IHT RF는 비슷한 양상을 보였다. IHT RF가 다소 좋은 결과를 나타냈다. 다른 방법들은 나머지 언더샘플링 방법들이 서로 비슷하고, 오버샘플링 방법들과 혼합 방법들이 비슷한 경향을 보였다. IHT RF를 포함한 CNN계열 방법들은 불균형 정도가 높아짐에 따라 민감도가 낮아져 논문에서 목표로 하는 결과를 내지 못하였다. IHT SVM를 포함한 NM, RUS 방법은 민감도가 가장 높은 결과를 나타내었다. 오버샘플링과 혼합방법은 SMOTE B1, B2 방법이 불균형 정도에 따라 민감도가 빠르게 떨어지지만 나머지 방법들은 IR = 199인 경우 0.3을 조금 넘었다. 민감도만 보면 IHT SVM이 가장 좋은 결과를 나타낸다. IHT SVM은 극단적으로 양쪽의 그룹을 나누기 때문에 민감도가 좋은 것으로 판단할 수 있다. NM 방법들의 민감도 결과는 좋았지만 Figure 2.3을 참고할 때 적절한 분류가 이루어졌다고 생각하기는 어렵다. 특별한 경우로 NM2 방법의 경우 양쪽으로 계급이 분리된 경우 우연히 분류가 되는 경우를 발견했다 (Figure 3.4).

정확도에 대한 결과를 살펴보면 불균형 정도가 높아지면서 정확도가 높아지는 것을 확인할 수 있다. 불균형 정도가 큰 자료이므로 희소계급값을 정확히 예측하지 못하면서 일어나는 현상으로 보인다. 민감도가 낮았던 그룹에서 공통적으로 높은 결과를 나타냈다. 반대로 민감도가 높았던 NM 방법들과 IHT

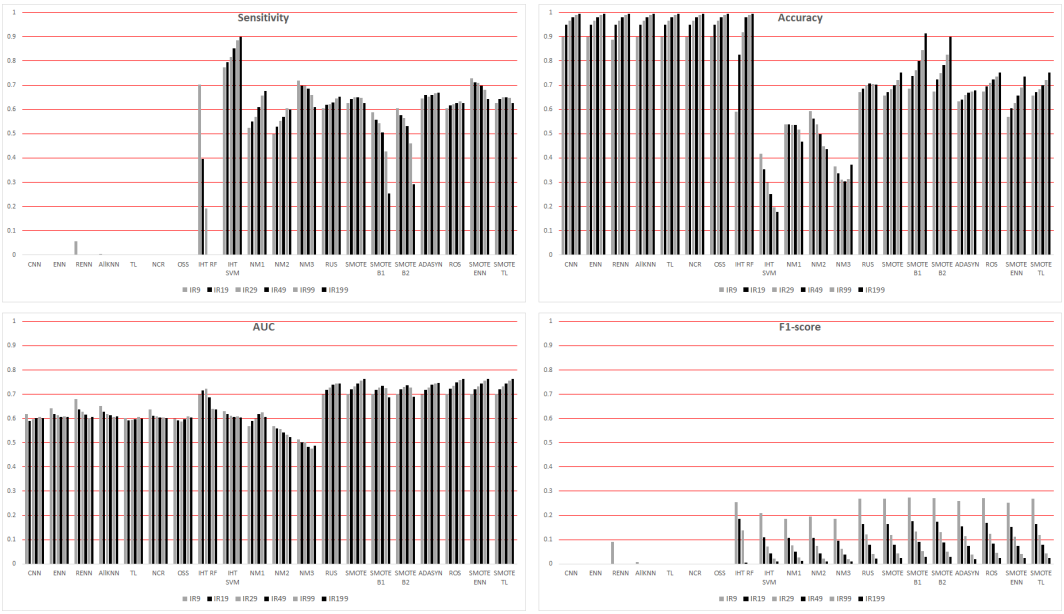

Figure 3.2. Sensitivity, accuracy, ACU, and F1-score of SVM for simulation 3.

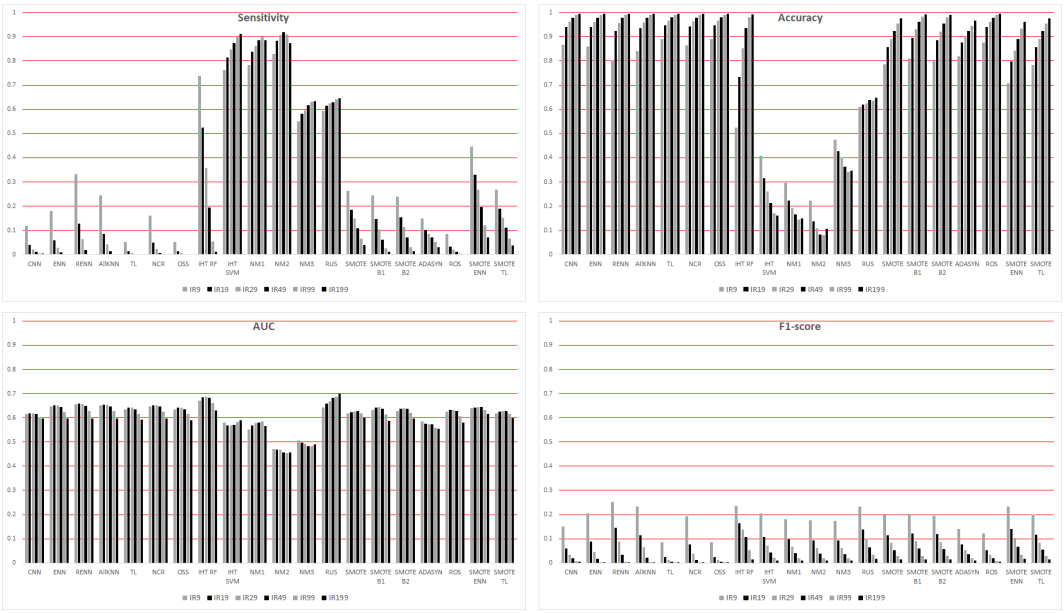

Figure 3.3. Sensitivity, accuracy, ACU, and F1-score of random forest for simulation 3.

SVM 방법들은 정확도가 0.5에 미치지 않는 결과를 나타냈다. 그 중에서 RUS만이 민감도와 정확도 모든 결과에서 0.5를 넘는 결과를 보여준다. AUC 결과는 일부 방법(NM2, NM3)에서는 불균형 정도가 커짐에 따라 0.5를 넘지 못하는 경우가 발생했다. F1-점수의 결과는 마찬가지로 매우 좋지 않았다. 민

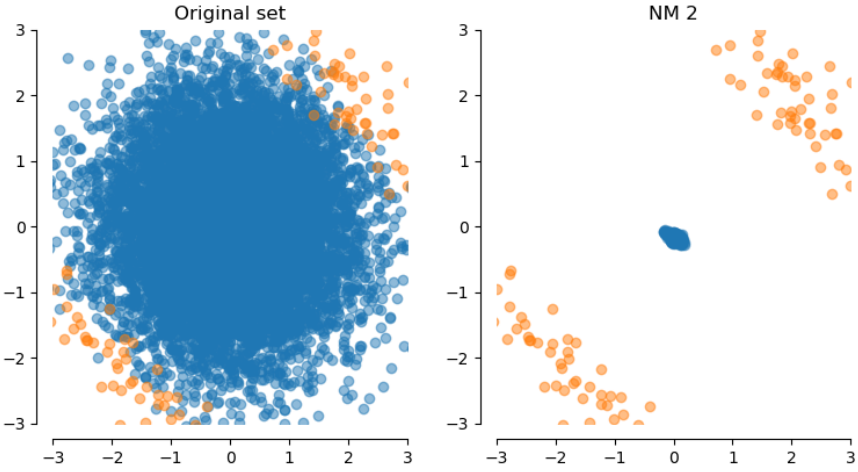

Figure 3.4. An example of original data set and changed data set after applying the NM2 method when the rare class values were distributed in two extremes.

