ISSN 2288-4866 (Print)
ISSN 2288-4882 (Online)
http://www.jiisonline.org
J Intell Inform Syst 2022 September: 28(3): 23~43 http://dx.doi.org/10.13088/jiis.2022.28.3.023

데이터셋 유형 분류를 통한 클래스 불균형 해소 방법 및 분류 알고리즘 추천

김정훈
국민대학교 비즈니스IT전문대학원
4단계 BK21 교육연구팀
(jeonghun0077@kookmin.ac.kr)

곽기영
국민대학교 경영대학/비즈니스 IT전문대학원
(kykwahk@kookmin.ac.kr)

AI(Artificial Intelligence)를 다양한 산업에서 접목하기 위해 알고리즘 선택에 대한 관심이 증가하고 있다. 알고리즘 선택은 대부분 데이터 과학자의 경험에 의해 결정되는 경우가 많다. 하지만 경험이 부족한 데이터 과학자의 경우 데이터셋 특성 기반의 메타학습(meta learning) 을 통해 알고리즘을 선택한다. 기존의 알고리즘 추천은 선정 과정이 블랙박스이기 때문에 어떠한 근거에 의해 도출되는지 알 수 없었다. 이에 따라 본 연구에서는 k-평균 군집분석을 활용하여 데이터셋 특성에 따라 유형을 나누고 적합한 분류 알고리즘과 클래스 불균형 해소 방법을 탐색한다. 본 연구 결과 네 가지 유형을 도출하였으며 데이터셋 유형에 따라 적합한 클래스 불균형 해소 방법과 분류 알고리즘을 추천하였다.

주제어 : 클래스 불균형, 메타학습, 데이터셋 유형, 군집분석, 데이터 특성

논문접수일 : 2022년 6월 16일 논문수정일 : 2022년 6월 30일 게재확정일 : 2022년 7월 4일
원고유형 : Regular Track 교신저자 : 곽기영

1. 서론

인터넷의 사용 기록, 모바일 거래 기록, 소셜 미디어, 센서, 네트워크 데이터의 급증과 함께 기존의 문서, 미디어 등의 디지털화가 가능해지면서 기업의 내외부에 축적되는 데이터가 증가하고 있다(George et al., 2014). 이러한 데이터는 바이오, 소셜, 생산, 금융, 통신 등 많은 분야에서 활발하게 활용되고 있다. 이에 따라 데이터 활용 능력은 기업뿐만 아니라 국가의 경쟁력을 결정하는 중요한 요소가 되었다. 많은 조직들은 데이터를 활용하여 새로운 가치를 창출할 수 있는 능력을 갖추기 위하여 많은 노력을 하고 있다. 하지만 데이터셋이 생성되는 과정에서 클래스의 불균형, 이상치, 결측값 문제 등이 흔히 발생한다. 따라서 이러한 문제들을 해결하기 위해 데이터셋의 특성에 따라 보정하는 다양한 방법이 연구되고 있다(Krawczyk, 2016). 첫째, 데이터셋 차원에서 데이터셋을 재추출(resampling)하거나, 특징공학(feature engineering)을 통해 데이터셋의 성격을 바꾸는 노력을 할 수 있다(Kim and Hong, 2015). 예를 들어 특징공학에서는 요인분석을 사용하여 성능을 높이려는 노력이 있었다(Dogan and Tanrikulu, 2013). 그러나 과도한 차원의 축소는 오히려 정확도를 떨어뜨리는 결과를 낳기도 한다(Blagus and Lusa, 2013). 둘째, 알고리즘의 하이퍼 패러미터(hyper parameter)를 조절하여 분류의 성능을 높이거나, 클러스터링의 정확도를

높이는 노력을 하고 있다. 하지만 알고리즘의 패러미터(parameter)를 조정하는 방법은 한계가 있다는 점을 고려할 때 데이터셋 차원에서의 보정이 보다 효과적인 것으로 알려져 있다(Japkowicz and Stephen, 2002; Weiss and Provost, 2003; Sun et al., 2009). 특히 클래스가 불균형을 이루고 있는 경우 샘플링 방법을 통한 성능의 향상이 더 효과적이다(Zhang et al., 2019; Khoshgoftaar et al., 2010). 클래스 불균형은 분류를 어렵게 하는 주요 원인 중 하나이다(Jo and Japkowicz, 2004). 따라서 클래스 불균형을 해소할 수 있는 방법들이 연구되고 있었다. 하지만 모든 상황에서 효과적인 성능을 내는 클래스 불균형 해소 방법은 없었다. 이에 따라 López, et al. (2013)은 데이터셋이 갖고 있는 특성들이 다르다는 점에 착안하여 분류 알고리즘과 다양한 클래스 불균형 해소 방법을 조합하여 분류 알고리즘별로 적합한 샘플링 방법이 존재할 수 있다는 것을 보여주었다. 하지만 적합한 데이터셋 보정 방법이나 분류 알고리즘을 선정하는 일은 데이터 과학자의 노하우와 지식에 근거하여 결정되기 때문에 비교적 경험이 적은 데이터 과학자들은 데이터셋 특성 기반 메타학습(meta learning)에 의존하는 경향이 있다. 그러나 기존의 메타학습은 머신러닝을 통해 추천되기 때문에 블랙박스라는 한계점이 존재한다(Muñoz et al., 2015). 따라서 명확한 선정 기준을 알 수 없기 때문에 분류 알고리즘이나 클래스 불균형 해소 방법을 선정하는 가이드라인을 설정하기 어렵다. 따라서 본 연구에서는 k-평균 군집분석을 통해 데이터셋 유형을 나누고 유형에 따라 적합한 클래스 불균형 해소 방법과 분류 알고리즘을 선정할 수 있는 가이드라인을 제시하고자 한다.

