The Korean Journal of Applied Statistics (2017) DOI: https://doi.org/10.5351/KJAS.2017.30.5.681
30(5), 681–690

On sampling algorithms for imbalanced binary data: performance comparison and some caveats

HanYong Kim[a] · Woojoo Lee[a,1]

[a]Department of Statistics, Inha University

(Received July 17, 2017; Revised September 2, 2017; Accepted September 12, 2017)

Abstract

Various imbalanced binary classification problems exist such as fraud detection in banking operations, detecting spam mail and predicting defective products. Several sampling methods such as over sampling, under sampling, SMOTE have been developed to overcome the poor prediction performance of binary classifiers when the proportion of one group is dominant. In order to overcome this problem, several sampling methods such as over-sampling, under-sampling, SMOTE have been developed. In this study, we investigate prediction performance of logistic regression, Lasso, random forest, boosting and support vector machine in combination with the sampling methods for binary imbalanced data. Four real data sets are analyzed to see if there is a substantial improvement in prediction performance. We also emphasize some precautions when the sampling methods are implemented.

Keywords: imbalanced binary data, sampling, classifier, prediction

1. 서론

파산 감지, 스팸메일 감지, 불량품 감지 등은 우리 주변에서 쉽게 접할 수 있는 이항 자료 분류 문제이다 (Galarr 등, 2012). 이러한 자료의 주요 특성은 반응변수에서 0의 비율이 1의 비율에 비해 매우 높다는 사실인데, 반응변수의 불균형한 비율이 분류 모형의 성능에 문제를 준다고 알려져 있다 (Longadge와 Dongre, 2013). 예를 들어 0의 비율이 90%일 때 모든 예측을 0으로 하는 분류 모형의 정확도는 90%이기 때문에 마치 좋은 분류 모형인 것처럼 보일 수 있으나, 실제로는 1에 대한 예측 능력이 전혀 없는 적절하지 않은 모형이다. 이에 불균형한 상태에서의 분류 모형의 성능을 개선시키고자 그 동안 다양한 샘플링 방법이 제안되었다 (He와 Ma, 2013). 특히 주목할 만한 방법은 1을 여러번 복제하는 오버 샘플링(over-sampling) 방법, 0을 랜덤하게 제거하는 언더 샘플링(under-sampling) 방법, 오버 샘플링과 언더 샘플링을 합성하여 만든 synthetic minority over-sampling technique (SMOTE) (Chawla 등, 2002)의 3가지 방법이다.

본 연구에서는 이항 자료 분류 모형으로 많이 사용되는 기계 학습모형인 로지스틱 회귀모형, Lasso, 랜덤 포레스트, 부스팅, 서포트 벡터 머신(support vector machine; SVM)에 위의 3가지 샘플링 기법을

This research was supported by a grant [MOIS-DP-2015-05] through the Disaster and Safety Management Institute funded by Ministry of the Interior and Safety of Korean government.
[1]Corresponding author: Department of Statistics, Inha University, 100 Inharo, Nam-gu Incheon 22212, Korea. E-mail: lwj221@gmail.com

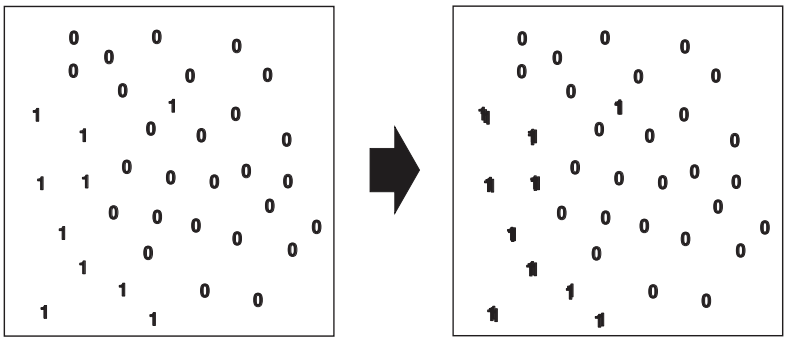

Figure 2.1. Oversampling: replication of 1.