감도가 0에 수렴했던 CNN계열과 IHT RF 방법의 결과가 0이였고 나머지 방법들도 IR199에서 0에 수렴하는 매우 낮은 결과가 나왔다.

SVM 결과는 대체로 로지스틱회귀분석의 결과와 비슷한 경향을 보였다. 전체적으로 더 극단적이고 결과 값이 나쁜 경우가 더 많았다. 랜덤포레스트의 결과는 모든 언더샘플링 방법들의 민감도가 로지스틱 회귀분석의 결과에 비해서 컸다. CNN, IHT RF 방법의 민감도가 0.1을 넘는 정도이고 나머지 CNN계열은 여전히 낮았다. IHT SVM, NM, RUS 방법의 결과는 로지스틱 회귀분석에서도 높았지만 더 높아졌다. 반면에 오버샘플링 방법들의 결과는 좋지 않았다. 민감도와 연관지어 정확도의 경향은 비슷했다. 민감도가 높았던 그룹에서는 정확도가 떨어졌지만 RUS 결과의 민감도와 정확도는 둘 다 상승했다. AUC 수치는 로지스틱 회귀분석, SVM에 비해 높아졌다. CNN계열의 결과가 IR = 199에서 0.7을 넘는 정도로 상승했다. 그중에서 CNN과 IHT RF 결과가 가장 좋았다. 민감도가 극단적으로 높았던 다른 언더샘플링 방법들의 AUC가 모두 0.5보다 큰 값을 가지게 되었다. 그 중에서 NM3와 RUS는 가장 좋은 결과를 보였다. F1-점수를 보면 로지스틱 회귀분석에 비해 다소 개선된 결과를 보여준다. 이 중에서는 IHT RF가 IR = 199에서 가장 높은 결과를 나타내었다.

종합적으로 볼 때 1차원으로 분류할 수 없는 상황에서도 민감도가 가장 높으면서 정확도도 0.5를 넘는 RUS 방법이 가장 좋다고 할 수 있다. 최대한 희소계급값을 찾아낸다고 생각하면 가정이 맞지 않는 경우에도 IHT SVM이 희소계급값을 가장 잘 찾아내었다. IHT RF는 CNN계열과 비슷한 경향을 보이면서 수치가 좋았다. NM 방법은 양쪽 계급값이 겹치는 부분이 크게 존재하는 경우에 적용하기 힘든 방법인 것 같다. 자료가 선형으로 분류해 내기 어려운 경우에는 가정 없이 랜덤으로 재추출 하는 것이 적절하다고 할 수 있다.

3.2.2. 모의실험 2 결과 모의실험 2에서는 표본의 크기가 두 배로 증가하고 수식의 한쪽만 희소계급값으로 정해 분류하기가 쉽도록 자료를 만들었다. 로지스틱 회귀분석 결과는 모의실험 1과 경향이 비슷했다. IHT RF의 결과는 불균형 정도가 낮을 때의 정확도와 F1-점수를 제외하고는 CNN계열의 결과값들보다 좋았다. 민감도가 낮고 F1-점수와 AUC의 값이 다른 결과에 비해서 좋았다. IHT SVM은 모의실험 1보다 극단적인 결과를 나타냈다. 민감도가 IR = 199에서 0.95를 넘어 대부분의 희소계급

를 맞추지만 정확도가 0.3 미만으로 낮아 희소계급 주변의 값들만 재추출 했다고 볼 수 있다. NM 방법은 민감도가 높고 정확도도 0.5를 넘었지만 비슷한 경향을 보이는 RUS가 대부분의 경우 더 좋은 결과를 나타냈다. 이번 모의실험에서도 RUS가 가장 좋은 방법으로 보인다. 오버샘플링 방법들 중에서는 ADASYN 방법은 RUS와 비슷한 결과를 보였고 나머지는 민감도가 다소 낮고 AUC와 F1-점수가 약간 좋은 수준이었다.

SVM의 결과는 로지스틱 회귀분석의 결과와 경향은 매우 비슷하고 전체적인 결과가 다소 낮은 부분에서 모의실험 1과 비슷했다. 모의실험 1에서는 대부분의 결과가 나빠졌던 것에 비해 모의실험 2의 RUS를 적용한 SVM 모형에서는 민감도가 다소 떨어지고 정확도가 조금 높아졌다.

랜덤포레스트 결과에서는 전체적인 경향은 선형분류기와 비슷하나 모의실험 1과 마찬가지로 오버샘플링의 민감도가 낮아진 것을 확인할 수 있다. NM 방법 중에서는 NM3의 정확도가 IR = 199일 때도 0.5를 넘었다. 모의실험 1의 랜덤포레스트 결과와 비교하여 IHT RF를 포함한 CNN계열의 민감도가 낮아지면서 AUC와 F1-점수가 낮아지는 경향을 보였다. 모의실험 1에서는 희소계급값의 집단중심이 2개에서 1개로 줄어들었으므로 언더샘플링하는 값이 더 줄어들어 점수가 낮아진 것으로 추정된다. SMOTE와 혼합모형의 F1-점수 값이 상위권에 위치했다. 민감도와 정확도를 위주로 보면 RUS가 눈에 띄게 좋은 결과를 나타냈다.

모의실험 2의 결과를 종합하여 민감도와 정확도의 관점에서 보면 RUS 방법이 가장 좋다. 이는 모의실험 자료의 분류가 복잡하기 때문에 아무런 가정 없이 랜덤샘플링 하는 것이 가장 좋은 방법이라고 생각된다. 불균형 정도가 큰 경우에 ROS는 단순히 값을 중복하여 뽑는다. 이 때문에 같은 랜덤샘플링인 RUS에 비해 ROS결과가 좋지 않았다고 추정된다.

3.2.3. 모의실험 3 결과 모의실험 3에서는 독립변수들의 1차식과 오차로만 종속변수가 결정되도록 하였다. 모의실험 3의 자료가 실제 로지스틱 회귀분석모형을 따르도록 만들었다. 따라서 로지스틱 회귀분석과 선형 SVM에서 좋은 성능을 낼 수 있을 것으로 생각되었다. 다만 현실에서의 데이터는 독립변수들끼리 완전히 독립이라 가정하기 어렵다. 이 때문에 상관계수가 존재하도록 설정하였다. 분류성능이 지나치게 높아지는 것을 피하기 위해 오차의 분산을 다소 높게 설정하였다.

로지스틱 회귀분석의 결과, 모의실험 1에서처럼 IHT RF를 포함한 CNN계열의 민감도가 0에 수렴했다. 이전 결과들과 마찬가지로 IHT SVM과 NM 방법은 민감도결과는 좋지만 정확도가 0.5에 미치지 못할 정도로 낮았다. RUS 방법과 오버샘플링, 혼합 방법은 민감도가 높으면서 일정 수준 이상의 정확도를 유지했다. AUC와 F1-점수의 결과도 가장 높았다. 그 중에서 RUS, ADASYN 방법은 민감도가 특히 높았다.