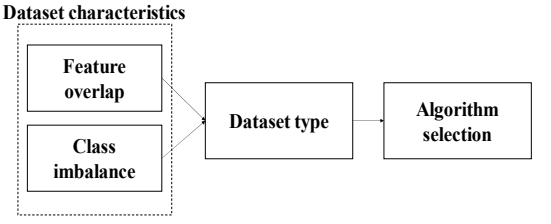
2. 관련연구

최적의 알고리즘을 선정하는 문제는 통계, 머신러닝, 공학 등의 전문 지식이 필요한 영역일뿐만 아니라, 데이터셋이 생성된 도메인에대한 지식과 경험이 요구된다 (Pfahringer et al., 2000, June; Blum et al., 2011). 특히 경험은 단시간에 얻을 수 없기 때문에 경험이 적은 데이터 과학자들은 어려움을 겪을 수 밖에 없다. 이를 보완하기 위하여 알고리즘 선정 문제에 메타학습을 적용하고 있다. 메타학습은 머신러닝 및 데이터마이닝 프로세스에 적용하여 자기개선 학습시스템을 구축하기 위해 메타지식을 사용하는방법으로 정의할 수 있다(Lee and Shin, 2018;Khan et al., 2020). 메타학습을 활용하여 다양한 알고리즘 선택이 연구되었다. 알고리즘 선택 문제는 판별분석(Kim and Kwon, 2021), 군집분석(Pimentel and De Carvalho, 2019), 인스턴스 선택(Leyva et al, 2014), 회귀분석(Lorena et al., 2018), 최적화(Muñoz et al., 2015), 시계열 분석(Rossi, 2014) 등이 있었다. 메타학습이 다양한 알고리즘 선정에 이용되고 있는 이유는 기본적으로 데이터셋의 성격이 다르고 데이터셋의 성격은 데이터셋이 갖고 있는 특성에 의해 결정되기 때문이다. 데이터의 특성은 일반적으로 생성되는 데이터셋의 원천에 의해 결정된다. 예를 들어, 의학 진단의 경우를 보면 어떠한 질병의 양성보다 음성의 경우가 많아 자연스럽게 클래스의 불균형이라는 특성이 생기게 된다. 센서에 의해 데이터가 생성되는 IoT(Internet of Things) 영역에서는 실시간으로 데이터가 생성되기 때문에 인스턴스의 개수가 많아진다. 이와 같이 데이터셋은 생성되는 원리와 주변의 환경에 의해 특성을 갖게 되며 이러한 데이터셋의 특성에 따라 분류성능에 있어

서도 차이가 발생한다(Van der Walt and Barnard, 2007). 상이한 특성을 갖고 있는 데이터셋의 경우 서로 다른 알고리즘이나 방법의 적용이 필요하겠지만 유사한 특성을 갖고 있을 때는 동일한 방법에 적용하여 유사한 결과를 얻을 수 있을 것으로 기대할 수 있다. 예를 들어, IoT 영역에서 생성되는 데이터셋과 공장에서 생성되는 데이터셋은 실시간성이라는 공통점을 갖고 있기 때문에 동일한 알고리즘이나 방법을 적용하였을 유사한 결과를 낼 것이라고 기대할 수 있다. 이와 같이 데이터셋은 생성 방식과 도메인에 따라 특성의 차이와 유사성이 존재한다(Kitchin and McArdle, 2016). 따라서 성능향상을 위해서는 데이터셋의 특성을 잘 이해하여 그에 적합한 분류 알고리즘을 적용하는 것을 고려할 필요가 있다(Oreski et al., 2017). 이에 따라 기존의 연구자들은 다양한 데이터셋 특성과 분류 알고리즘 성능 간의 관계를 밝히는 데 주목하였다. 예를 들어, Kiang(2003)은 로지스틱 회귀분석이 단봉분포(unimodal distribution)에서 성능이 우수하고, LDA(Latent Dirichlet Allocation)는 정규분포에서 성능이 우수함을 보여주었다. Marron et al.,(2007)은 거리기반의 분류 전략이 높은 차원의 데이터셋에서 적합하였으나, 인스턴스가 충분하지 않을 경우 문제가 발생함을 발견하였다.