적용하여 분류 성능의 개선 여부를 사례 연구를 통해 살펴보고자 한다. 특히 이 과정에서 우리는 기존의 많은 문헌들이 위의 3가지 샘플링 기법을 잘못 적용하는 여러 사례를 살펴볼 수 있었다 (Ren 등, 2015). 따라서, 본 논문에서는 실제 샘플링 방법을 사용할 때 쉽게 실수할 수 있는 부분을 먼저 지적하고, 잘못 적용되었을 때 나타나는 문제점에 대해 논의하고자 한다. 이는 네 개의 실제 이항 자료 분류 문제를 통해서 구체적으로 설명될 것이다.

본 논문은 2절에서 불균형한 이항자료를 분석하기 위해 제안된 3가지 샘플링 방법에 대해서 설명한 후, 3절에서 각 샘플링 방법이 사용될 때 주의해야할 점에 대해서 구체적으로 살펴본다. 결론은 4절에서 주어진다.

2. 샘플링 기법

불균형한 이항 자료를 분석할 때 가장 널리 사용되는 3가지 샘플링 방법인 오버샘플링, 언더 샘플링, SMOTE를 순서대로 살펴본다.

2.1. 오버샘플링

오버샘플링의 경우 Figure 2.1과 같이 클래스 1인 데이터를 복제함으로써 불균형 문제를 해결하는 것이다 (Longadge와 Dongre, 2013). 예를 들어, 0의 개수가 90개이고 1의 개수가 10개인 경우 1을 복제하여 90개로 만들어 전체 자료의 개수는 180개가 되고, 0과 1의 비율은 1 : 1이 되도록 하는 방법이다. 일반적으로는 1 : 1 대신 다른 비율이 되도록 조절하는 것 또한 가능하다. Figure 2.1에서 클래스 1의 복제를 표현하기 위해, 1을 약간의 지터링(jittering)을 주어 표현하였다. 그러나 오버샘플링은 원래 데이터의 수가 많을 때에서는 데이터의 수가 더 늘어나게 되어 모형구축에 시간이 더 걸린다는 단점이 있고, 과적합(overfitting)의 문제가 있을 수 있다는 지적을 받았다 (He와 Garcia, 2009).

2.2. 언더샘플링

언더샘플링의 경우 반응변수가 0인 클래스를 랜덤하게 제거하여 데이터의 불균형 문제를 해결하는 것이다 (Longadge와 Dongre, 2013). Figure 2.2는 언더샘플링의 방법을 설명하는 것인데, 0을 랜덤하게 제거하여 0과 1의 비율이 1 : 1이 되도록 맞춰준다. 일반적으로는 1 : 1 대신 다른 비율이 되도록 조절하는 것 또한 가능하다. 언더샘플링은 이처럼 데이터를 없애는 방법이기 때문에 정보손실이라는 문제가 생길

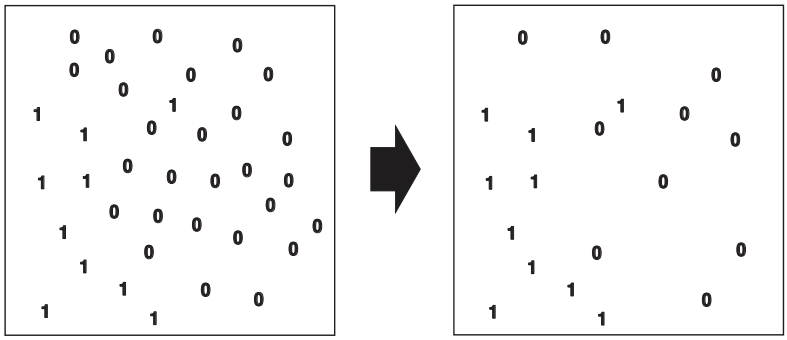

Figure 2.2. Undersampling: removing 0 randomoly.