SVM의 결과는 모의실험 1, 2와 마찬가지로 로지스틱 회귀분석의 결과와 경향은 비슷하고 결과 수치가 다소 낮았다. 모의실험 2와 마찬가지로 RUS와 ADASYN으로 재추출한 후 SVM을 적용한 결과에서는 민감도가 다소 떨어지고 정확도가 다소 높아졌다.

랜덤포레스트 결과에서는 로지스틱 회귀분석, SVM 결과와 비슷한 경향을 보이지만 오버샘플링 방법의 민감도 결과는 다른 분류방법에 비해 떨어진다. 결국 모의실험 3에서 랜덤포레스트 분류기를 적용한 결과에서 민감도가 높으면서 어느 정도 정확도가 높은 방법은 RUS밖에 없었다. 경향은 비슷하지만 전체적인 점수가 로지스틱회귀분석, SVM에 비해 떨어진 것을 확인할 수 있었다. 모의실험 3은 실제 선형분류기로 분류할 수 있는 자료이므로 로지스틱 회귀분석, SVM 모형의 실행 결과가 좋은 것은 자명하다.

모의실험 3에서는 오버샘플링 방법이 좋은 성능을 냈다. 그중 ADASYN이 RUS와 비슷한 정도의 수치를 나타냈다. 여전히 RUS가 가장 좋은 방법 중 하나로 보인다. 모의실험 3에서 희소계급과 다수계급이

겹치는 범위가 넓었으므로 CNN계열의 방법은 특히 결과가 좋지 않을 것이라 예상할 수 있었다. 예상과 달리 IHT RF 방법도 CNN계열의 방법과 같은 경향을 보였다. RF분류기를 적용할 때 오버샘플링 방법은 결과가 좋지 않았다. 이는 좁은 공간에서 오버샘플링이 과도하게 일어나기 때문이라 추측된다.

3.2.4. 모의실험 4 결과 모의실험 4는 모의실험 3에서 설정한 경계의 오차의 분산이 너무 높아 분류가 적절히 이루어지지 않았다고 판단되어 오차의 분산을 줄이고 분류식을 다소 수정하였다. 모의실험 3과 결과가 다르지 않을 것이라 예상했다.

로지스틱 회귀분석 결과를 보면 민감도는 다른 것과 마찬가지로 RUS와 오버샘플링이 높은 것을 알 수 있다. 오버샘플링 방법들 중에서 SMOTE B2의 민감도가 특히 높았다. 그에 비해 NM 방법들의 민감도는 IR = 199일 때 0.1 미만으로 상대적으로 적게 증가하였다. IR = 199일 때의 정확도는 모든 오버샘플링 방법과 혼합 방법의 결과가 0.9 이상이였지만 RUS의 정확도는 0.9에 미치지 못했다. 민감도가 낮았던 것에 비해 NM 방법의 정확도는 0.9 근처로 높았다. IHT SVM만이 정확도가 좋지 않았지만 불균형 정도가 커짐에 따라 올라 IR = 199일 때는 0.6 이상이 되었다. AUC 결과는 모두 비슷했기 때문에 방법들 간의 비교가 어려웠다. IHT RF를 포함한 CNN계열은 민감도는 상대적으로 낮지만 F1-점수는 높았다. 민감도와 정확도 둘 다 높은 방법은 IHT SVM 다음으로 민감도가 높고 정확도가 0.8 이상인 RUS나 민감도와 정확도 둘 다 0.9를 넘는 SMOTE B2이었다. F1-점수는 이전까지의 방법들에 비해 IR = 199에서도 0.3을 넘는 분류기가 존재했다. 그중에서는 IHT RF 방법의 민감도가 가장 높았다.

다른 모의실험과 같이 SVM에서는 로지스틱 회귀분석과 경향이 비슷했다. SVM에서는 RUS의 정확도가 0.9를 넘었고 IHT SVM의 민감도가 1에 거의 수렴하여 정확도는 IR = 199에서 0.5를 조금 넘는 수준이었다. 로지스틱회귀분석과 달라진 부분은 IR = 199에서 F1-점수가 0.3이 넘는 방법이 IHT RF 하나라는 것이었다. IHT RF의 F1-점수가 약간 높아진 데 비해 다른 CNN계열 언더샘플링 방법들의 민감도가 떨어지면서 F1-점수도 같이 낮아졌다. 랜덤포레스트 방법에서는 오버샘플링 방법들의 민감도가 낮아졌기 때문에 정확도가 0.5 이상이면서 민감도가 가장 높은 방법은 RUS이었다. 선형 분류기와 또 하나 다른 점은 NM1, 2 방법의 정확도가 떨어졌다는 것이었다. 이와 마찬가지로 AUC도 낮아졌다. NM 방법은 Figure 3.4에서 확인한 것과 같이 본 논문의 모의실험에서는 적절치 않은 방법이라 판단된다. F1-점수 또한 선형 분류기에 비해 낮아졌다. 0.3을 넘는 방법이 존재하지 않았고, CNN계열의 재추출 방법과 함께 오버샘플링 혼합모형 방법의 F1-점수가 높은 그룹에 속하였다.

모의실험 4에서는 선형 분류기가 유리한 조건이라 랜덤포레스트의 성능은 좋지 않았다. 앞선 모의실험 결과들과 같이 정확도를 적절히 유지하면서 민감도를 높이는 방향으로 생각한다면 종합적으로는 RUS가 가장 좋은 선택이었다. 선형분류기를 사용한다면 오버샘플링 방법도 RUS와 마찬가지로 좋은 방법이며 SMOTE B2가 RUS보다 좋은 결과가 나왔다. 이외의 오버샘플링 방법은 좋은 방법이나 RUS보다 떨어진다고 볼 수 있었다. 다른 모의실험과 달리 모형의 분류성능이 향상되어서 F1-점수에 다소 의미를 둘 수 있었다. F1-점수로 보면 종합적으로 IHT RF 방법도 좋은 방법이라 할 수 있다. 하지만 가장 분류성능이 좋았던 로지스틱 모형에서의 성능이 CNN계열의 언더샘플링 방법들과 큰 차이는 없었기 때문에 IHT RF 방법이 좋다고 하기 어렵다.