이처럼 모든 데이터셋에는 다른 데이터셋과 차별화되고 분류를 용이하게 하거나 방해하는 특성들이 존재한다(Das et al., 2018; Pascual-Triana et al.,2021). Jo and Japkowicz(2004)는 데이터 분석에 가장 어려움을 주는 문제로 데이터 분리성과 클래스 불균형을 언급하였다. 특히 클래스 불균형 문제를 해결하기 위하여 다양한 방법들이 제시되었다. 대표적인 예로, 비용 민감 학습 방법(cost sensitive learning), 특징공학(feature engineering), 데이터 샘플링 방법(data sampling)이 있다. 본 연구에서는 그 가운데 데이터 샘플링 방법에 초점을 맞추어 연구를 진행하였다.

데이터셋 특성은 분류 알고리즘과 클래스 불균형 해소방법을 선택하는 데 있어 중요한 근거가 될 수 있다(Morán-Fernández et al., 2017; Kim and Kwon, 2021). 본 연구에서는 우선 데이터 복잡성 중 변수겹침과 클래스 불균형을 이용하여 특성에 따라 데이터셋의 그룹을 나눈다. 그리고 각 개별 그룹에 실험을 통해 적합한 샘플링 방법과 분류 알고리즘을 제안하고 비교한다. 전체적인 연구의 흐름은 <Figure 1>과 같다.



〈Figure 1〉 연구모형

3. 실험

3.1. 데이터셋 특성

클래스 불균형 문제는 소프트웨어 결함 진단(Feng et al., 2021), 오염 탐지(Lu and Wang, 2008), 위험 관리(Huang et al., 2006), 고객이탈 예측(Amin et al., 2016), 의학 진단(Pasupa et al., 2020) 등의 분야에서 중요하게 다루어지고 있다. 클래스 불균형 같은 데이터셋 특성은 대부분 도메인 특성에 따라 자연스럽게 결정된다(Kim and Kwon, 2021). 클래스 불균형은 분류 문제에서 분

류 성능을 매우 저하시키는 것으로 알려져 있다(Jo and Japkowicz, 2004; Krawczyk, 2016). Kim and Kwon(2021)은 클래스 불균형을 나타내는 지표로 HHI(Herfindahl–Hirschman index)를 이용하였다. HHI는 원래 엔트로피 지수(entropy index)와 같이 산업 집중도를 나타내는 지표이다(Matsumoto et al., 2012). HHI는 시장 내 모든 사업자의 각 시장점유율을 제곱한 후 합하여 산출한다. 기존의 IR(imbalance ratio)은 다수 클래스(majority class)와 소수 클래스(minority class)의 차이를 비율로 나타낸다. 이 방법은 전체적인 클래스의 균형 상태를 한눈에 알기 어렵다는 단점이 있다(Kim et al, 2020). 반면, HHI는 모든 클래스의 균형상태를 한번에 파악할 수 있다는 장점이 있다. 하지만 Kim and Kwon(2021)의 HHI는 클래스의 개수에 따라 균형을 나타내는 정도가 달라질 가능성이 존재하기 때문에 HHI에 클래스의 개수를 곱하여 1 이상인 경우 클래스 불균형이 있다고 판단한다. 식(1)은 클래스의 HHI를 산출하는 방법이다.

$$\text{HHI of Class} = \sum_{i=1}^{N} c_i^2 * N \text{ ---------- 식(1)}$$

여기서 N은 클래스의 개수, i는 클래스의 인덱스, c_j는 i번째 클래스의 비율

Ho and Basu(2002)는 변수겹침(feature overlap), 선형 분리성(linear separability), 기하학적 배치, 위상 배치와 밀도(geometry, topology and density)에 대해 측정방법을 제안하였으며, 데이터셋 특성을 정량적으로 나타낼 수 있는 기준을 제시하였다. 변수겹침 현상은 클래스 경계에 모호함을 발생시켜 분류를 어렵게 만든다(Ho, 2002; Anwar et al.,2014; Pascual-Triana et al., 2021). 변수겹침을 측정하는 방법에는 Maximum Fisher's Discriminant Ratio(F1), The Directional-vector Maximum Fisher's Discriminant Ratio(F1v), Volume of Overlapping Region(F2), Maximum Individual Feature Efficiency(F3), Collective Feature Efficiency(F4)등이 있다. 변수겹침 측정 방법 중 일부 연구에서 F1이 분류 성능과 밀접한 연관이 있는 것으로 나타났다(Cano, 2013; Kraiem et al.,2021). Kraiem et al.(2021)은 F1이 0에 가까워질수록 F1-score와 G-mean의 성능이 향상되는 것을 증명하였다. 또한 메타학습 분야에서도 F1이 머신러닝 알고리즘 선택, 샘플링 방법 선택에 중요한 요인으로 나타났다(Mollineda et al., 2005; Kraiem et al., 2021).

