The Korean Journal of Applied Statistics
2024, Vol. 37, No. 5, 615–629

DOI: https://doi.org/10.5351/KJAS.2024.37.5.615
Print ISSN1225-066X / Online ISSN 2383-5818

GIR-based canonical forest: An ensemble method for imbalanced big data

Solji Han[a], Jaesung Myung[b], Hyunjoong Kim[1,a]

[a]Department of Statistics and Data Science, Yonsei University; [b]AI advanced technology, SK hynix

Abstract

In the field of big data mining, the challenge of imbalanced classification problem has been actively researched for decades. While imbalanced data issues manifest in various forms, past research mainly focused on addressing sample size imbalance between classes. However, recent studies have revealed that rather than the imbalance in sample size alone, the degradation of classification performance significantly worsens when the class overlap is combined. In response, this study introduces GC-Forest (GIR-based canonical forest), an effective ensemble classification method that utilizes weighted resampling technique considering the degrees of overlap between classes. This method measures the imbalance ratio in terms of class overlap at each stage of ensemble and balances the classes by increasing the representativeness of the minority class. Additionally, to improve overall classification performance, the GC-Forest method adopts the canonical forest method as an ensemble classifier, which is designed to enhance both the performance and diversity of individual classifiers. The performance of the proposed method was compared and verified through experiments using 14 different types of real imbalanced data. GC-Forest showed very competitive classification performance in terms of AUC, PR-AUC, G-mean, and F1-score compared to 7 other ensemble methods.

Keywords: classification, imbalanced data, ensemble, imbalance ratio, canonical forest

1. 서론

빅데이터 마이닝 분야의 주된 문제는 분류(classification)에 관한 것이다. 그중, 클래스 불균형 문제는 클래스의 분포가 상당히 치우쳐 있는 경우로, 분류 모형 적합 시 모형이 다수 클래스에 편향되어 학습되기 때문에 분류 성능 저하를 이루는 주된 원인이 된다 (Anand 등, 1993; Fernández 등, 2018; López 등, 2013; SUN 등, 2009). 이러한 불균형 데이터 문제는 학술적인 가치 외에도 의학, 금융, 보안 등 다양한 실생활의 영역에서 빈번한 도전 과제가 된다. 분류 모형의 학습에서 불균형 데이터를 다루는 일반적인 접근 방식은 크게 데이터 수준 방법과 알고리즘 수준 방법으로 구분되며, 이 두 방법을 함께 적용하는 하이브리드 접근법도 광범위하게 연구되어 왔다. 데이터 수준 접근법은 주로 과소 표집(under sampling)과 과대 표집(over sampling) 같은 재표집 방법을 활용하여, 각 클래스 간의 데이터 수를 균형 있게 맞추는 것을 목표로 한다 (Chawla 등, 2002; He 등, 2008). 반면, 알고리즘 수준 접근법은 다수 클래스의 편향(bias)을 완화하는 데 중점을 두며 기존 모형을

Hyunjoong Kim's work was supported by the MSIT (Ministry of Science and ICT), Korea, under the ICAN (ICT Challenge and Advanced Network of HRD) support program (IITP-2023-00259934) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation) and by the National Research Foundation of Korea (NRF) grant funded by the Korean government (No. 2016R1D1A1B02011696).

[1]Corresponding author: Department of Statistics and Data Science, Yonsei University, Seoul 03722, Korea. E-mail: hkim@yonsei.ac.kr

Published 31 October 2024 / journal homepage: http://kjas.or.kr

수정하는 방법을 포함한다 (Buda 등, 2018; Cheng 등, 2017; Fan 등, 1999). 하이브리드 접근법은 데이터 수준 접근법과 알고리즘 수준 접근법을 결합하여 두 접근법의 장점을 모두 활용하고자 한다 (Gong 등, 2017; Rayhan 등, 2017; Seiffert 등, 2010).

전통적으로 클래스 간 불균형 정도를 평가하는 척도는 클래스 간의 표본 크기 비율을 의미하는 불균형 비율(imbalance ratio; IR)로 정의된다. 대다수의 데이터 수준 접근법은 이러한 불균형 비율(IR)을 기반으로 클래스간 표본 크기의 균형을 맞춘 뒤 분류기(classifier)를 학습하고자 한다. 그러나, 최근 몇 년간의 연구는 클래스 간 표본 크기 불균형보다 클래스 중첩(overlap)의 문제가 분류 성능 저하에 더 두드러진 영향을 미치는 것을 확인하였다 (Japkowicz 등, 2003; Jo 등, 2004; Tang과 He, 2017). 또한, 표본 크기의 불균형과 중첩 문제가 결합된 경우, 분류 성능이 훨씬 더 악화되는 것을 확인하였다 (Garcia 등, 2007; Vuttipittayamongkol 등, 2021). 전통적으로 표본 크기를 기반으로 클래스 간 불균형의 정도를 측정하는 불균형 비율(IR)은 실제 클래스 분류의 난이도를 반영하지 못한다. Tang과 He (2017)는 클래스간 중첩을 반영하는 새로운 클래스 불균형 척도로서 클래스 간 내적 일관성(intra-class coherence)을 기반으로 한 일반화된 불균형 비율(generalized IR; GIR)을 제안하고, 그 이론적 특성을 증명하였다. 일반화된 불균형 비율(GIR)은 클래스 간 중첩의 정도를 바탕으로 분류 성능에 영향을 미치는 학습 난이도를 반영한 불균형 척도이다.

본 연구는 불균형 분류 문제에 대처하기 위해 GC-Forest라는 새로운 하이브리드 앙상블 알고리즘을 제안한다. 이 알고리즘은 일반화된 불균형 비율(GIR)을 기반으로 한 과소 표집(under sampling) 기법과 Chen 등 (2013)이 제안한 canonical forest를 결합하여 불균형 데이터의 분류 성능을 향상시키고자 한다. GC-Forest에서 불균형 데이터는 GIR을 기준으로 여러 개의 균형 잡힌 붓스트랩(bootstrap) 표본으로 분할되며, 개별 분류기는 학습이 어려운 예제들에 집중하여 학습을 진행하도록 고안되었다. 다만, 이러한 재 표집 과정에서 비교적 학습이 쉬운 다수 클래스의 예제들은 과소 표집을 통해 제외되기 때문에, 전체 알고리즘의 분류 성능을 보장하기 위해서는 강력한 성능을 갖춘 분류 방법을 사용하는 것이 매우 중요하다. 이러한 맥락에서 GC-Forest는 정준 선형 판별 분석(canonical linear discriminant analysis; CLDA)과 배깅(bagging)을 결합한 canonical forest를 앙상블 방법으로 적용한다. Canonical forest는 선형 변환된 부분 공간에서 분류기를 적합시키기 때문에, 분류기의 다양성을 제공할 뿐만 아니라 어려운 분류 환경에서도 우수한 클래스 분류 성능을 보여주는 것으로 알려져 있다 (Chen 등, 2013). 실제 이진 불균형 데이터를 이용한 실험을 통해 본 연구가 제안한 GC-Forest는 AUC와 PR-AUC 측면에서 대표적인 앙상블 방법 및 불균형 분류를 위해 널리 사용되는 하이브리드 접근법들보다 우수한 성능을 보이는 것을 확인하였다. 이후에 이어지는 본 연구의 구성은 다음과 같다. 2장에서는 제안된 알고리즘의 기초가 되는 일반화된 불균형 비율(GIR)과 앙상블 방법인 canonical forest에 대해 검토하고, 3장에서는 본 연구가 제안하는 GC-Forest에 대해 소개하며, 4장에서는 실제 불균형 데이터를 통한 실험 결과를 제시한다.