4. 실제 자료 분석

이 절에서는 scikit-learn 파이썬 패키지에 예제 데이터로 포함되어 있는 자료를 분석한 결과를 비교하였다. 선행논문 (Moon, 2018)에서 사용한 자료(yeast_me2, mammography, abalone_19)와 추가로 세 가지 자료를 분석하였다 (Table 4.1). 각각의 자료에서 원 자료의 20%를 평가자료(test set)로 설정한 후

Table 4.1. Example data sets

Name	Repository	Class Ratio	Observation	Attributes
solar_flare_m0	UCI	19 : 1	1,389	32
car_eval_4	UCI	26 : 1	1,728	21
letter_img	UCI	26 : 1	20,000	16
yeast_me2	UCI	28 : 1	1,484	8
mammography	UCI	42 : 1	11,183	6
abalone_19	UCI	130 : 1	4,177	10

UCI: University of California Irvine Machine Learning Repository (https://archive.ics.uci.edu/ml/index.php)

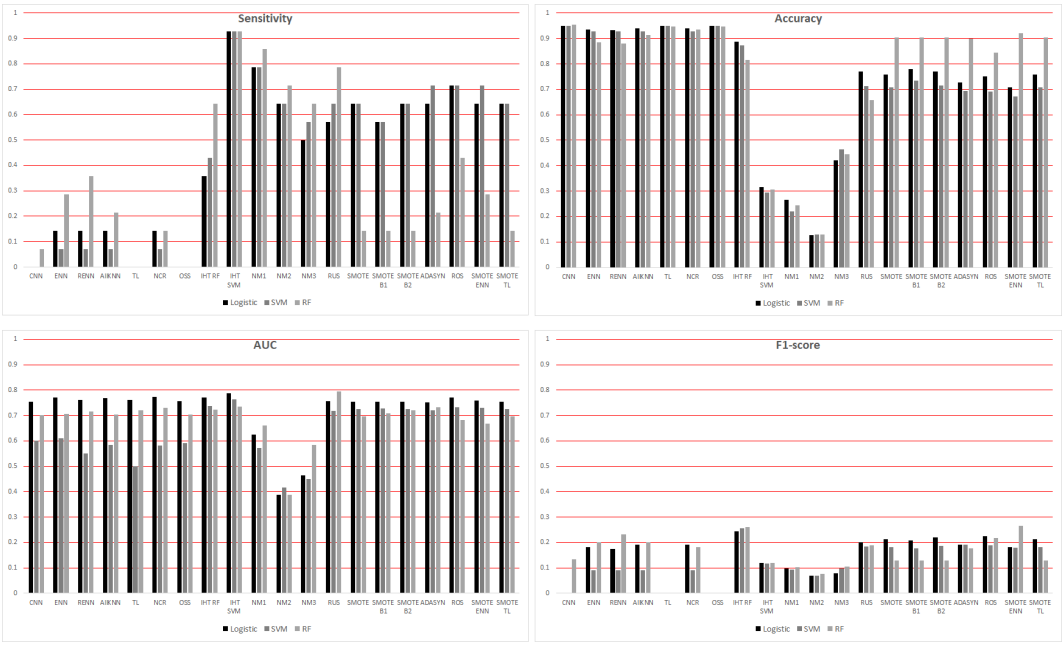

Figure 4.1. Sensitivity, accuracy, ACU, and F1-score of logistic regression, SVM, and random forest for solar_flare_m0 data.

80%의 자료로 학습하여 평가하였다. solar_flare_m0 자료 분석 결과는 Figure 4.1에 있고, 나머지 결과는 부록에 제시하였다 (Figures A.10–A.14).

4.1. 자료분석 결과

sola_flare_m0는 모의실험 결과들과 비슷한 양상을 보였다 (Figure 4.1). IHT RF의 랜덤포레스트에서 민감도 수치가 0.5 이상으로 높게 나타났다. RUS보다 오버샘플링 방법들이 좋은 결과를 보였다. CNN계열의 민감도 결과가 좋지 않았다.

car_eval_4는 모든 방법에서의 AUC 결과가 0.9 이상이고, 민감도 결과가 0.6 이상으로 높았다 (Figure A.10). 민감도와 정확도에서는 방법들 간의 큰 차이가 없고, F1-점수를 보면 CNN계열 방법들이 좋은 결과를 보였다.

letter_img는 모든 방법에서의 AUC 결과가 0.9 이상이고, 민감도 결과가 0.6 이상으로 높았다 (Figure A.11). RF분류기가 좋은 수치를 보였다. CNN계열의 방법들과 오버샘플링, 혼합 방법 모두 랜덤포레스트 분류기에서 좋은 수치를 보였다.

yeast_me2는 CNN계열의 민감도 결과가 전부 0.5에 미치지 않았다 (Figure A.12). RUS와 오버샘플링 계열의 민감도가 높았다. F1점수는 일부 CNN계열과 오버샘플링 방법들이 상대적으로 높은 값을 가졌다.

mammography의 경우, 오버샘플링, RUS의 민감도 결과에서 0.8 이상인 결과가 있었다 (Figure A.13). 정확도가 0.5를 넘기는 결과 중 가장 수치가 높은 결과였다. F1점수는 CNN계열의 방법들과 오버샘플링에 랜덤포레스트 분류기를 사용한 방법들의 수치가 0.4 이상이었다.

abalone_19의 경우, 정확도가 0.5 이상이면서 민감도가 0.5 이상인 결과를 가진 방법은 RUS, 오버샘플링 방법이었다 (Figure A.14). CNN계열 방법들의 민감도는 0이지만 AUC 결과는 높았다.

종합하면 CNN계열의 결과가 모의실험에 비해 좋았다. IHT RF도 CNN계열의 결과와 비슷하면서 민감도가 높은 경향을 보였다. IHT SVM과 NM 방법은 정확도가 0.5에 미치지 못하는 경우가 많았다. RUS 방법은 민감도와 정확도가 동시에 높은 경향을 보였다. 그에 비해 F1-점수는 다른 방법들과 비교하여 높지 않았다. 오버샘플링 방법은 RUS 방법과 비슷한 패턴을 보였다. 모의실험에 비해 오버샘플링 방법이 RUS에 비해 떨어지지 않고 높은 수치를 보이는 상황이 발생했다. 불균형 정도가 모의실험에 비해 높지 않았다. 이로 인해 결과가 다소 개선되었다. 경향은 모의실험과 비슷했으나 모의실험 상황에 비해 개발된 방법들이 좋은 수치를 나타냈다. 극단적인 경우일수록 RUS가 좋은 결과를 보였다.

5. 결론 및 고찰

불균형 자료의 분류기 성능을 향상시키기 위한 적절한 표본재추출 방법을 찾기 위해 4가지 상황에서 모의실험을 해보고 실제 자료를 분석해 보았다. 모두 20가지의 표본 재추출 방법을 비교하였다. 각 방법들의 분류 성능에 어떤 영향을 주는지 알고자 하였다. 본 논문에서는 희소한 계급을 찾아내는 것을 목적으로 하여 민감도를 주 평가척도로 해석하였다. 정확도가 0.5에 미치지 못하는 모형은 의미가 없다고 보았다. 보조 지표로서 AUC와 F1-점수를 이용하였다. 정확도가 낮은 경우 AUC도 낮은 경향이 있었지만, AUC 값은 표본재추출 방법에 따른 차이가 다른 지표들에 비해 크지 않았다. 앞서 언급하였듯이 본 논문에서는 희소 계급에 초점을 맞추어 민감도를 중요하게 생각하였지만, 다른 목적으로 분류기의 성능을 평가할 때에는 AUC가 중요한 지표가 될 수 있다.

모의실험에서는 IR = 9부터 IR = 199까지의 불균형정도를 가지는 자료를 분석하였다. 많은 방법에서 불균형 정도가 커짐에 따라 민감도가 0에 수렴하거나 정확도가 0.5 보다 낮아지는 현상이 발생했다. 같은 상황이더라도 자료의 불균형 정도에 따라 적합한 방법이 다를 수 있음을 알 수 있었다. 자료의 특성에 따라 표본 재추출 방법이 가지는 특성을 확인해 볼 수 있었다. 가정한 모의실험의 특성에 따라 적합하지 않은 방법들이 다수 존재했다.