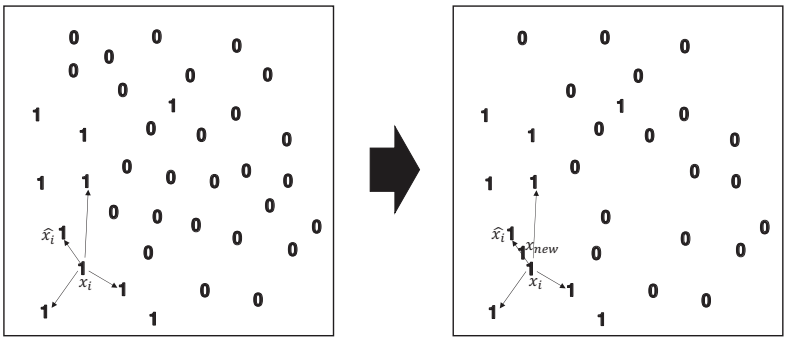

Figure 2.3. SMOTE: synthesis of over-sampling and under-sampling.

수 있다.

2.3. SMOTE

SMOTE는 오버샘플링과 언더샘플링을 합성한 방법이다. 먼저 반응변수의 클래스가 1인 데이터의 개수를 증가시켜주는 오버 샘플링 부분을 살펴보자. 먼저 설명변수 x_i를 갖는 클래스 1인 데이터를 하나 생각해보자. 그러면 이 데이터로 부터 구해진 k개의 클래스 1인 근접 이웃을 찾는 것이 첫 단계이다. Chawla 등 (2002)에서는 k를 5로 선택하였는데, 문제에 따라 다른 k를 선택하는 것이 가능하다. 일단 k개의 클래스 1인 근접 이웃을 찾으면, 이들 가운데 랜덤하게 하나를 선택한다. 이 선택된 데이터의 설명변수를 $\hat{x}_i$이라고 하자. 새롭게 생성되는 점의 설명변수 x_{new}는 x_i와 $\hat{x}_i$을 잇는 선분에서 임의로 뽑은 한 점이 되는데, 이는

$$x_{\text{new}} = x_i + (\hat{x}_i - x_i) \times \delta \tag{2.1}$$

으로 표현된다. 여기서, δ는 0과 1사이의 값에 균일하게 분포하는 랜덤 변수이다. x_{new}에서 클래스 1인 자료가 추가되는데, 이 과정을 사용자가 지정해주는 만큼 반복하여 클래스 1의 자료의 수를 늘리게 된다. 주목할만한 점은 앞서 설명하였던 오버샘플링과는 다르게 Figure 2.3과 같이 기존의 데이터와 같은 위치가 아닌 약간 이동된 클래스 1인 점들을 추가하는 방식으로 동작한다. 이를 통해 SMOTE는 기존의 오버샘플링의 오버피팅의 문제를 일부 개선해준다고 알려져 있다 (Chawla 등, 2002).

언더샘플링 부분은 사용자가 지정한 0과 1의 비율이 맞춰지도록, 0을 랜덤하게 제거해나가는 방법을 사용한다. 기타 SMOTE의 세부적인 알고리즘은 Chawla 등 (2002)에 자세히 설명되어 있다.

3. 예측 성능 비교시 주의점과 분석결과

3.1. 예측력 평가 방법

본 연구에서는 Autopart, Page-Black, German Credit, Secom의 네가지 실제 데이터를 분석한다. 각 자료를 다운로드 받을 수 있는 웹 주소는 부록 A에 주어져있다. Autopart 데이터는 자동차 부품의 생산데이터로서 12개의 설명변수를 가지고 있다. 반응변수는 oil gasket의 탕구 길이에 따라 불량 여부를 나타내는 이항 자료이며, 설명변수는 separation, mpa와 같이 공정에 관한 변수들로 구성되어있다. 클래스 1의 비율은 10% 정도이고, 전체 샘플수는 21,767개이다. Page-Balck 데이터는 문서의 페이지 레이아웃의 블락을 분류하는 것을 반응 변수로 하고, 11개의 설명변수를 가지고 있다. 설명변수에는 블락의 길이와 넓이, 위치에 관한 블락의 정보로 구성되어 있으며, 클래스 1의 비율은 10% 정도이며 샘플수는 5,471이다. German Credit 데이터는 관측자의 신용정도를 나타낸 자료로서 설명변수의 개수가 590개이다. 반응변수는 Good과 Bad로 신용상태를 나타내고 있으며 설명변수에는 대출정도, 대출목적과 관측자의 신상정보가 포함되어 있다. 클래스 1의 비율은 30% 정도이며, 총 샘플수는 1,000이다. Secom 데이터는 반도체 공정에 대한 자료로 53개의 설명변수가 주어져 있다. 반응변수는 신호 처리에 따라 house line testing의 성공, 실패 여부를 나타내며 설명변수는 신호 처리와 관련된 정보를 나타내고 있다. 클래스 1의 비율은 6% 정도이고 샘플수는 1,567개이다.