2. 이론적 배경

본 절에서는 GC-Forest의 이론적 배경이 되는 일반화된 불균형 비율(GIR)과 이를 이용한 재 표집 방법, 그리고 앙상블 방법인 canonical forest에 대해 소개하고자 한다.

2.1. Generalized imbalanced ratio (GIR)

이진 분류 문제에서 클래스 불균형의 심각성을 측정하는 전통적인 기준은 불균형 비율(IR)로, 이는 다수 클래스과 소수 클래스 간 표본 크기의 비율로 정의된다. 그러나, 실제로 이진 분류의 문제에서 분류 성능 저하에 영향을 미치는 것은 클래스간 표본 크기의 불균형 그 자체보다는 클래스의 중첩 문제이다. 불균형 비율(IR)은 클래스 중첩을 고려하지 않는 척도이기 때문에, 이를 기반으로 한 데이터 수준 접근 방법들 역시 분류기의 성능을 보장하지 못할 수 있다. 이러한 관점에서, Tang과 He (2017)는 클래스 중첩의 정도를 고려한 일반화된

불균형 비율(GIR)을 새로운 불균형 측정 척도로 제안하였다. GIR은 개별 클래스의 내부 일관성(intra-class coherence)를 기반으로 계산되며, 소수 클래스의 내부 일관성은 아래와 같이 정의된다.

$$T_+ = \frac{1}{N_+}\sum_{\mathbf{x}\in\mathcal{P}}\frac{1}{k}\sum_{r=1}^{k} I_r(\mathbf{x},\mathcal{X}) = \frac{1}{N_+}\sum_{\mathbf{x}\in\mathcal{P}} t_k(\mathbf{x}). \tag{2.1}$$

식 (2.1)에서, k는 고려되는 최근접 이웃(nearest neighbors)의 개수이며, $\mathcal{P}$는 N_+의 표본 크기를 갖는 소수 클래스 집합이고, $I_r(\mathbf{x},\mathcal{X})$는 개별 표본 $\mathbf{x}$과 그의 r번째 최근접 이웃이 같은 클래스 라벨에 속하는지를 나타내는 지시 함수이다. 주어진 클래스 내 특정 예제 $\mathbf{x}$에 대해, $t_k(\mathbf{x})$는 $\mathbf{x}$의 k-최근접 이웃 중 몇 개의 예제가 같은 클래스 내에 분포하는지를 계산하며, 이를 개별 클래스 내 모든 예제에 대해 요약한 것이 클래스 내부의 일관성이다. $t_k(\mathbf{x})$는 각 예제의 학습 난이도를 반영하며, 이를 통해 계산된 클래스 내부의 일관성은 클래스 내 각 예제의 근접 이웃들이 다른 클래스와 겹치지 않고 동일한 클래스에 얼마나 집중되어 있는지를 나타낸다.

불균형 문제에서 소수 클래스의 예제들은 종종 다수 클래스 내에 중첩되어 분포한다. 이로 인해 소수 클래스의 클래스 내부 일관성은 다수 클래스의 경우보다 현저히 낮아진다. GIR은 이러한 현상을 반영하여 클래스 간 클래스 내부 일관성의 차이를 통해 불균형을 측정하고자 하였으며, 다음과 같이 계산된다.

$$\mathrm{GIR} = \Delta T = T_- - T_+, \quad 0 \le |\Delta T| \le 1. \tag{2.2}$$

식 (2.2)에서, T_-와 T_+는 각각 다수 및 소수 클래스의 내부 일관성을 의미한다. GIR은 이론적으로는 −1 과 +1 사이 값을 가질 수 있으나 보통 다수 클래스의 내부 일관성이 높아서 0과 1사이의 값을 가지며, 표본 크기의 차이와 관계없이 클래스 간 중첩과 그로 인해 발생하는 학습 난이도를 반영하여 클래스 간 불균형을 평가할 수 있다. 큰 값의 GIR은 소수 클래스 예제가 다수 클래스 내에 극도로 겹쳐져 있고, 같은 소수 클래스 라벨을 가진 데이터가 멀리 흩어져 분포되어 있어 분류기의 학습이 어려운 상황을 나타낸다. 또한, Tang과 He (2017)는 클래스 간 표본 크기가 동일할 때 GIR의 평균이 근사적으로 0이 되어 전통적인 IR과 유사한 특성을 갖는 것을 확인하였으며, 이를 기반으로 GIR을 활용한 적응적 재 표집과 부스팅(boosting) 기법을 결합하는 앙상블 방법을 제안하였다.

2.2. Canonical forest

앙상블 방법은 분류 문제에서 뛰어난 성능을 발휘해 왔으며, 지난 수십 년간 부스팅(boosting)과 배깅(bagging) 개념을 기반으로 한 다양한 앙상블 기법들이 개발되었다. Canonical forest (Chen 등, 2013)은 정준 선형 판별 분석(CLDA)과 배깅을 결합한 기법이다.

Canonical forest는 정준 선형 판별 분석(CLDA)을 적용하여 개별 붓스트랩 표본에서 클래스 간의 분리를 최대화하는 정준 좌표(canonical coordinate)를 얻고, 이 정준 좌표를 통해 선형 변환된 부분 공간에서 분류기를 구축한다. 이로 인해 개별 분류기의 다양성을 보장하며, 클래스 간의 분산을 최대화하면서, 중첩된 데이터가 존재하는 경우에도 우수한 분류 성능을 가진 개별 분류기를 확보할 수 있다. 또한, canonical forest는 전체 데이터가 아닌 부분 집합으로 쪼개진 붓스트랩 데이터에 대해 CLDA를 적용함으로써 앙상블 내 분류기의 다양성을 확보한다. 개별 분류기의 높은 성능과 다양성을 바탕으로 canonical forest는 전통적인 부스팅 및 배깅 앙상블 방법들에 비해 우수한 분류 성능을 보여주었다. Algorithm 1은 canonical forest의 각 단계들을 요약한 것이다.

3. GC-Forest

본 절에서는 불균형 이진 분류 데이터를 위한 새로운 알고리즘인 GC-Forest (GIR based canonical forest)에 대해 소개한다.

Algorithm 1 : Canonical forest

Input

X: n개의 예제로 구성된 훈련 데이터 ($n \times p$ 행렬).

Y: 훈련 데이터의 클래스 레이블(label) ($n \times p$ 행렬).