오버샘플링과 혼합 방법은 전체적으로 비슷한 경향을 보이면서 ROS가 특별한 차이를 보여주지는 못했다. 방법들 중에는 ADASYN이 좋은 결과를 나타냈다. SMOTE B1, 2는 다른 오버샘플링과 결과의 양상이 조금 다르게 나타났다. CNN과 그 알고리즘에서 파생된 재추출 방법은 다수계급과 소수계급의 경계가 뚜렷할 때 경계를 찾을 수 있는 알고리즘이다. 이 논문의 모의실험은 경계가 뚜렷하지 않고 항상 다수계급의 분포가 압도적인 상황을 가정하였다. 이에 따라 결과가 좋지 않았던 것이라 생각된다. 따라서 모의실험 상황이 간단해 질수록 결과가 좋아졌다. 분류가 쉬운 데이터의 경우에는 CNN 결과도 나쁘지 않았으며 특히 F1-점수가 높은 경향을 보였다. 하지만 CNN계열과 비슷한 경향을 보이면서 결과가

좋은 IHT RF 방법이 있었기 때문에 본 논문의 상황만을 본다면 CNN과 관련 방법들이 좋다고 하기는 어려웠다.

IHT SVM 방법은 Figure 2.3에서도 나타나듯이 양 극단으로 데이터를 모아 재추출 한다. 이로 인해 경계를 알기 어려워 정확도가 크게 떨어지는 문제가 있다. 같은 재추출 실험을 반복해보면 IHT SVM으로 인한 표본 재추출 결과들 간의 편차가 크다. 안정적으로 사용하기 위해서는 IHT 방법을 적용하는데 SVM분류기만을 사용하는 것은 좋은 방법이 아니었다. NM 방법은 민감도는 높게 나오지만 대부분의 경우 정확도가 0.5 미만으로 낮은 수치를 나타냈다. 특히 Figure 2.3을 보면 NM1, 2, 3 모든 방법이 적절한 표본을 추출하지 못한다고 할 수 있었다. 본 논문에서 확인한 예제와 모의실험 자료에는 NM 방법은 부적절하다고 판단할 수 있다. RUS 방법은 본 논문의 실험에서 대부분 좋은 결과를 보였다. 다른 방법들에 비해 민감도가 높았다. F1-점수는 상대적으로 낮았다. 따라서 논문에서 찾고자 하는 민감도를 높이는 방법이 RUS라고 할 수 있다. 임의로 자료를 선택하는 것이기 때문에 특별한 가정 없이 사용할 수 있는 장점이 있다.

본 논문에서는 IHT 방법을 적용할 때 하나의 분류기만 사용하여 각각 적용시켰지만 성능을 향상시키기 위해 여러 가지 다른 시도를 할 수 있을 것이다. 추가적으로 본 논문에서는 파이썬 패키지에 내장되어 있는 혼합 방법을 사용하였다. 새로운 방법을 개발하는 것과 같이 적절한 재추출 방법의 조합을 찾는 것에 대해서도 연구할 필요가 있을 것이다.

부록

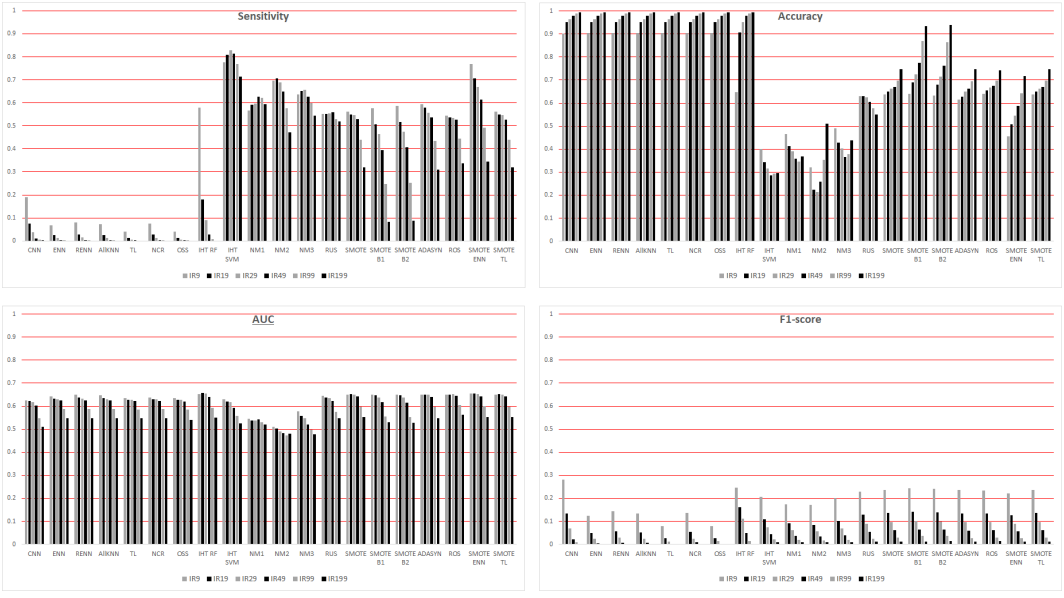

Figure A.1. Sensitivity, accuracy, ACU, and F1-score of logistic regression for simulation 1.

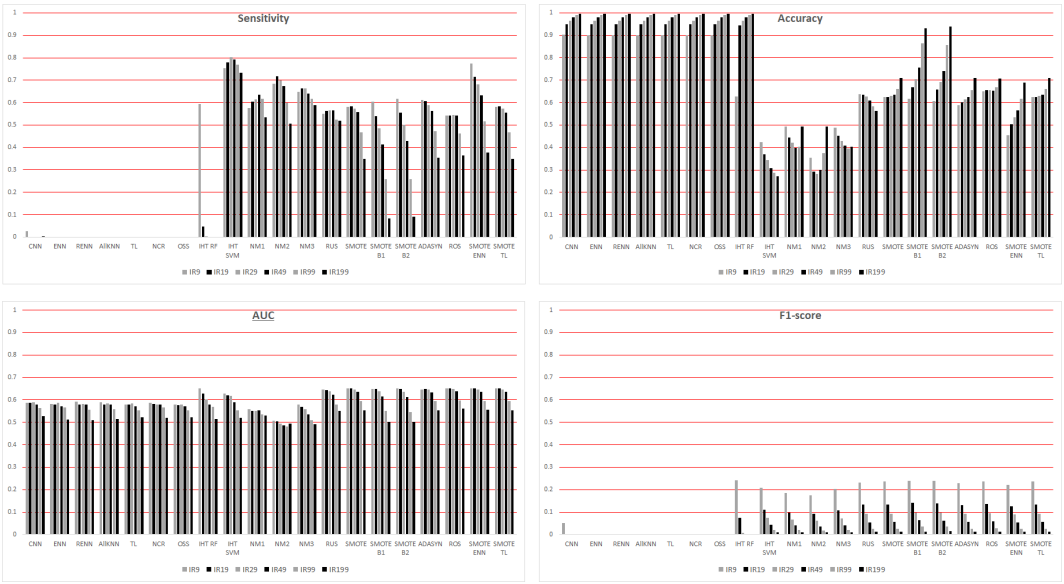

Figure A.2. Sensitivity, accuracy, ACU, and F1-score of SVM for simulation 1.