데이터는 먼저 학습 데이터와 평가 데이터를 7 : 3의 비율로 분할하였다. 학습 데이터에서만 샘플링 기법 및 10-폴드 교차검증(cross validation; CV)을 진행하였고 학습이 완료된 모형은 평가 데이터를 활용해 분류의 예측력이 평가되었다. 분류 모형의 예측력을 평가하는 방법은 receiver operating characteristic (ROC) 곡선의 밑의 면적인 area under the curve (AUC)를 이용하였다. AUC값이 1에 가까울수록 모형의 예측력이 우수한 것으로 판단 할 수 있다.

모형적합에서 오버샘플링, 언더샘플링, SMOTE 모두 학습데이터에서의 1과 0의 비율이 1 : 1이 되도록 설정하였다. 로지스틱 회귀모형은 데이터의 모든 설명변수를 주효과로 사용하여 적합을 하였고 Lasso는 CV를 사용하여 조절모수(tuning paremeter)를 선택하여 변수선택을 하였다. 랜덤 포레스트 모형에서는 트리 수는 500으로 고정하고 데이터별로 붓스트랩 데이터에서 모형을 구축할 때 사용되는 변수의 수를 CV를 통해 설정하였다. 부스팅에서도 트리수는 500으로 고정하였고 축소 모수(shrinkage parameter)와 트리의 복잡도를 나타내는 모수는 CV를 통하여 선택하였다. SVM에서는 커널은 선형(linear)커널을 사용하였고 비용(cost) 조절모수는 CV를 통하여 결정하였다. 모형 적합에는 R의 Caret (Kuhn, 2016) 패키지와 gbm (Ridgeway, 2017), randomForest (Liaw와 Wiener, 2002), e1071 (Meyer 등, 2017), glmnet (Friedman 등, 2010) 패키지를 활용하였다.

3.2. 오버샘플링과 SMOTE에서의 교차 검증

먼저 학습 데이터에 오버샘플링을 적용한 후 교차 검증을 통해 AUC가 최대가 되는 모형을 얻는 절차를 CV1이라고 하자. CV1으로부터 교차검증에서 얻어진 AUC와 평가 데이터로 평가하였을 때 얻어진 AUC를 Table 3.1에 보고하였다. Autopart와 Page-Black의 경우에는 사용되는 모형에 관계없이 두 AUC 사이의 차이가 크지 않은 편이다. 그러나 German Credit과 Secom의 경우에는 상황이 다르다. 교차 검증을 통해 얻어진 AUC와 평가 데이터에서 얻어진 AUC 사이의 차이는 상당하며, 모든 경우 교차 검증을 통해 얻어진 AUC가 더 큰 값을 가지고 있는 것으로 나타났다. 이는 교차 검증에서 얻어진

Table 3.1. Comparison of AUC under over-sampling

Data	Model	CV AUC (CV1)	Test AUC (CV1)	CV AUC (CV2)	Test AUC (CV2)
Autopart	Logistic	0.9560	0.9631	0.9572	0.9571
	Lasso	0.9558	0.9628	0.9912	0.9919
	Random Forest	0.9999	0.9945	0.9920	0.9917
	Boosting	0.9760	0.9163	0.9895	0.9895
	SVM	0.9809	0.9786	0.9327	0.9283
Page-Black	Logistic	0.9652	0.9570	0.9613	0.9630
	Lasso	0.9652	0.9570	0.9592	0.9747
	Random Forest	0.9997	0.9894	0.9925	0.9861
	Boosting	0.9966	0.9906	0.9925	0.9772
	SVM	0.9869	0.9848	0.9142	0.9051
German Credit	Logistic	0.8580	0.7432	0.7658	0.7234
	Lasso	0.8551	0.7546	0.7836	0.7377
	Random Forest	0.9990	0.7682	0.7809	0.7526
	Boosting	0.8579	0.7598	0.7976	0.7597
	SVM	0.6532	0.5941	0.6927	0.6468
Secom	Logistic	0.9999	0.6768	0.5925	0.6953
	Lasso	0.9945	0.6594	0.6884	0.6917
	Random Forest	0.9999	0.8003	0.7490	0.7653
	Boosting	0.9757	0.7452	0.6862	0.8062
	SVM	0.8728	0.5612	0.6045	0.5614

AUC = area under the curve; CV = cross validation; SVM = support vector machine.