B: 앙상블 내 분류기의 개수

F: 훈련 데이터 (X, Y) 내의 전체 특성 집합 (feature set)

K: 겹치치 않는 (disjoint) 특성 부분집합의 수

w: 클래스 레이블의 집합 $(1, 2, \ldots, C)$

procedure CanonicalForest(B, K)

for i in $(1, 2, \ldots, B)$ **do**

1. F를 무작위로 K개의 겹치지 않는 특성 부분 집합 $F_{i,j}$로 나눈다.
만일, p가 K로 나누어 떨어지면, $F_{i,j}$는 $m = p/K$개의 특성을 포함하고, 그렇지 않은 경우 각 $F_{i,j}(j < K)$는 $m = (p/K) + 1$개의 특성을 포함하며 마지막 부분 집합 ($j = K$)은 남은 특성을 포함.

for j in $(1, 2, \ldots, K)$ **do**

2-1. $F_{i,j}$의 특성에 대한 데이터 행렬을 $\underset{(n\times m)}{X_{i,j}}$ 라고 하자.

2-2. 붓스트랩 표본 $X'_{i,j}$을 추출하며, 이때 표본 크기는 $X_{i,j}$ 내 예제 개수의 75%이다.

2-3. $X'_{i,j}$에 CLDA를 적용하여 계수 행렬 $\underset{(m\times m)}{A_{i,j}}$ 를 얻는다.

$$A_{i,j} = \left[A_{i,j}^{(1)}, A_{i,j}^{(2)}, \ldots, A_{i,j}^{(m)}\right]$$

end for

3. $A_{i,j}$ $(j = 1, \ldots, K)$를 블록 대각행렬(block diagonal matrix) $\underset{(m\times m)}{R_i}$ 로 정렬시킨다.

$$R_i = \begin{pmatrix} A_{i,1}^{(1)}, A_{i,1}^{(2)}, \ldots, A_{i,1}^{(m)} & 0 & \cdots & 0 \\ 0 & A_{i,2}^{(1)}, A_{i,2}^{(2)}, \ldots, A_{i,2}^{(m)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{i,K}^{(1)}, A_{i,K}^{(2)}, \ldots, A_{i,K}^{(m)} \end{pmatrix}$$

4. 행렬 R_i의 행을 F의 특성 순서에 맞춰 재배열하여 회전 행렬 R_i^a를 구성한다.

5. 선형 변환된 데이터 (XR_i^a, Y)를 새로운 훈련 데이터로 사용하여 분류기 L_i를 구축한다.

end for

end procedure

Output: 주어진 예제 $\mathbf{x}$에 대해, 최종 분류기 L을 통해 예측된 클래스 레이블은 다음과 같다.

$$L(\mathbf{x}) = \arg\max_{y\in w} \sum_{i=1}^{B} I\left(L_i\left(\mathbf{x}R_i^a\right) = y\right)$$

3.1. 표본추출 전략

분류 문제에서 불균형 데이터의 어려움은 소수 클래스 예제가 다수 클래스 예제와 극심히 겹치는 상황에서 발생되며, 두 클래스간 중첩 지역은 군집의 형태로 넓은 영역에 펼쳐져 있는 경우가 대다수이다. 따라서, 중첩 지역이 아닌 곳에 위치한 다수 클래스의 예제를 과소 표집하는 표본 추출 전략은 중첩 영역 내 분류가 어려운

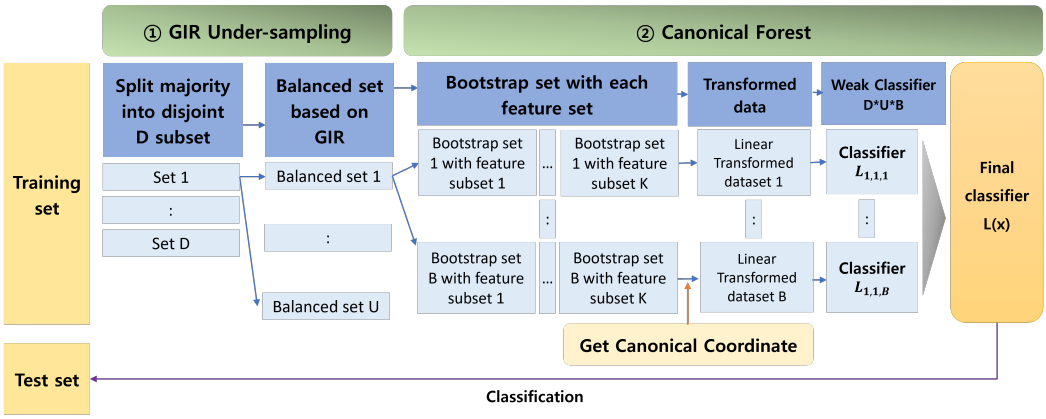

Figure 1: *The structure of GC-Forest.*

데이터에 더 많은 초점을 맞추게 되므로 소수 클래스의 예제를 더 명확하게 감지하는데 도움이 될 수 있다. GIR 기반의 과소 표집은 다수 클래스 예제에 대해 다음과 같이 계산되는 $p_{-}(\mathbf{x})$를 개별 예제에 대한 표본 추출 확률로 사용하여 구성할 수 있다.

$$p_{-}(\mathbf{x}) = \frac{t_k(\mathbf{x}) + 1}{\sum_{\mathbf{x}\in N} t_k(\mathbf{x}) + N_{-}}, \quad \sum_{\mathbf{x}\in N} p_{-}(\mathbf{x}) = 1. \tag{3.1}$$

식 (3.1)에서, $\mathbf{x}$는 다수 클래스 내의 개별 예제이며, $t_k(\mathbf{x})$는 식 (2.2)에서 정의된 값이다. $p_{-}(\mathbf{x})$는 학습이 쉬운 정도를 반영하므로 이를 추출 확률로 이용해서 얻어진 다수 클래스 예제를 붓스트랩 표본에서 제외하면, 학습이 어려운 예제들로 이루어진 균형 잡힌 붓스트랩 표본을 구축할 수 있다. 다수 클래스의 가중 표본 추출은 두 클래스가 GIR 관점에서 균형을 이룰 때까지, 즉 표본 데이터의 GIR이 0 이하의 값을 가질 때까지 반복된다. 다만, 다수 클래스 예제에 대해 k-최근접 이웃을 바탕으로 클래스 내부 일관성을 계산하는 것은 상당히 많은 계산량을 필요로 한다. 따라서, 본 알고리즘은 전체 다수 클래스 예제를 D개의 겹치지 않는 부분집합으로 나누어 각 부분 집합에서 GIR 기반의 과소 표집을 적용하는 것을 표본추출 전략으로 설계하였다. 또한, 분할된 다수 클래스의 개별 부분집합에서 GIR을 기반으로 클래스의 균형을 이룬 데이터 셋을 U번 반복하여 생성하도록 고안하였다. 이를 통해, k-최근접 이웃의 계산 속도를 높일 뿐 아니라 각 붓스트랩 표본 간 다수 클래스 예제의 이질성을 보장하여, 앙상블 알고리즘이 보다 더 다양한 분류기를 학습할 수 있도록 기회를 제공한다.