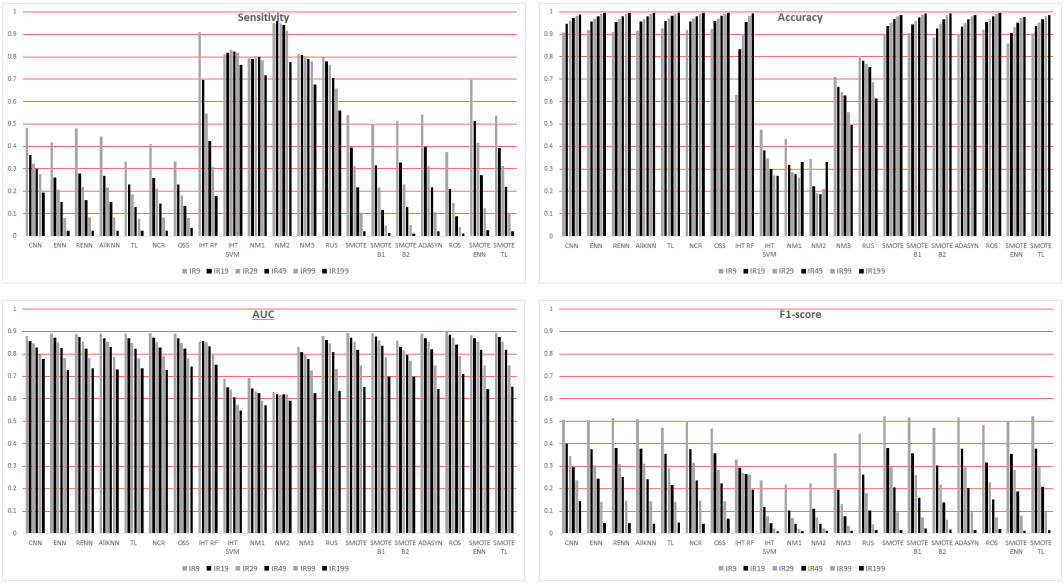

Figure A.3. Sensitivity, accuracy, ACU, and F1-score of random forest for simulation 1.

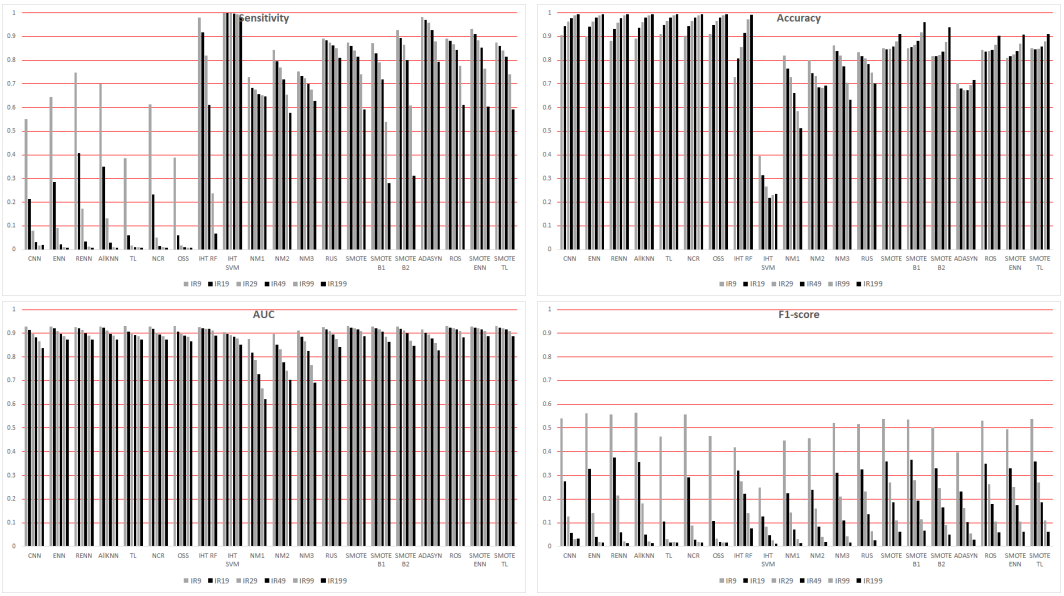

Figure A.4. Sensitivity, accuracy, ACU, and F1-score of logistic regression for simulation 2.

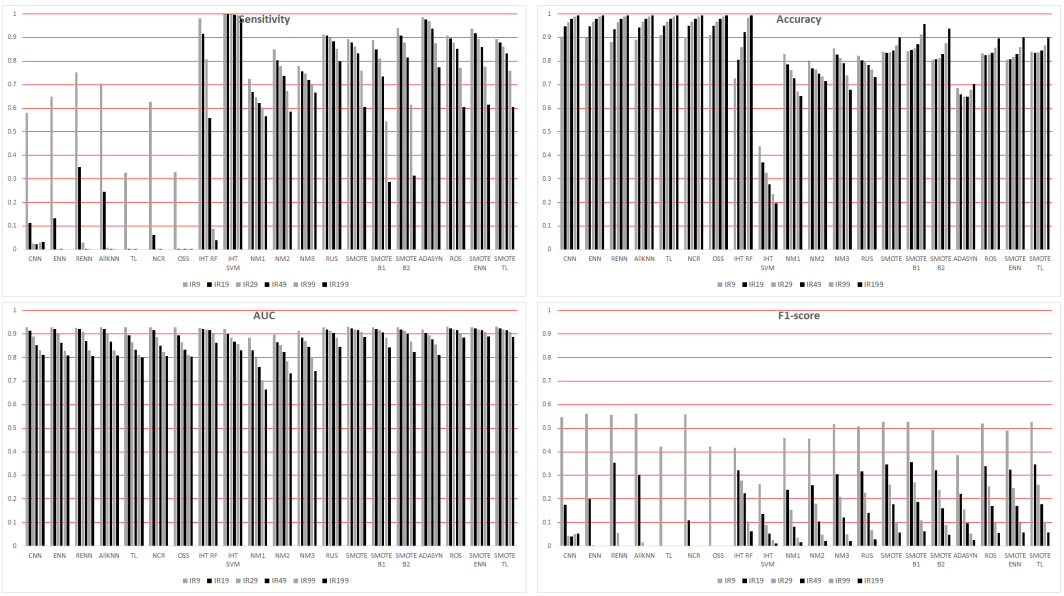

Figure A.5. Sensitivity, accuracy, ACU, and F1-score of SVM for simulation 2.