Table 3.2. Comparison of AUC under synthetic minority over-sampling technique

Data	Model	CV AUC (CV1)	Test AUC (CV1)	CV AUC (CV2)	Test AUC (CV2)
German Credit	Logistic	0.8270	0.7363	0.7754	0.7405
	Lasso	0.8251	0.7384	0.7799	0.7421
	Random Forest	0.9832	0.7281	0.7839	0.7526
	Boosting	0.8916	0.7622	0.7914	0.7537
	SVM	0.6681	0.5797	0.6897	0.6302
Secom	Logistic	0.6329	0.5594	0.6066	0.6953
	Lasso	0.9924	0.7179	0.6662	0.6797
	Random Forest	0.9988	0.7302	0.7666	0.7653
	Boosting	0.9733	0.7611	0.7389	0.8063
	SVM	0.7273	0.5162	0.5834	0.5614

AUC = area under the curve; CV = cross validation; SVM = support vector machine.

AUC 값에 과적합 현상이 나타난 것으로 이해할 수 있다.

이러한 현상은 오버 샘플링에서 뿐만 아니라 SMOTE에서도 나타났다. Table 3.2에서 확인되는 것처럼, German Credit과 Secom 데이터에서 SMOTE는 교차 검증을 통한 AUC 값이 평가 데이터에서 얻어지는 AUC 값에 비해 상당히 높에 나오는 것으로 확인되었다. 여기서, Autopart와 Page-Black 데이터에서는 R의 SMOTE함수가 작용하지 않아 비교에서 제외 시켰다.

Altini (2015)가 지적하였듯이 Ren 등 (2015)와 같은 문헌들에서는 오버 샘플링과 SMOTE 방법을 적용할 때 CV1과 유사한 방법을 통해 분석한 후 결과의 개선이 있음을 보고하였다. 이 부분은 그러나 오버샘플링과 SMOTE의 방법은 클래스가 1인 데이터를 복제하여 사용하기 때문에 교차 검증을 하

기 전에 오버샘플링이나 SMOTE를 적용하면 복제된 데이터가 검증(validation) 데이터에 들어가기 때문에 과적합이 발생하게 됨을 알 수 있다. 따라서 올바른 교차 검증이 되기 위해서는 오버 샘플링과 SMOTE는 교차 검증 내부에서 적용되어야 한다. 예를들어 5-폴드 교차검증을 진행한다고 가정하여 보자. 학습데이터 셋은 20%씩 5개의 조각으로 나누어지게 된다. 첫 단계에서 첫 번째 20% 조각을 검증 데이터로 사용하고, 나머지 80%를 모형을 적합하는데 사용한다고 하자. 그러면 오버 샘플링과 SMOTE는 이 80%에만 적용되어야 한다는 것이다. 그리고 두 번째 단계에서 두 번째 20% 조각을 제외한 나머지 80% 자료에 다시 오버 샘플링과 SMOTE가 적용되어야 한다. 이와 같은 방식이 5회 반복되어야 한다. 이러한 방식은 몇몇 문헌에서 언급되어 있는데, 예를 들어 Xie와 Qiu (2007)는 선형 판별분석문제에서 반응변수의 불균형이 심할 때 교차검증이 검증데이터와 분리된 학습 데이터에 적용되어야 함을 간략히 언급하였다. 본 논문에서는 이처럼 교차검증 안에서 오버 샘플링을 사용한 방법을 CV2라 하였고 결과는 Table 3.1에 보고되어있다. CV1과는 달리 교차 검증에서 얻어진 AUC와 평가 데이터에서 얻어진 AUC 사이에 이전과 같은 체계적인 차이는 거의 나타나지 않았다. SMOTE의 경우도 마찬가지로 CV2 방법이 사용될 때 교차 검증을 통해 보고된 AUC값은 평가데이터에서 얻어진 AUC와 상당히 유사하였다.