3.2. GC-Forest 알고리즘

GIR 기반의 과소 표집을 통해 얻어진 붓스트랩 표본은 상대적으로 학습이 어려운 데이터로 구성된다. 이러한 표본에서 높은 수준의 분류 정확도를 달성하기 위해서는 클래스 간 중첩된 영역을 효과적으로 분리하고, 높은 분류 성능을 가진 앙상블 방법을 적용하는 것이 필수적이다. 본 논문은 이러한 조건에 부합하는 canonical forest를 앙상블 방법으로 사용하였다.

요약하자면, GC-Forest는 다수 클래스의 개별 예제의 내부 일관성($p_{-}(\mathbf{x})$)을 재 표집의 가중치로 사용하고, 학습의 난이도를 반영한 불균형 척도인 GIR을 바탕으로 클래스간 균형을 맞추어 각 붓스트랩 표본 내에서 소수 클래스의 대표성을 높힌다. 또한, 앙상블 방법 중 배깅의 체계에서 클래스 간 중첩 환경에서도 뛰어난 분류 성능을 보이는 것으로 알려진 canonical forest를 적용하여 최종 알고리즘의 분류 성능을 향상하고자 하였다. Figure 1는 GC-Forest의 구조를 도식화한 것이며, Algorithm 2는 GC-Forest의 세부 단계를 정리하였다.

Algorithm 2 : GC-Forest

Input

$\mathcal{X}$: $N = N_- + N_+$개의 예제로 구성된 원본 훈련 데이터로 $\mathcal{X} = \mathcal{N} \cup \mathcal{P}$.

$\mathcal{N}$: $\mathcal{X}$ 내 N_-개의 예제로 구성된 다수 클래스(−).

$\mathcal{P}$: $\mathcal{X}$ 내 N_+개의 예제로 구성된 소수 클래스(+).

w: $\mathcal{X}$ 내 클래스 레이블(label) 집합으로, $w = \{-, +\}$.

procedure GC-Forest(D, U, k, B, K)

1. 다수클래스 ($\mathcal{N}$)를 D개의 겹치지 않는 부분 집합 $\mathcal{N}_d (d = 1, \ldots, D)$로 나눈다.

for d in $1, \ldots, D$ **do**

2-1.$\mathcal{N}_d$ 안의 $N_{-,d} = \frac{N_-}{D}$개의 표본과 $\mathcal{P}$ 안의 N_+개의 표본으로 구성된 $\mathcal{X}_d = \mathcal{N}_d \cup \mathcal{P}$을 생성한다.

2-2. $\mathcal{X}_d$의 특성(feature)들을 표준화하고(Z_d), k-최근접 이웃으로 KNN 그래프 $\mathcal{G}_{KNN}(Z_d)$를 구축한다.

3-1. Z_d내의 모든 표본들에 대해 통계량 $t_k(\mathbf{x})$을 계산하고, T_-, T_+를 계산한다.

3-2. 다수 클래스 표본들에 대해 표본 추출 확률 $p_-(\mathbf{x})$를 계산한다.

for u in $1, \ldots, U$ **do**

4-1. GIR = $\Delta T = T_- - T_+$을 계산하고 $\mathcal{N}_{d,u} = \mathcal{N}_d, \mathcal{P}_{d,u} = \mathcal{P}$로 설정한다.

while $\Delta T \leq 0$ **do**

4-2. $p_-(\mathbf{x})$를 추출 분포로 사용하여 다수 클래스 표본 $\mathcal{N}_s$를 추출하고, $\mathcal{N}_s$를 $\mathcal{N}_{d,u}$에서 제외 $\mathcal{N}_{d,u} = \{\mathcal{N} : \mathcal{N} \in \mathcal{N}_{d,u}, \mathcal{N} \neq \mathcal{N}_s\}$.

4-3. $\mathcal{P}$와 $\mathcal{N}_{d,u}$를 사용하여 T_-와 T_+를 업데이트 한다.

end while

for i in $1, \ldots, B$ **do**

5. F를 무작위로 K개의 겹치지 않는 부분집합 $F_{d,u,i,j} (j = 1, \ldots, K)$로 나눈다.

for j in $1, \ldots, K$ **do**

6-1. $\underset{(N_{d,u} \times m)}{X_{d,u,i,j}}$을 $F_{d,u,i,j}$안의 특성들에 대응되는 데이터 행렬이라 하자.

6-2. $X_{d,u,i,j}$로 부터 붓스트랩 표본 $X'_{d,u,i,j}$을 추출한다.
이때, 표본 크기는 $X_{d,u,i,j}$ 예제 개수의 75%로 한다.

6-3. $X'_{d,u,i,j}$에 CLDA를 적용하여 계수 행렬 $\underset{m \times m}{A_{d,u,i,j}}$을 구한다.

end for

7. $A_{d,u,i,j} (j = 1, \ldots, K)$를 블록대각 행렬 $\underset{p \times p}{R_{d,u,i}}$로 정렬한다.

8. $R_{d,u,i}$의 행을 F내 특성들의 원본 순서에 맞춰 재정렬하여 회전 행렬 $R^a_{d,u,i}$을 구성한다.

9. 선형 변환된 데이터 $(X_{d,u} R^a_{d,u,i}, Y_{d,u})$를 새로운 훈련 데이터로 하여 분류기 $L_{d,u,i}$를 생성한다.

end for

end for

end for

end procedure

Output: 주어진 예제 $\mathbf{x}$에 대해 최종 분류기 L을 통해 예측된 클래스 레이블은 다음과 같다.

$$L(\mathbf{x}) = \arg\max_{y \in w} \sum_{d=1}^{D} \sum_{u=1}^{U} \sum_{i=1}^{B} I\left(L_{d,u,i}\left(\mathbf{x} R^a_{d,u,i}\right) = y\right).$$

Table 1: Imbalanced datasets for the experiment

Dataset	Description	Class label (minority/majority)	# Obs	# Features	IR	GIR	Source
info	Innfocamere	Defaulter/non-defaulter firms	11156	27	151.82	0.97	Menardi와 Torelli (2014)
aba1	Abalone19	19/others	4177	7	115.03	0.96	KEEL
eco2	Ecoli4	positive/negative	316	8	15.80	0.89	KEEL
wine	Wine	≤ 4/4 ≥ wine quality	4898	11	25.77	0.80	UCI
gla5	Glass016vs2	2/0, 1, 6	192	9	10.29	0.76	KEEL
gla4	Glass2	2/others	214	9	11.59	0.76	KEEL
aba2	Abalone9vs18	18/9	731	7	16.40	0.72	KEEL
yea7	Yeast1vs7	1/7	459	7	14.30	0.69	KEEL
yea1	Yeast6	Exc/others	1484	8	41.40	0.53	KEEL
yea5	Yeast2vs8	pox/cyt	482	8	23.1	0.44	KEEL
yea2	Yeast5	ME1/others	1484	8	32.73	0.38	KEEL
page	Pageblocks13vs2	graphic/horiz, line, picture	472	10	15.86	0.17	KEEL
vow	0vs14~10	0/others	988	10	9.98	0.02	KEEL
shu2	Shuttlec0vsc4	0/4	1829	9	13.87	0.01	KEEL

4. 실험데이터 분석

본 절에서는 실제 불균형 이진 분류 데이터를 바탕으로 GC-Forest 알고리즘의 성능을 기존의 방법들과 비교하고 결과에 대해 논의한다.