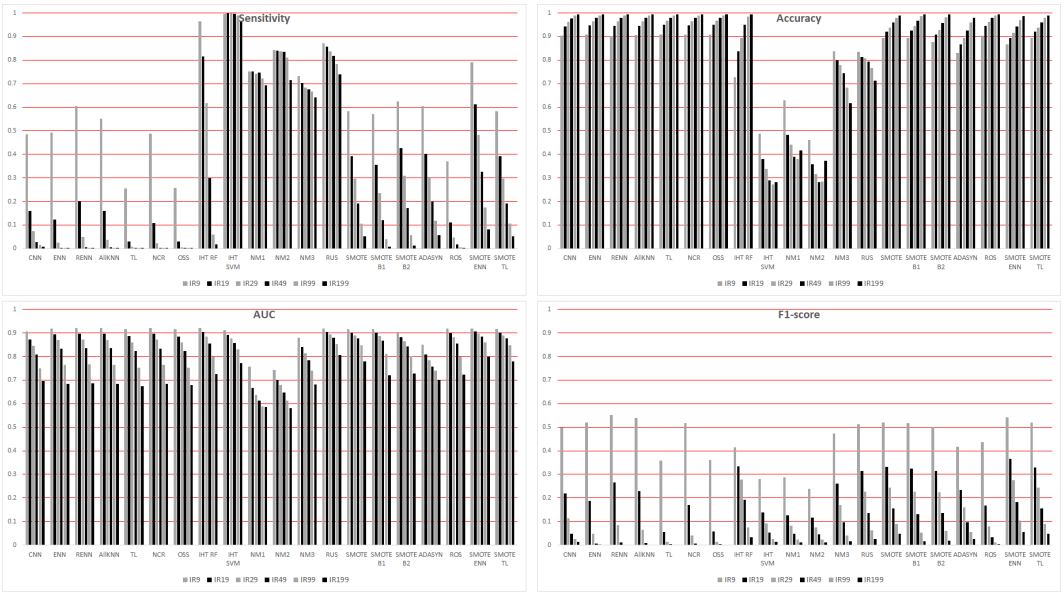

Figure A.6. Sensitivity, accuracy, ACU, and F1-score of random forest for simulation 2.

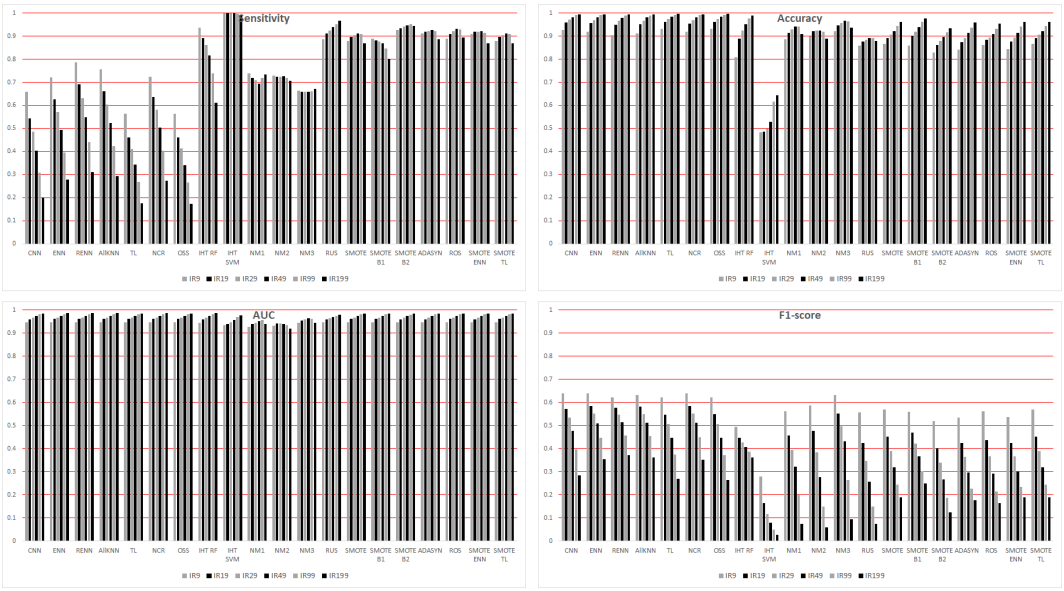

Figure A.7. Sensitivity, accuracy, ACU, and F1-score of logistic regression for simulation 4.

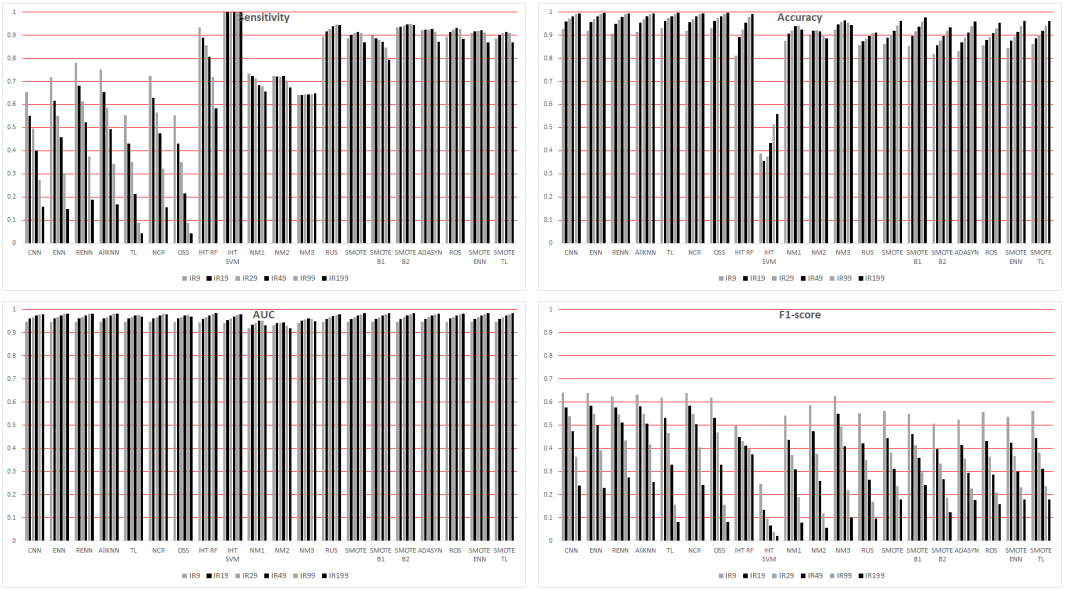

Figure A.8. Sensitivity, accuracy, ACU, and F1-score of SVM for simulation 4.

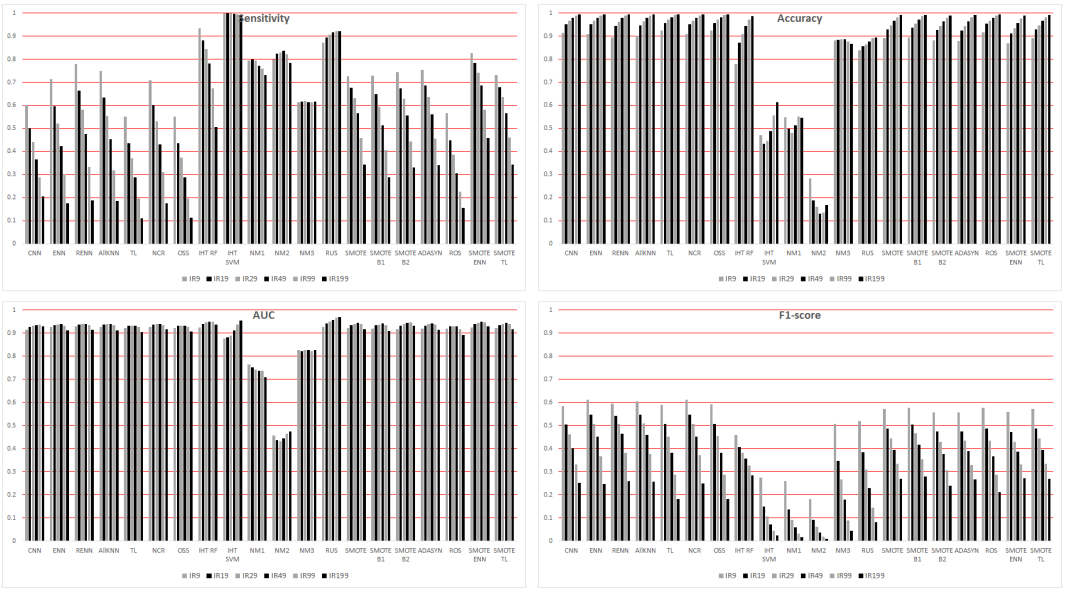

Figure A.9. Sensitivity, accuracy, ACU, and F1-score of random forest for simulation 4.