F2~F4는 클래스를 고려하지 않고, 특히 F2와 F3은 다수계급의 편향성을 갖고 있기 때문에 변수겹침 측정으로 적절하지 않다(Barella et al., 2018). 게다가 F2는 이산화 문제가 존재하고 적절한 이산화에 따라 값이 달라질 수 있기 때문에 적절하지 않다(Kotsiantis and Kanellopoulos, 2006). F1의 경우 겹침 정도가 가장 심한 값을 찾는 것이 효율적이다(Ho, 2002). 식(2)와 식(3)은 F1의 산출 산식이다.

$$f_i = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2} \text{ ---------- 식(2)}$$

$$\text{F1} = \text{MAX}(f_i) \text{ ---------- 식(3)}$$

여기서, $\mu_1, \mu_2, \sigma_1^2, \sigma_2^2$는 각각 두 클래스의 평균 및 분산, i는 변수의 인덱스

측정 결과의 범위는 0~+∞이고, F1 값이 작을수록 강한 중첩을 나타낸다. 본 연구에서는 F1값이 0.6 미만 값을 가질 때 강한 겹침, 0.6 이상 0.9

미만의 값은 조금 겹침, 0.9 이상의 값은 겹침 거의 없음으로 정의 한다. HHI는 1보다 커질수록 클래스 간 불균형이 크다는 것을 나타낸다. HHI의 경우 1.2 이하는 클래스 불균형 거의 없음, 1.2 초과 1.5 이하는 클래스 불균형 있음, 1.5 초과는 클래스 불균형이 심함으로 정의한다.

3.2. 메타데이터 수집

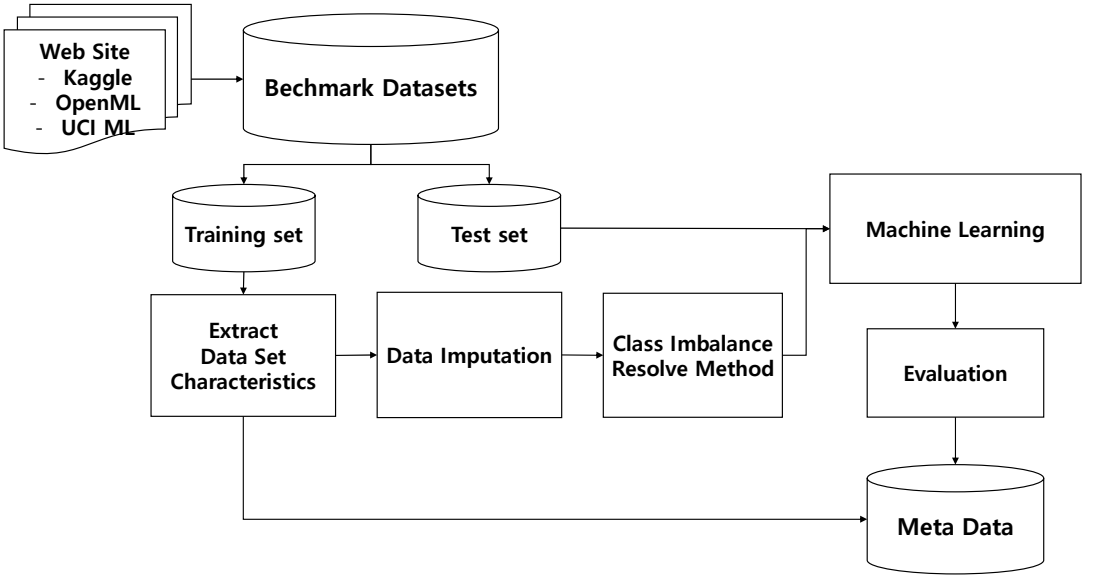
본 연구는 OpenML, Kaggle, UCI Machine learning repository에서 제공하는 데이터셋을 대상으로 실험을 수행하였다. OpenML, Kaggle, UCI Machine learning repository는 데이터 과학자들이 머신러닝 데이터셋, 실험 결과 또는 논문을 공유하는 온라인 플랫폼이다. 세 가지 플랫폼에서 제공하는 데이터셋 중 분류문제에 해당하는 데이터셋을 대상으로 하였다. 또한 클래스 불균형 문제가 존재하는 데이터셋을 선별하여 사용하였다. 클래스 불균형의 기준은 본 연구에서 고려한 클래스 HHI를 사용하였으며, 클래스 HHI가 1보다 큰 경우 클래스 불균형이 있다고 판단하였다. 수집된 데이터셋은 총 33개이다. 수집된 데이터셋의 정보는 <Table 1>과 같다.

메타데이터의 수집 방법은 <Figure 2>와 같이

〈Table 1〉 Characteristics of Dataset Used

Number of Classes	Number of Features	Number of Instances	HHI
28	9	4,177	2.008941
6	13	329	1.817699
4	7	1,845	1.115005
6	5	797	1.006726
8	41	1,000	1.129193
8	41	750	1.161894
6	30	1,742	1.00303
6	11	420	1.310855
2	8	169	1.166695
6	12	105	1.471489
11	4	5,052	1.871953
5	6	200	1.530000
7	17	13,611	1.210083
5	20	736	1.085209
3	22	2,126	1.896309
4	28	198	1.027518
8	9	1,000	1.907152
4	9	155	1.092107
3	7	44,819	1.275685