4.1. 데이터 소개

GC-Forest와 다른 분류 앙상블 알고리즘들의 성능을 비교하기 위해 KEEL (Alcalá 등, 2010), UCI machine learning repository (Lichman, 2013), 그리고 Menardi와 Torelli (2014)에서 14개의 실제 불균형 데이터를 수집하였다. Table 1은 실험에 사용된 데이터의 세부 정보를 제공하며, GIR을 기준으로 정렬되어 있다. 실험에 사용된 데이터는 숫자형 변수만 독립변수로 활용되었고, 종속변수는 이진형의 클래스 레이블을 갖는다. Table 1을 통해 데이터의 IR과 GIR이 비례하지 않음을 확인할 수 있다. 실험을 통해 다양한 표본 크기의 불균형 및 학습 난이도 하에서 기존의 알고리즘대비 제안된 GC-Forest의 분류 성능을 검증하고자 한다.

4.2. 실험 디자인

분류 성능 비교를 위해 먼저 원본 데이터를 60 : 40의 비율로 훈련 데이터와 평가 데이터로 나누었고, 이 때 원본 데이터의 클래스 간 표본 크기의 불균형한 분포를 유지하도록 분할하였다. Table 2는 성능 비교에 사용된 알고리즘에 대한 자세한 정보를 제공하며, 해당 8가지 알고리즘은 모두 앙상블 기법으로, 4가지 부스팅 방식과 4가지 배깅 방식을 포함하고 있다. 데이터 재 표집의 방법을 적용하지 않은 대표적인 부스팅 방법(그룹1), 데이터 재 표집의 방법을 적용하지 않은 배깅 방법(그룹2), 불균형 분류를 위해 데이터 재 표집을 결합한 부스팅 방법(그룹3), GIR 기반의 과소 표집하에서 배깅을 결합한 방법(그룹4)을 비교하였다. 해당 알고리즘들은 R (R core Team, 2015)의 패키지를 사용하여 구현되었고 구현을 위해 사용한 R 패키지는 Table 3에 기술되어 있다.

실험을 위한 하이퍼파라미터로 GIR 기반의 과소 표집을 적용한 방법들 (Table 2의 그룹4)에 대해서는, B = 50 (분류기의 개수), D = 5 (분할된 다수 클래스 부분 집합의 수)와 U = 20 (각 다수 클래스 부분 집합별로 생성되는 GIR 균형 데이터셋의 수)으로 하였다. 하이퍼파라미터 D와 U에 대해 그 외 다양한 값을 실험해 보았지만, 결과는 큰 차이가 없었다.

추가적으로 canonical forest에서 입력 변수의 부분 집합의 수를 나타내는 $K = 5$로 설정했고, GC-Fores

Table 2: Classification methods used in the experiment

Groups	Classification methods
Group 1. Boosting family without data resampling	1. Adaboost (AdaB)
	2. Gradient Boosting (GradB)
Group 2. Bagging family without data resampling	3. Random Forests (RF)
	4. Canonical Forest (CF)
Group 3. Boosting family with data resampling	5. RUSBoost (RUSB)
	6. RHSBoost (RHSB)
Group 4. Bagging family combined with GIR-based undersampling	7. GIR Random Forests (GRF)
	8. GIR Canonical Forests (GC-Forest)

Table 3: R packages for implementation

Packages	Algorithms
`rpart` (Therneau와 Atkinson, 2019)	CART algorithm
`adabag` (Alfaro 등, 2013)	Bagging and Adaboost.M1 algorithm
`randomForest` (Liaw 와 Wiener, 2002)	Random Forests algorithm
`ROSE` (Lunardon 등, 2014)	RHSBoost 내 ROSE algorithm
`GBM` (Ridgeway 등, 2024)	Gradient Boosting algorithm

에서 최근접 이웃의 수를 의미하는 $k = 5$로 하였다. 그 외의 방법들 (Table 2의 그룹1, 그룹2, 그룹3) 에서는 분류기의 수를 1,000으로 고정하여 실험을 시행했다.

각 알고리즘의 분류 성능은 AUC, PR-AUC, G-mean 및 F1-score를 기준으로 측정되었다. 불균형 분류 데이터 문제에서 분류 성능을 적절히 표현하는 지표로 널리 알려진 AUC는 재현율과 1-특이도를 표현하는 ROC 곡선 아래의 면적 (Huang 와 Ling, 2005)이며, PR-AUC는 정밀도-재현율(PR) 곡선 아래의 면적 (boyd 등, 2013)으로 AUC 대비 소수 클래스의 성능 개선에 더 민감한 지표로 알려져 있다. 재현율와 특이도의 기하평균으로 측정되는 G-mean (Kubat 등, 1997)과 정밀도와 재현율의 조화평균으로 측정되는 F1-score (Manning, 2008) 역시 클래스 불균형의 경우에 공정한 관점에서 분류 성능을 측정할 수 있다. 안정적 비교를 위해 데이터 마다 성능 평가를 30번씩 반복하였으며, 얻어진 30개의 각 평가지표는 유의 수준 0.05하에서 대응표본 t-검정을 통해 비교되었다.

4.3. 분류 성능 비교

Figure 2와 Figure 3은 GC-Forest와 다른 방법들 간의 반복 비교에서 얻어진 AUC, PR-AUC에 대한 대응 표본 t-검정 통계량의 상자그림(boxplot)결과이다. 해당 그림에서 AUC와 PR-AUC의 t-검정 통계량의 다수가 0 위에 분포되어 있는데, 이는 GC-Forest가 다양한 불균형 상황에서 비교된 방법들 대비 AUC와 PR-AUC의 측면에서 우월한 분류 성능을 가지고 있는 것을 보여준다.

Table 4은 AUC와 PR-AUC를 이용한 대응표본 t-검정의 결과에 따라 승패 횟수를 요약한 win-loss table 이다. Win-loss table은 열(column)에 있는 방법이 행(row)에 있는 방법과의 검정을 통해 양수의 t-검정 통계량을 가지면, 이를 '승리'로 표시하며, t-검정 통계량이 유의하게 크면 '유의미한 승리'로 표시한다. 예를 들어, GC-Forest는 canonical forest 대비 13개의 데이터에서 평균 AUC가 큰 값을 보였으며, 이중 9개 데이터에서 유의수준 0.05하에 통계적으로 유의하게 우월한 성능을 보여 준다. Table 5의 dominance는 t-검정 통계량을 통해 확인된 유의미한 승의 횟수에서 패의 횟수를 차감한 것이며, average rank는 각 데이터별 순위들의 평균값이다. Dominance의 값이 클수록, average rank의 값이 작을수록 더 높은 AUC 값을 보였다고 할 수 있다.