따라서 오버 샘플링과 SMOTE 방법에서 교차검증을 사용할 때 얻어진 예측력이 평가 데이터에서의 예측력을 대표하기 위해서는 CV2를 사용해야 함을 알 수 있다.

3.3. 언더샘플링에서의 주의점

교차 검증 전에 언더 샘플링을 실행할 때에는 3.2절에서 언급된 문제점이 발생하지는 않는다. 왜냐하면 언더 샘플링은 자료를 축소시키는 방법이기 때문에 오버 샘플링에서 처럼 복제된 자료가 학습 데이터셋과 검증 데이터셋에 나누어져서 들어가는 일이 언더샘플링에서는 발생하지 않기 때문이다. 그러나, 언더 샘플링의 경우 랜덤하게 데이터가 버려질 때 추가적으로 발생할 수 있는 결과의 변동성을 고려해야 한다. Dal Pozzolo 등 (2013)의 경우 언더샘플링을 한 번만 진행하고 결과를 보고하고 있는데, 이는 매우 위험한 방식이다. 언더샘플링의 경우 seed에 따라 사라지는 데이터가 달라지게 되므로 모형의 예측력 결과가 바뀔 수 있다. 만약 모형의 예측력이 seed에 관계없이 거의 일정하다면 문제는 없으나, seed에 따라 모형의 예측력의 변동이 상당하다면 큰 문제가 된다. 이를 확인하기 위해 seed를 300번 바꿔가면서 모형을 적합한 후 평가데이터에서의 AUC를 확인해보았다. Table 3.3은 네가지 데이터에서의 seed를 바꿔가며 구한 언더샘플링의 AUC의 평균과 표준편차이다.

Table 3.3을 보면 Autopart와 Page-Black에서 seed에 따라 AUC결과의 변동이 크지 않은 것을 볼 수 있다. 하지만 German Credit과 Secom에서는 seed에 따라 AUC결과의 변동이 큰 것을 볼 수 있다. 특히 Secom 데이터에서 주목할 만한 큰 변동이 나타났다. 따라서 언더샘플링을 적용할 때 하나의 seed에서 얻어진 결과를 의미 있게 해석하기 위해서는 상당한 주의가 필요하다. 왜냐하면 다른 seed에서는 훨씬 작거나 큰 AUC 결과가 얻어 질 수 있기 때문이다. 따라서 언더샘플링을 통한 분석 결과를 보고할 때에는, 표준편차와 같은 평가 측도의 변동에 대한 정보를 같이 제공해주어야 바람직하다.

3.4. 예측력 평가 결과

4개의 데이터 셋에 대하여 오버샘플링, 언더샘플링, SMOTE를 올바르게 적용하였을 때 로지스틱 회귀모형, Lasso, 랜덤포레스트, 부스팅, SVM을 적합하여 예측력을 평가한 평가데이터의 AUC 결과는 Table 3.4와 같다. 여기서 SMOTE 샘플링의 경우 Autopart와 Page-Black 데이터에서 R의 SMOTE함수가 작용하지 않아 비교에서 제외 시켰다.

Table 3.3. Variability of AUC under under-sampling (different seed numbers)

Data	Model	Mean AUC	SD of AUC
Autopart	Logistic	0.9585	0.0001
	Lasso	0.9582	0.0007
	Random Forest	0.9734	0.0010
	Boosting	0.9897	0.0007
	SVM	0.9617	0.0011
Page-Black	Logistic	0.9623	0.0047
	Lasso	0.9592	0.0043
	Random Forest	0.9890	0.0015
	Boosting	0.9912	0.0012
	SVM	0.9746	0.0019
German Credit	Logistic	0.7638	0.0121
	Lasso	0.7698	0.0099
	Random Forest	0.7656	0.0113
	Boosting	0.7645	0.0095
	SVM	0.5930	0.0187
Secom	Logistic	0.5412	0.0374
	Lasso	0.6738	0.0459
	Random Forest	0.6983	0.0381
	Boosting	0.7431	0.0301
	SVM	0.5183	0.0324

AUC = area under the curve; SVM = support vector machine.