Number of Classes	Number of Features	Number of Instances	HHI
18	7	28,056	1.878369
10	25	3,200	1.001699
3	8	1,059	1.041801
11	15	17,335	1.60464
3	7	310	1.126951
3	11	478	1.168442
6	10	214	1.579695
4	3	1,024	1.189205
3	17	10,000	1.288248
7	28	1,941	1.554572
4	12	1,000	1.010011
5	27	2,800	2.13155
3	7	310	1.126951
4	33	1,891	1.068333



〈Figure 2〉 Meta Dataset Collection Process

수행하였다(Kim and Kwon, 2021). 먼저 웹에서 수집한 데이터셋을 10 폴드 교차검증(10-folds

cross validation) 방법을 사용하여 훈련 데이터셋과 검증 데이터셋으로 분할한다. 다음으로 훈련 데이터셋의 데이터셋 특성을 추출한다. 이때 추출되는 데이터셋 특성은 F1과 클래스 HHI이다. 데이터셋에 결측값이 있는 경우 결측값을 대체한다. 결측값은 수치형 데이터인 경우 평균값으로 대체하고, 명목형 변수인 경우 최빈값을 사용한다. 다음으로 클래스 불균형 해소방법을 적용한 후 분류 알고리즘을 수행한다. 클래스 불균형 해소방법의 비율은 과대표집방법은 식(4)와 같이 10~99%까지 10%단위로 다수 클래스를 증가시켰다. 또한 과소표집방법의 경우에는 식(5)와 같이 10%~99%까지 10%씩 소수 클래스를 감소시켰다.

$$Weighted\ Minority\ Class_c = (Majority\ class - Minority\ class_c) * raito$$ ----- 식(4)

여기서, *Minority Class*는 다수 클래스의 개수, *Minority Class*는 소수 클래스의 개수, *ratio*는 비율, c는 클래스의 인덱스

$$Weighted\ Majority\ Class = -Majoirty\ class * ratio$$ ---------- 식(5)

여기서, *Minority Class*는 다수 클래스의 개수, *ratio*는 비율

본 연구에서 고려된 분류 알고리즘에는 SVM (Support Vector Machine), 로지스틱 회귀분석, 랜덤포레스트, 나이브베이즈, knn(k-nearest neighbors), ANN(Artificial Neural Network)을 고려하였다. 이때, ANN의 경우 은닉층(hidden layer)을 3, 5, 7개로 설정하였다. 또한 클래스 불균형 해소 방법에는 과소표집방법(under sampling)과 과대표집방법(over sampling)을 사용하였다. 과대표집방법에는 Adasyn(Adaptive Synthetic Sampling), SMOTE (Synthetic Minority Over-sampling Technique), ROS(Random Over Sampling) 등이 있으며, 과소표집방법에는 NCR(Neighborhood Cleaning Rule), Tomek (Tomek Link), RUS(Random Under Sampling), CNN(Condensed Nearest Neighbor), ENN(Edited Nearest Neighbor) 등을 고려하였다.

3.3. 절차

본 연구의 실험은 크게 세 가지 단계를 거친다. 첫째, 메타데이터셋을 벤치마크 데이터셋 별로 요약한다. <Figure 3>는 메타데이셋 예시이며 메타데이터셋의 구성은 분류성능, 수행시간, 수행한 분류 알고리즘, 벤치마크 데이터셋의 이름, 샘플링 방법, 폴드, 클래스 HHI, F1, F1-score, 샘플링 비율로 이루어져 있다. 벤치마크 데이터셋 별로 요약하기 위하여 각 클래스 HHI, F1, F1-score를 평균으로 요약한다. <Figure4>는 평균 요약된 결과이다. 둘째, 평균 요약된 메타데이터셋 중 가장 좋은 분류 성능(F1-score)을 보여준 데이터만 선별하는 과정을 거친다. F1-score가 동률인 경우 G-mean이 높은 케이스를 선택하였고, G-mean이 동률인 경우 정확도(accuracy)가 더 높은 케이스를 선택하였다. 그럼에도 불구하고 동률인 경우 샘플링 비율이 최소값을 같는 경우를 선택하였다. 최종 요약된 데이터셋의 예시는 <Figure 5>와 같다. 최종적으로 도출된 메타데이터셋에는 데이터 이름, 클래스 불균형 해소 방법, 분류알고리즘, 샘플링 비율, F1-score, G-mean, 정확도, F1, 클래스 HHI, 인스턴스의 개수, 변수의 개수, 클래스의 개수가 포함되어 있다. 마지막으로 k-평균 군집분석을 실시한다. 이때, k는 4