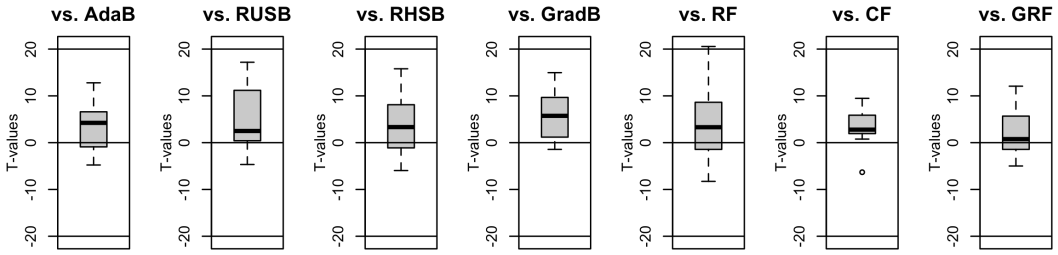

Figure 2: *Paired t-test statistics of AUC.*

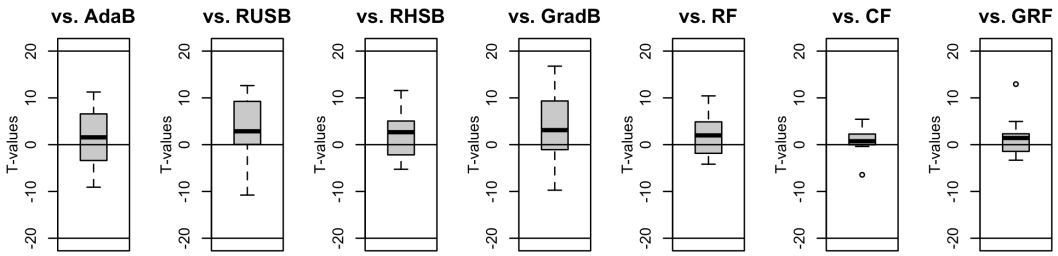

Figure 3: *Paired t-test statistics of PR-AUC.*

Table 4: Win–loss table of AUC

Method	AdaB	RUSB	RHSB	GradB	RF	CF	GRF	GC-Forest
AdaB	0	8 | 4	10 | 6	2 | 1	8 | 5	5 | 4	11 | 6	10 | 9
RUSB	6 | 5	0	11 | 9	7 | 4	9 | 8	7 | 5	12 | 9	11 | 8
RHSB	4 | 2	3 | 3	0	3 | 2	6 | 2	5 | 4	7 | 4	8 | 8
GradB	12 | 9	7 | 6	11 | 9	0	12 | 10	8 | 7	14 | 10	11 | 10
RF	6 | 3	5 | 3	8 | 6	2 | 1	0	5 | 5	10 | 7	9 | 9
CF	9 | 5	7 | 6	9 | 7	6 | 1	9 | 6	0	9 | 7	13 | 9
GRF	3 | 1	2 | 1	7 | 4	0 | 0	4 | 1	5 | 4	0	7 | 5
GC-Forest	4 | 1	3 | 1	6 | 2	3 | 0	5 | 2	1 | 1	7 | 3	0

Table 6와 Table 7은 PR-AUC 값을 이용한 win-loss table과 dominance rank표이며, GC-Forest가 비교된 다른 분류 알고리즘 중 PR-AUC 측면에서도 가장 우월한 성능을 보이는 것을 알 수 있다.

Figure 4는 GC-Forest와 다른 방법들 간의 반복 비교에서 얻어진 G-mean에 대한 대응 표본 t-검정 통계량의 상자그림 결과이다. G-mean에서는 부스팅 방법에 재 표집 방법을 결합한 알고리즘(RHSBoost, RUSBoost)의 성능이 GC-Forest 보다 더 우수하다는 것을 주목할만 하다. 그 외의 경우에 GC-Forest는 좋은 성능을 나타냈다. Figure 5는 F1-score의 상자그림의 결과이다. 여기서 GC-Forest는 F1-score의 측면에서 비교된 모든 방법들과 유사하거나 우월한 성능을 보여준다. 즉, GC-Forest가 클래스 간의 일반적인 판별력 뿐 아니라 소수 클래스에 대한 판별에 있어서도 우월한 성능을 지닌 것을 확인할 수 있다.

Table 5: Dominance rank of AUC

Methods	Dominance	Wins	Losses	Average rank
AdaB	−9	26	35	3.74
RUSB	−24	24	48	4.17
RHSB	18	43	25	3.09
GradB	−52	9	61	4.91
RF	0	34	34	3.54
CF	−11	30	41	3.55
GRF	30	46	16	2.89
GC-Forest	48	58	10	2.11

Table 6: Win–loss table of PR-AUC

Method	AdaB	RUSB	RHSB	GradB	RF	CF	GRF	GC-Forest
AdaB	0	6 | 3	7 | 3	3 | 1	8 | 4	6 | 5	8 | 5	8 | 7
RUSB	8 | 7	0	11 | 7	8 | 5	10 | 8	9 | 7	11 | 7	11 | 8
RHSB	7 | 4	3 | 2	0	5 | 3	9 | 4	7 | 6	9 | 6	8 | 7
GradB	11 | 8	6 | 2	9 | 5	0	12 | 8	11 | 8	10 | 7	10 | 8
RF	6 | 4	4 | 2	5 | 3	2 | 1	0	7 | 5	9 | 4	9 | 7
CF	8 | 4	5 | 3	7 | 3	3 | 1	7 | 2	0	8 | 3	10 | 4
GRF	6 | 4	3 | 0	5 | 4	4 | 1	5 | 3	6 | 4	0	9 | 5
GC-Forest	6 | 4	3 | 1	6 | 4	4 | 2	5 | 2	4 | 1	5 | 2	0

Table 7: Dominance rank of PR-AUC

Methods	Dominance	Wins	Losses	Average rank
AdaB	7	35	28	3.31
RUSB	−36	13	49	4.36
RHSB	−3	29	32	3.64
GradB	−32	14	46	4.23
RF	5	31	26	3.33
CF	16	36	20	3.17
GRF	13	34	21	3.21
GC-Forest	30	46	16	2.76

4.4. GIR 값에 따른 성능 비교

4.3절에서 사용된 불균형 데이터는 Table 1과 같이 다양한 GIR 값을 포함하고 있다. 본 절에서는 GIR 값이 GC-Forest의 성능에 미치는 영향을 조사하였다. Figure 6은 다양한 GIR 값에 대한 각 방법의 GC-Forest 대비 상대적인 AUC 값을 보여주며, 그 외 PR-AUC, G-mean, F1-score 등에 대한 결과도 이와 크게 다르지 않다. 결과적으로, GIR > 0.3일 때 GC-Forest 방법이 다른 방법들에 비해 더 우수한 성능을 보이는 것을 확인하였다. 이는 GIR 값을 고려한 GC-Forest 방법의 타당성을 입증하는 결과라 할 수 있다.