Table 3.4. Comparison of AUC for test datasets

Data	Model	No sampling	Oversampling	Undersampling	SMOTE
Autopart	Logistic	0.9458	0.9571	0.9596	Not applicable
	Lasso	0.9424	0.9611	0.9544	
	Random Forest	0.9717	0.9919	0.9717	
	Boosting	0.9939	0.9895	0.9927	
	SVM	0.9649	0.9283	0.9728	
Page-Black	Logistic	0.9450	0.9630	0.9669	Not applicable
	Lasso	0.9494	0.9747	0.9650	
	Random Forest	0.9902	0.9861	0.9904	
	Boosting	0.9876	0.9772	0.9899	
	SVM	0.9798	0.9051	0.9781	
German Credit	Logistic	0.8157	0.7234	0.7532	0.7405
	Lasso	0.8573	0.7377	0.8499	0.7421
	Random Forest	0.7707	0.7526	0.7852	0.7526
	Boosting	0.7607	0.7597	0.7882	0.7537
	SVM	0.6371	0.6468	0.5352	0.6302
Secom	Logistic	0.6822	0.6953	0.5234	0.5090
	Lasso	0.7191	0.6917	0.7203	0.6797
	Random Forest	0.7366	0.7653	0.6585	0.7653
	Boosting	0.7754	0.8062	0.7310	0.8020
	SVM	0.5410	0.5614	0.5127	0.5977

AUC = area under the curve; SMOTE = synthetic minority over-sampling technique; SVM = support vector machine.

Autopart에서는 샘플링을 사용하지 않은 부스팅이 가장 높은 AUC를 보이는 것을 볼 수 있고, Page-Black에서는 언더샘플링을 적용한 랜덤포레스트 모형이 가장 높은 AUC를 보이는 것을 볼 수 있다. 또한 German Credit에서는 샘플링을 사용하지 않은 Lasso 모형이 제일 높은 AUC를 보였으며 Secom에서는 오버샘플링을 적용한 부스팅 모형이 가장 높은 AUC를 보이는 것을 볼 수 있다. 따라서 사실상 불균형한 이항자료를 다룰 때 어떤 특정한 분석기법이 우선적으로 고려될만한 근거는 없어 보인다고 할 수 있다. 특정 샘플링 기법을 추천하는 것을 어려우므로, 문제마다 각 샘플링 기법을 모두 적용하여 교차검증에서 가장 높은 AUC를 제공하는 방법을 사용하는 것이 현실적인 접근법이 된다. Hulse 등 (2007)의 실제 자료분석 및 대규모 수치 실험 결과에서도 불균형한 이항 자료의 분석에서 전체적으로 우수한 성능을 보이는 특정한 샘플링 기법은 없었다. 샘플링 기법의 효과에 영향을 주는 요인은 굉장히 다양한 것으로 파악되고 있으며, 불균형한 이항 자료에 고도화된 샘플링 기법이 오히려 단순한 샘플링 기법에 비해 성능이 훨씬 떨어지는 것으로 파악되고 있다 (Hulse 등, 2007). 따라서, 특정한 분석 기법을 우선시 하는 것보다 본 논문에서 고려된 것처럼 몇 가지 기계학습 기법과 샘플링 기법의 다양한 조합을 고려하는 것이 필요해 보인다.

4. 결론

본 연구에서는 반응변수가 이항 자료이고 두 클래스의 비율이 불균형할 때의 분류 기법과 샘플링 방법의 모형 성능을 비교해 보았다. 오버샘플링과 SMOTE의 경우 교차 검증에서 과적합을 방지하기 위하여 주의할 점을 확인 하였고, seed에 따른 언더샘플링의 AUC의 변동성을 보고해야함을 강조하였다. 실제 데이터 분석 결과 특정 샘플링 기법을 추천하기는 어려워 보이며, 데이터에 따라 샘플링 기법의 사용이 예측 성능을 개선하지 못하는 경우도 확인 할 수 있었다. 따라서 실제에서는 여러 샘플링 기법을 적용해보고 성능의 우수성이 두드러지는 방법을 데이터에 맞게 사용하는 것이 필요하다.

부록: 데이터 출처

본 논문에서 사용한 데이터의 출처는 아래와 같다.