Accuracy	G.mean	F.score	Elapsed	Algorithm	data	method	fold	HHI	F1	ratio	ID
0.916667	0.866025	0.941176	0.03	RF	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.1	blood_transfusion_service_center_Adasyn_fold_1_10%_RF
0.916667	0.924662	0.935065	0.01	knn	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.1	blood_transfusion_service_center_Adasyn_fold_1_10%_knr
0.55	0.519615	0.64	0.01	NB	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.1	blood_transfusion_service_center_Adasyn_fold_1_10%_NB
0.5	0.486056	0.583333	0	LR	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.1	blood_transfusion_service_center_Adasyn_fold_1_10%_LR
0.6	0.636396	0.6	0.02	SVM	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.1	blood_transfusion_service_center_Adasyn_fold_1_10%_SVI
0.833333	0.848528	0.864865	0.07	ANN3	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.1	blood_transfusion_service_center_Adasyn_fold_1_10%_AN
0.816667	0.824621	0.853333	0.19	ANN5	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.1	blood_transfusion_service_center_Adasyn_fold_1_10%_AN
0.883333	0.874643	0.911392	0.33	ANN7	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.1	blood_transfusion_service_center_Adasyn_fold_1_10%_AN
0.866667	0.774597	0.909091	0.03	RF	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.2	blood_transfusion_service_center_Adasyn_fold_1_20%_RF
0.866667	0.874643	0.894737	0	knn	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.2	blood_transfusion_service_center_Adasyn_fold_1_20%_knr
0.566667	0.547723	0.648649	0	NB	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.2	blood_transfusion_service_center_Adasyn_fold_1_20%_NB
0.533333	0.537355	0.6	0.02	LR	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.2	blood_transfusion_service_center_Adasyn_fold_1_20%_LR
0.6	0.636396	0.6	0.01	SVM	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.2	blood_transfusion_service_center_Adasyn_fold_1_20%_SVI
0.766667	0.762398	0.815789	0.08	ANN3	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.2	blood_transfusion_service_center_Adasyn_fold_1_20%_AN
0.883333	0.874643	0.911392	0.15	ANN5	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.2	blood_transfusion_service_center_Adasyn_fold_1_20%_AN
0.916667	0.924662	0.935065	0.23	ANN7	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.2	blood_transfusion_service_center_Adasyn_fold_1_20%_AN
0.966667	0.948683	0.97561	0.03	RF	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.3	blood_transfusion_service_center_Adasyn_fold_1_30%_RF
0.9	0.886707	0.925	0	knn	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.3	blood_transfusion_service_center_Adasyn_fold_1_30%_knr
0.566667	0.547723	0.648649	0	NB	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.3	blood_transfusion_service_center_Adasyn_fold_1_30%_NB
0.533333	0.537355	0.6	0.01	LR	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.3	blood_transfusion_service_center_Adasyn_fold_1_30%_LR
0.6	0.636396	0.6	0.02	SVM	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.3	blood_transfusion_service_center_Adasyn_fold_1_30%_SVI
0.616667	0.65192	0.596491	0.06	ANN3	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.3	blood_transfusion_service_center_Adasyn_fold_1_30%_AN
0.833333	0.793725	0.878049	0.09	ANN5	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.3	blood_transfusion_service_center_Adasyn_fold_1_30%_AN
0.866667	0.832917	0.902439	0.46	ANN7	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.3	blood_transfusion_service_center_Adasyn_fold_1_30%_AN
0.866667	0.815475	0.904762	0.03	RF	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.4	blood_transfusion_service_center_Adasyn_fold_1_40%_RF
0.916667	0.89861	0.938272	0.01	knn	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.4	blood_transfusion_service_center_Adasyn_fold_1_40%_knr
0.516667	0.479583	0.613333	0	NB	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.4	blood_transfusion_service_center_Adasyn_fold_1_40%_NB
0.566667	0.574456	0.628571	0.02	LR	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.4	blood_transfusion_service_center_Adasyn_fold_1_40%_LR
0.6	0.636396	0.6	0	SVM	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.4	blood_transfusion_service_center_Adasyn_fold_1_40%_SVI
0.633333	0.67082	0.62069	0.07	ANN3	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.4	blood_transfusion_service_center_Adasyn_fold_1_40%_AN
0.766667	0.762398	0.815789	0.11	ANN5	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.4	blood_transfusion_service_center_Adasyn_fold_1_40%_AN
0.9	0.9	0.923077	0.28	ANN7	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.4	blood_transfusion_service_center_Adasyn_fold_1_40%_AN
0.9	0.855132	0.928571	0.04	RF	blood_transfusion_service_center	Adasyn	1	1.097584	0.997166	0.5	blood_transfusion_service_center_Adasyn_fold_1_50%_RF