5. 결론

본 논문은 GC-Forest라는 새로운 불균형 데이터 분류 앙상블 알고리즘을 제안하였다. 이 알고리즘은 불균형 데이터의 학습 난이도를 가중치로 하는 과소 표집과 앙상블을 결합한 하이브리드 접근법을 활용하여 해결

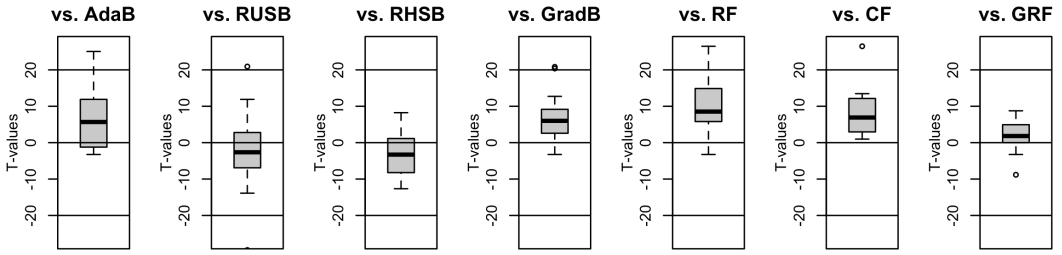

Figure 4: *Paired t-test statistics of G-mean.*

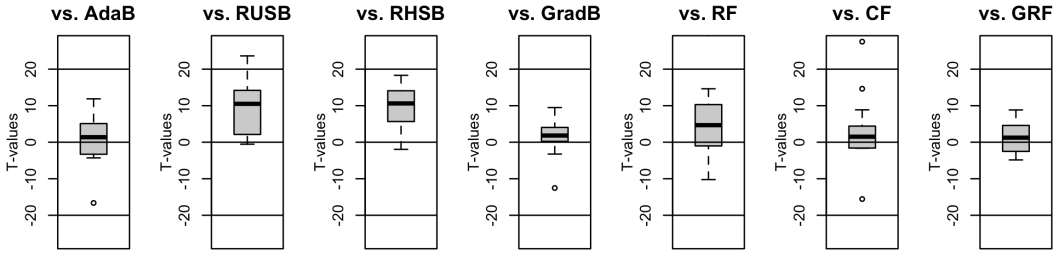

Figure 5: *Paired t-test statistics of F1-score.*

하고자 하였다. 일반화된 불균형 비율(GIR)을 활용한 과소 표집은 클래스 내 일관성을 기반으로 표본 추출 확률을 계산하며, 중첩이 일어나지 않는 학습하기 쉬운 다수 클래스의 표본을 제거하는 방식으로 소수 클래스의 대표성을 높인다. 이후 학습이 어려운 예제에 집중할 수 있도록 고안된 여러 개의 GIR 관점의 균형 잡힌 붓스트랩 데이터 집합을 생성하여, 개별 분류기가 분류 성능에 핵심이 되는 클래스 간 중첩된 영역을 집중적으로 학습할 수 있는 환경을 제공한다. 이러한 상황에서 앙상블 방법으로 적용된 canonical forest는 정준 선형 판별 분석과 배깅의 장점을 활용하여 클래스 분리 가능성과 분류기의 다양성을 최대화함으로써 소수 클래스에 대한 민감한 감지와 정확한 분류를 수행한다. 제안된 GC-Forest의 불균형 데이터에 대한 분류 성능을 입증하기 위해, 다양한 수준의 클래스 불균형을 가진 14개의 실제 데이터를 AUC와 PR-AUC, G-mean과 F1-score 측면에서 비교하였다. 실험 결과에 따르면, GC-Forest는 AUC, PR-AUC, G-mean, F1-score의 모든 평가 척도에서 비교된 다른 방법들보다 평균적으로 유사하거나 우월한 성능을 보여주었다. AUC, PR-AUC, F1-score의 측면에서 GC-Forest는 가장 우수한 결과를 보임으로써, 다양한 불균형 데이터의 상황에서 GC-Forest의 전반적인 분류 우수성이 증명되었다. G-mean의 측면에서는 부스팅 방법에 데이터 재 표집 방법을 활용한 방법들 (그룹2)이 더 안정적인 성능을 보여 주었으나, 데이터 재 표집이 없는 부스팅 방법 (그룹1)이나 배깅 방법 (그룹3) 대비 GC-Forest가 더 우수한 분류성능을 보여주었다.

결과를 종합하여 보면, GC-Forest는 일반화된 불균형 비율(GIR) 기반의 가중치 재 표집 방법 통해 소수 클래스을 더 민감하게 감지하는 이점을 갖고, canonical forest를 통해 개별 분류기의 정확도와 다양성을 높여 소수 클래스과 다수 클래스의 분류 성능 향상에 기여하는 것으로 판단할 수 있다.

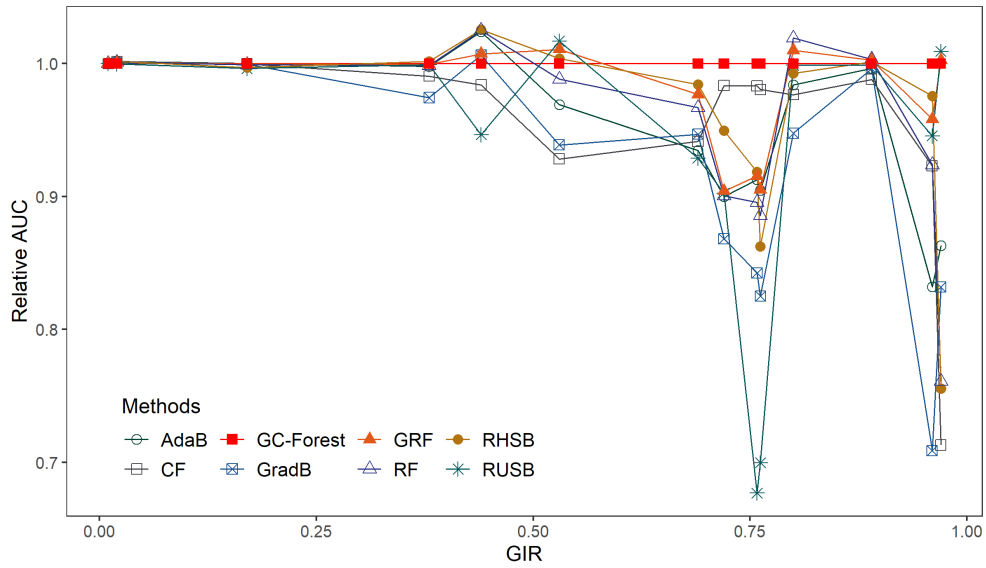

Figure 6: *Relative AUC values of the GC-Forest method for different GIR values.*

References

Alcal-Fdez J, Fernndez A, Luengo J, Derrac J, Garca S, Snchez L, and Herrera F (2011). Keel datamining software tool: Data set repository, integration of algorithms and experimental analysis framework, *Journal of Multiple-Valued Logic and Soft Computing*, **17**, 255–287.

Alfaro E, Gamez M, and Garcia N (2013). Adabag: An r package for classification with boosting and bagging, *Journal of Statistical Software*, **54**, 1–35.

Anand R, Mehrotra K, Mohan C, and Ranka S (1993). An improved algorithm for neural network classification of imbalanced training sets, *IEEE Transactions on Neural Networks*, **4**, 962–969.