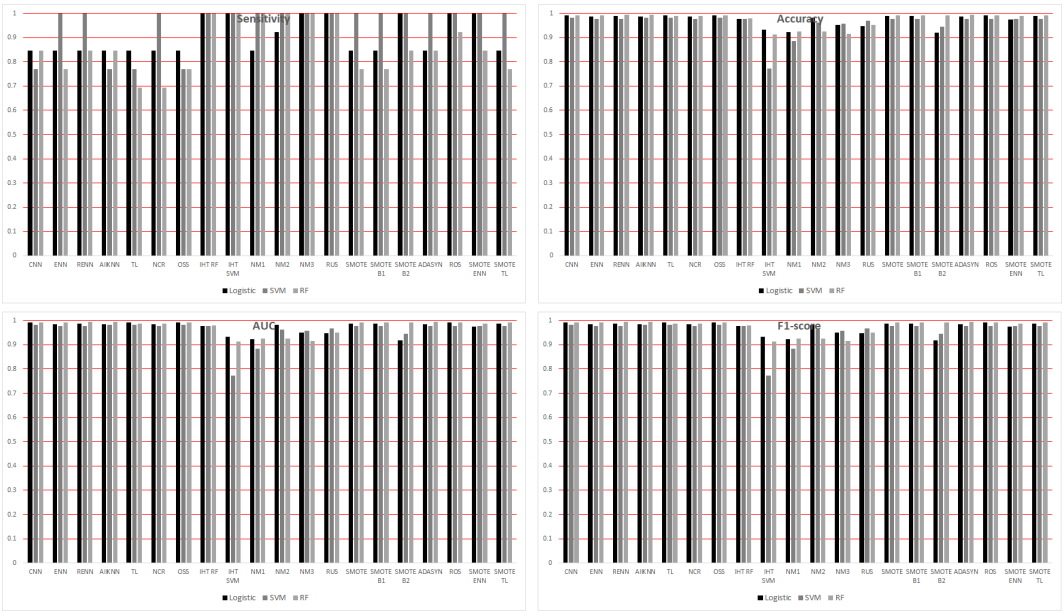

Figure A.10. Sensitivity, accuracy, ACU, and F1-score of logistic regression, SVM, and random forest for car_eval_4 data.

References

Batista, G. E., Prati, R. C., and Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data, *ACM SIGKDD Explorations Newsletter*, **6**, 20–29.

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.

Gates, G. (1972). The reduced nearest neighbor rule (Corresp.), *IEEE Transactions on Information Theory*, **18**, 431–433.

Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., and Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications, *Expert Systems with Applications*, **73**, 220–239.

Han, H., Wang, W. Y., and Mao, B. H. (2005). Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In *International Conference on Intelligent Computing* (pp. 878–887). Springer, Berlin, Heidelberg.

Hart, P. (1968). The condensed nearest neighbor rule (Corresp.), *IEEE Transactions on Information Theory*, **14**, 515–516.

He, H., Bai, Y., Garcia, E. A., and Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference* (pp. 1322–1328). IEEE.

He, H. and Garcia, E. A. (2009). Learning from imbalanced data, *IEEE Transactions on Knowledge and Data Engineering*, **21**, 1263–1284.

Laurikkala, J. (2001). Improving identification of difficult small classes by balancing class distribution. In *Conference on Artificial Intelligence in Medicine in Europe* (pp. 63–66).

Lemaître, G., Nogueira, F., and Aridas, C. K. (2017). Imbalanced-learn: a python toolbox to tackle the curse of imbalanced datasets in machine learning, *Journal of Machine Learning Research*, **18**, 1–5.

Mani, I. and Zhang, I. (2003). kNN approach to unbalanced data distributions: a case study involving information extraction. In *Proceedings of Workshop on Learning from Imbalanced Datasets II, ICML* (Vol. 126), Washington.

Moon, S. Y. (2018). *Performance comparison of classification methods based on the random forest in class imbalanced data* (Master thesis), Korea University, Seoul.

Prati, R. C., Batista, G. E., Monard, M. C. (2009). Data mining with imbalanced class distributions: concepts and methods. In *Proceedings of the 4th Indian International Conference on Artificial Intelligence* (pp. 359–376), Tumkur, Karnataka.

Smith, M. R., Martinez, T., and Giraud-Carrier, C. (2014). An instance level analysis of data complexity, *Machine Learning*, **95**, 225–256.

Tomek, I. (1976a). An experiment with the edited nearest-neighbor rule, *IEEE Transactions on Systems, Man, and Cybernetics*, **6**, 448–452.

Tomek, I. (1976b). Two modifications of CNN, *IEEE Transactions on Systems, Man, and Cybernetics*, **6**, 769–772.

Wilson, D. L. (1972). Asymptotic properties of nearest neighbor rules using edited data, *IEEE Transactions on Systems, Man, and Cybernetics*, **2**, 408–421.

이분형 자료의 분류문제에서 불균형을 다루기 위한 표본재추출 방법 비교

박근우[a] · 정인경[a,1]

[a]연세대학교 의과대학 의생명시스템정보학교실 의학통계학과

(2019년 2월 15일 접수, 2019년 4월 2일 수정, 2019년 4월 2일 채택)

요 약

이분형 자료의 분류에서 자료의 불균형 정도가 심한 경우 분류 결과가 좋지 않을 수 있다.이런 문제 해결을 위해 학습 자료를 변형시키는 등의 연구가 활발히 진행되고 있다. 본 연구에서는 이러한 이분형 자료의 분류문제에서 불균형을 다루기 위한 방법들 중 표본재추출 방법들을 비교하였다. 이를 통해 자료에서 희소계급의 탐지를 보다 효과적으로 하는 방법을 찾고자 하였다. 모의실험을 통하여 여러 오버샘플링, 언더샘플링, 오버샘플링과 언더샘플링 혼합 방법의 총 20가지를 비교하였다. 분류문제에서 대표적으로 쓰이는 로지스틱 회귀분석, support vector machine, 랜덤포레스트 모형을 분류기로 사용하였다. 모의실험 결과, 정확도가 0.5 이상이면서 민감도가 높았던 표본재추출 방법은 random under sampling (RUS)였다. 그 다음으로 민감도가 높았던 방법은 오버샘플링 ADASYN (adaptive synthetic sampling approach)이었다. 이를 통해 RUS 방법이 희소계급값을 찾기 위한 방안으로는 적합했다는 것을 알 수 있었다. 몇 가지 실제 자료에 적용한 결과도 모의실험의 결과와 비슷한 양상을 보였다.

주요용어: 불균형 학습, 불균형 이분형 자료, 언더샘플링, 오버샘플링

[1]교신저자: (03722) 서울 서대문구 연세로 50-1, 연세대학교 의과대학 의생명시스템정보학교실 의학통계학과.
E-mail: ijung@yuhs.ac