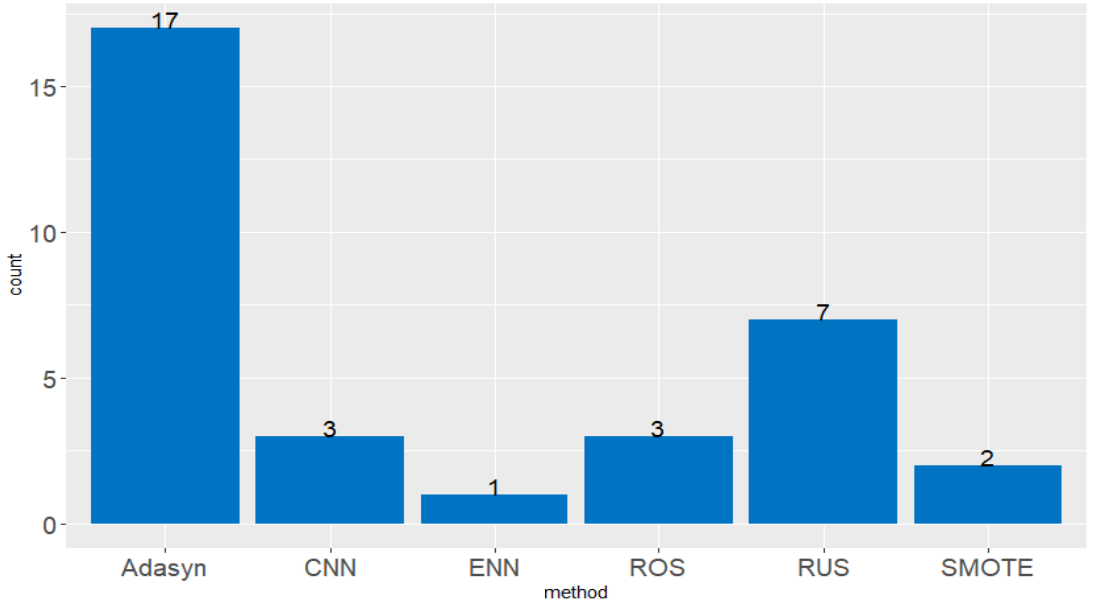
〈Figure 3〉 Example of Meta Dataset

data	method	Algorithm	ratio	F1_score	G.mean	Accuracy	F1	HHI
abalone	Adasyn	ANN3	0.1	0.029226694	0.022470359	0.028548422	0.633473699	2.008940954
abalone	Adasyn	ANN3	0.2	0.010416951	0.008655996	0.026873249	0.633473699	2.008940954
abalone	Adasyn	ANN3	0.3	0.033455429	0.02270858	0.028857229	0.633473699	2.008940954
abalone	Adasyn	ANN3	0.4	0.018704897	0.010769377	0.022549178	0.633473699	2.008940954
abalone	Adasyn	ANN3	0.5	0.019583503	0.010250297	0.039360575	0.633473699	2.008940954
abalone	Adasyn	ANN3	0.6	0.051712034	0.034395779	0.049228258	0.633473699	2.008940954
abalone	Adasyn	ANN3	0.7	0.030493791	0.021566304	0.024751411	0.633473699	2.008940954
abalone	Adasyn	ANN3	0.8	0.043013159	0.025005975	0.033165022	0.633473699	2.008940954
abalone	Adasyn	ANN3	0.9	0.078698288	0.046952777	0.047555519	0.633473699	2.008940954
abalone	Adasyn	ANN3	0.99	0.012767828	0.008002136	0.038002724	0.633473699	2.008940954
abalone	Adasyn	ANN5	0.1	0.024308227	0.007453978	0.012006194	0.633473699	2.008940954
abalone	Adasyn	ANN5	0.2	0.003508772	0	0.010088236	0.633473699	2.008940954
abalone	Adasyn	ANN5	0.3	0.00644343	0.00115818	0.004319482	0.633473699	2.008940954
abalone	Adasyn	ANN5	0.4	0.013035643	0.001338219	0.010560935	0.633473699	2.008940954
abalone	Adasyn	ANN5	0.5	0.015041209	0.004685294	0.015146759	0.633473699	2.008940954
abalone	Adasyn	ANN5	0.6	0.026731016	0.009567442	0.020201352	0.633473699	2.008940954
abalone	Adasyn	ANN5	0.7	0.011111111	0.002974738	0.015861594	0.633473699	2.008940954
abalone	Adasyn	ANN5	0.8	0.027544661	0.01165745	0.023064003	0.633473699	2.008940954
abalone	Adasyn	ANN5	0.9	0.021078341	0.007494888	0.017061799	0.633473699	2.008940954
abalone	Adasyn	ANN5	0.99	0.026684126	0.01358161	0.021836577	0.633473699	2.008940954
abalone	Adasyn	ANN7	0.1	0.019927328	0.001493452	0.005762942	0.633473699	2.008940954
abalone	Adasyn	ANN7	0.2	0.006780627	0.002225906	0.007694728	0.633473699	2.008940954
abalone	Adasyn	ANN7	0.3	0.003333333	0.000743685	0.007208751	0.633473699	2.008940954
abalone	Adasyn	ANN7	0.4	0.009276438	0.002857922	0.007923056	0.633473699	2.008940954
abalone	Adasyn	ANN7	0.5	0.008496732	0.002280068	0.011529421	0.633473699	2.008940954
abalone	Adasyn	ANN7	0.6	0.019953787	0.004778221	0.010569075	0.633473699	2.008940954
abalone	Adasyn	ANN7	0.7	0.019757285	0.008509653	0.013701653	0.633473699	2.008940954
abalone	Adasyn	ANN7	0.8	0.004819277	0	0.014402538	0.633473699	2.008940954
abalone	Adasyn	ANN7	0.9	0.007534008	0.003594009	0.011531159	0.633473699	2.008940954
abalone	Adasyn	ANN7	0.99	0.033540186	0.007949643	0.011530591	0.633473699	2.008940954