Boyd K, Eng KH, and Page CD (2013). Area under the precision-recall curve: Point estimates and confidence intervals. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2013, Prague, Czech Republic, September 23-27, 2013, Proceedings, Part III 13* (pp. 451–466), Springer, Berlin.

Buda M, Maki A, and Mazurowski MA (2018). A systematic study of the class imbalance problem in convolutional neural networks, *Neural Networks*, **106**, 249–259.

Chawla NV, Bowyer KW, Hall LO, and Kegelmeyer WP (2002). Smote: Synthetic minority oversampling technique, *Journal of Artificial Intelligence Research*, **16**, 321–357.

Chen YC, Ha H, Kim H, and Ahn H (2013). Canonical forest, *Computational Statistics*, **29**, 849–867.

Cheng F, Zhang J, Wen C, Liu Z, and Li Z (2017). Large cost-sensitive margin distribution machine for imbalanced data classification, *Neurocomputing*, **224**, 45–57.

Fan W, Stolfo S, Zhang J, and Chan P (1999). Adacost: Misclassification cost-sensitive boosting. In *Proceedings of the Sixteenth International Conference on Machine Learning (ICML'99)*, San Francisco, CA, USA, 97–105.

Fernández A, García S, Herrera F, and Chawla NV (2018). Smote for learning from imbalanced data: Progress

and challenges, marking the 15-year anniversary, *Journal of Artificial Intelligence Research*, **61**, 863–905.

García V, Sánchez J, and Mollineda R (2007). An empirical study of the behavior of classifiers on imbalanced and overlapped data sets. In Rueda L, Mery D, and Kittler J (Eds), *Progress in Pattern Recognition, Image Analysis and Applications* (pp. 397–406), Springer Berlin Heidelberg, Berlin, Heidelberg.

Gong J and Kim H (2017). Rhsboost: Improving classification performance in imbalance data, *Computational Statistics and Data Analysis*, **111**, 1–13.

He H, Bai Y, Garcia EA, and Li S (2008). Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *Proceedings of 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, Hong Kong, 1322–1328.

Huang J and Ling CX (2005). Using auc and accuracy in evaluating learning algorithms, *IEEE Transactions on Knowledge and Data Engineering*, **17**, 299–310.

Japkowicz N (2003). Class imbalances: Are we focusing on the right issue, *Workshop on Learning from Imbalanced Data Sets II*, **1723**, 63.

Jo T and Japkowicz N (2004). Class imbalances versus small disjuncts, *SIGKDD Explorations Newsletter*, **6**, 40–49.

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

Lichman M (2013). UCI machine learning repository, Available from: http://archive.ics.uci.edu/ml

Lunardon N, Menardi G, and Torelli N (2014). Rose: A package for binary imbalanced learning, *R Journal*, **6**, 79–89.

López V, Fernández A, García S, Palade V, and Herrera F (2013). An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics, *Information Sciences*, **250**, 113–141.

Manning CD (2008). *Introduction to Information Retrieval*, Cambridge University Press, Cambridge, England.

Menardi G and Torelli N (2014). Training and assessing classification rules with imbalanced data, *Data Mining and Knowledge Discovery*, **28**, 92–122.

R Core Team (2022). *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria.

Rayhan F, Ahmed S, Mahbub A, Jani R, Shatabda S, and Farid DM (2017). Cusboost: Cluster-based undersampling with boosting for imbalanced classification. In *Proceedings of 2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, Bengaluru, 1–5.

Ridgeway G and GBM Developers (2024). gbm: Generalized Boosted Regression Models. R package version 2.1.9, Available from: ¡https://CRAN.R-project.org/package=gbm¿

Seiffert C, Khoshgoftaar TM, Van Hulse J, and Napolitano A (2010). Rusboost: A hybrid approach to alleviating class imbalance, *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, **40**, 185–197.

Sun Y, Wong AKC, and Kamel MS (2009). Classification of imbalanced data: A review, *International Journal of Pattern Recognition and Artificial Intelligence*, **23**, 687–719.

Tang B and He H (2017). Gir-based ensemble sampling approaches for imbalanced learning, *Pattern Recognition*, **71**, 306–319.

Therneau T, Atkinson B, and Ripley B (2015). rpart: Recursive partitioning and regression trees, r package version 4.1-15, Retrieved, 13:2015, Available from: https://cran.r-project.org/web/packages/rpart/rpart.pdf

Vuttipittayamongkol P, Elyan E, and Petrovski A (2021). On the class overlap problem in imbalanced data clas-

sification, *Knowledge-Based Systems*, **212**, 106631.

Received August 1, 2024; Revised August 22, 2024; Accepted August 25, 2024

불균형 데이터의 분류 성능 향상을 위한 일반화된 불균형 비율(GIR) 기반의 과소 표집 canonical forest (GC-Forest)

한솔지[a], 명재성[b], 김현중[1,a]

[a]연세대학교 통계데이터사이언스학과; [b]SK하이닉스

요 약

빅데이터 마이닝 분야에서 불균형 분류 문제의 도전 과제는 수십 년 동안 활발히 연구되어 왔다. 불균형 데이터 문제는 그 양상과 형태가 매우 다양한데, 과거 연구는 주로 클래스 간 데이터 크기 불균형 해결에 초점을 두었다. 그러나 최근 연구에 따르면 데이터 수의 불균형만이 아니라, 클래스 간 중첩이 결합된 경우에 분류 성능의 저하가 더 심각해진다는 것이 밝혀졌다. 이에 따라 본 연구에서는 클래스 간 중첩 정도를 고려한 가중치 재샘플링 기법을 활용하는 효율적인 앙상블 분류 방법인 GC-Forest (GIR-based canonical forest)를 제안한다. 이 방법은 앙상블의 각 단계에서 데이터 개수의 불균형이 아닌 클래스 중첩 면에서 불균형 비율을 측정하고 소수 클래스의 대표성을 증가시킴으로써 클래스를 균형있게 맞춘다. 또한, 전체 분류 성능을 향상시키기 위해 GC-Forest 방법은 개별 분류기의 성능과 다양성을 모두 향상시키는 것으로 설계된 canonical forest 방법을 앙상블 분류기로 채택한다. 제안된 방법의 성능은 14개의 다양한 실제 불균형 데이터를 사용한 실험을 통해 비교 및 검증되었다. GC-Forest는 AUC, PR-AUC, G-mean, F1-score 측면에서 7개의 다른 앙상블 방법과 비교하여 매우 경쟁력 있는 분류 성능을 보여주었다.

주요용어: 분류, 불균형데이터, 앙상블, 불균형비율, 캐노니컬포레스트

김현중의 연구는 과학기술정보통신부 및 정보통신기획평가원의 학석사연계ICT핵심인재양성사업 (IITP-2023-00 259 934)과 한국연구재단(NRF) 연구비 (No. 2016R1D1A1B02011696)의 연구결과로 수행되었음.

[1]교신저자: (03722) 서울특별시 서대문구 연세로 50, 연세대학교 통계데이터사이언스학과. E-mail: hkim@yonsei.ac.kr