Autopart 데이터: https://kbig.kr/edu_manual/html/car_update/basic/car_chapter_1.html

Page-Black 데이터: https://archive.ics.uci.edu/ml/datasets/Page+Blocks+Classification

German Credit 데이터: https://artax.karlin.mff.cuni.cz/r-help/library/caret/html/GermanCredit.html

Secom 데이터: https://archive.ics.uci.edu/ml/datasets/SECOM

References

Altini, M. (2015). Dealing with imbalanced data: undersampling, oversampling and proper cross-validation. http://www.marcoaltini.com/blog/dealing-with-imbalanced-data-undersampling-oversampling-and-proper-cross-validation.

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.

Dal Pozzolo, A., Caelen, O., Waterschoot, S., and Bontempi, G. (2013). Racing for unbalanced methods selection. In *International Conference on Intelligent Data Engineering and Automated Learning*, (pp. 24–31), Springer, Berlin, Heidelberg.

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent, *Journal of Statistical Software*, **33**, 1–22.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., and Herrera, F. (2012). A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches, *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, **42**, 463–484.

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

He, H. and Ma, Y (2013). *Imbalanced Learning: Foundations, Algorithms, and Applications*, Wiley-IEEE Press, New Jersey.

Hulse, J. V., Khoshgoftaar, T. M., and Napolitano, A. (2007). Experimental perspectives on learning from imbalanced data. In *Proceedings of the 24th International Conference on Machine Learning*, 935–942.

Kuhn, M. (2016). Building predictive models in R using the caret package, *Journal of Statistical Software*, **28**(5).

Liaw, A. and Wiener, M. (2002). Classification and regression by randomForest, *R News*, **2**, 18–22.

Longadge, R. and Dongre, S. (2013). Class imbalance problem in data mining review, *arXiv preprint arXiv:1305.1707*

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., and Leisch, F. (2017). e1071: Misc Functions of the Department of Statistics, *R package version 1.6-8*.

Ren, P., Yao, S., Li, J., Valdes-Sosa, P. A., and Kendrick, K. M. (2015). Improved prediction of preterm delivery using empirical mode decomposition analysis of uterine electromyography signals, *PLOS ONE*, **10**, e0132116

Ridgeway, G. (2017). gbm: generalized boosted regression models, *R package version 2.1.3.*

Xie, J. and Qiu, Z. (2007). The effect of imbalanced data sets on LDA: a theoretical and empirical analysis, *Pattern Recognition*, **40**, 557–562.

불균형적인 이항 자료 분석을 위한 샘플링 알고리즘들: 성능비교 및 주의점

김한용[a] · 이우주[a,1]

[a]인하대학교 통계학과

(2017년 7월 17일 접수, 2017년 9월 2일 수정, 2017년 9월 12일 채택)

요 약

파산감지, 스팸메일 감지, 불량품 감지 등 일상생활에서 불균형적인 이항 분류 문제를 다양하게 접할 수 있다. 반응변수의 클래스의 비율이 상당히 불균형한 경우 이항 분류 모형의 예측 성능이 좋지 않다는 점은 이미 잘 알려진 사실이다. 이러한 문제점을 해결하기 위해 그 동안 오버 샘플링, 언더 샘플링, SMOTE와 같은 여러 샘플링 기법이 개발되어 왔다. 본 연구에서는 분류 모형으로 많이 사용되는 기계학습모형으로 로지스틱 회귀모형, Lasso, 랜덤포레스트, 부스팅, 서포트 벡터 머신을 위의 샘플링 기법들과 결합하여 사용했을 때의 예측 성능을 살펴보았다. 실질적인 예측 성능의 개선 여부를 확인하기 위해 네 개의 실제 자료를 분석하였다. 이와 더불어, 샘플링 방법이 사용될 때 주의해야 할 점에 대해서 강조하였다.

주요용어: 불균형적인 이항 자료, 샘플링, 분류, 예측

본 연구는 정부(행정안전부)의 재원으로 재난안전기술개발사업단의 지원을 받아 수행된 연구임 (MOIS-재난-2015-05).

[1]교신저자: (22212) 인천광역시 남구 인하로100, 인하대학교 통계학과. E-mail: lwj221@gmail.com