〈Figure 4〉 Example of Summarized Meta Dataset

data	method	Algorithm	ratio	F1_score	G.mean	Accuracy	F1	HHI	Number.of.features	Number.of.instances	Number.of.classes
Air Quality	RUS	knn	0.3	0.979453921	0.967492671	0.968566392	0.331427898	1.115004586	7	1845	4
analcatdata_dmft	SMOTE	SVM	0.5	0.299041379	0.263877856	0.205367741	0.974669981	1.006726352	5	797	6
autoUniv-au6-1000	Adasyn	RF	0.99	0.375013217	0.281732743	0.2449995	0.99168135	1.129192964	41	1000	8
autoUniv-au6-750	Adasyn	RF	0.1	0.437639405	0.32562809	0.267735723	0.989790849	1.161893549	41	750	8
Bangla Music Dataset	RUS	RF	0.8	0.752501832	0.698705216	0.710137381	0.80337613	1.003030191	30	1742	6
bird	Adasyn	ANN7	0.8	0.932405345	0.835049518	0.800926117	0.607390676	1.310855107	11	420	6
bmd	Adasyn	RF	0.4	0.901288161	0.847990384	0.864338235	0.927000068	1.166695322	8	169	2
bridges	Adasyn	knn	0.5	0.823080808	0.678587476	0.642380952	0.766328006	1.471489476	12	105	6
Color Classification 11 categories	Adasyn	RF	0.4	0.868884842	0.7402875	0.696928133	0.437130318	1.871952781	4	5052	11
drug200	Adasyn	RF	0.5	0.982972136	0.921005154	0.903982684	0.791952082	1.529999555	6	200	5
Dry_Bean	SMOTE	LR	0.5	1	0.961234988	0.926898635	0.25705938	1.210083028	17	13611	7
eucalyptus	Adasyn	RF	0.5	0.621119417	0.630484847	0.673983483	0.94347394	1.085208728	20	736	5
fetal_health	Adasyn	RF	0.1	0.971331136	0.900216914	0.946865691	0.888121021	1.89630939	22	2126	3
Forest Types	RUS	SVM	0.5	0.990909091	0.977240248	0.974736842	0.599512889	1.027517532	28	198	4
german_credit_data	Adasyn	NB	0.5	0.463850466	0.397034108	0.361492569	0.972929726	1.907152207	9	1000	8
grub-damage	Adasyn	RF	0.8	0.620887446	0.527014563	0.494460784	0.955266724	1.092107233	9	155	4
jungle_chess_2pcs_raw_endgame_complete	Adasyn	RF	0.5	0.849201797	0.811761048	0.813539378	0.937869871	1.275684727	7	44819	3
kropt	ROS	RF	0.5	0.798809524	0.830213222	0.715601898	0.964225254	1.87836913	7	28056	18
led24	CNN	SVM	0.9	0.81843225	0.758892902	0.720901516	0.822419544	1.001699044	25	3200	10
Milk Grading	Adasyn	RF	0.1	1	0.997774244	0.997169643	0.840875812	1.041801276	8	1059	3
Music Genre	CNN	RF	0.6	0.824215744	0.65496364	0.566713348	0.814434613	1.604640191	15	17335	11
orthopedic patients	RUS	RF	0.3	0.951393789	0.819520406	0.825806452	0.690523125	1.126951093	7	310	3
PopularKids	Adasyn	RF	0.99	0.52871676	0.502741015	0.508206687	0.980683689	1.168442028	11	478	3
prnn_fglass	RUS	LR	0.6	0.88	0.684191335	0.629366648	0.734677243	1.579695018	10	214	6
rmftsa_sleepdata	RUS	ANN3	0.2	0.576023697	0.069691765	0.367249152	0.94338486	1.189204706	3	1024	4
Skyserver_SQL2	Adasyn	RF	0.1	0.996523188	0.990735126	0.989299998	0.808793074	1.288248248	17	10000	3
steel-plates-fault	ENN	RF	0.1	0.897474747	0.821838353	0.791314472	0.751159171	1.554572261	28	1941	7
Telecust1	Adasyn	LR	0.4	0.382980663	0.389990579	0.40519552	0.98246545	1.010011166	12	1000	4
Video_games_esrb_rating	ROS	RF	0.4	0.896048883	0.869874514	0.855057442	0.913733695	1.068332813	33	1891	4

〈Figure 5〉 Final Summarized Meta Dataset



〈Figure 6〉 Frequency of Best Class Imbalance Resolution Methods

Abbreviations: RF = Random Forest, LR = Logistic Regression, knn = k-Nearest Neighbor, SVM = Support Vector Machine, ANN = Artificial Neural Network, NB = Naïve Bayes, Adasyn = Adaptive Synthetic Sampling, CNN = Condensed Nearest Neighbor, ENN = Edited Nearest Neighbor, SMOTE = Synthetic Minority Over-sampling Technique, NCR =Neighborhood Cleaning Rule, Tomek = Tomek Link, ROS = Random Over Sampling, RUS = Random Under Sampling, origin = None sampling method

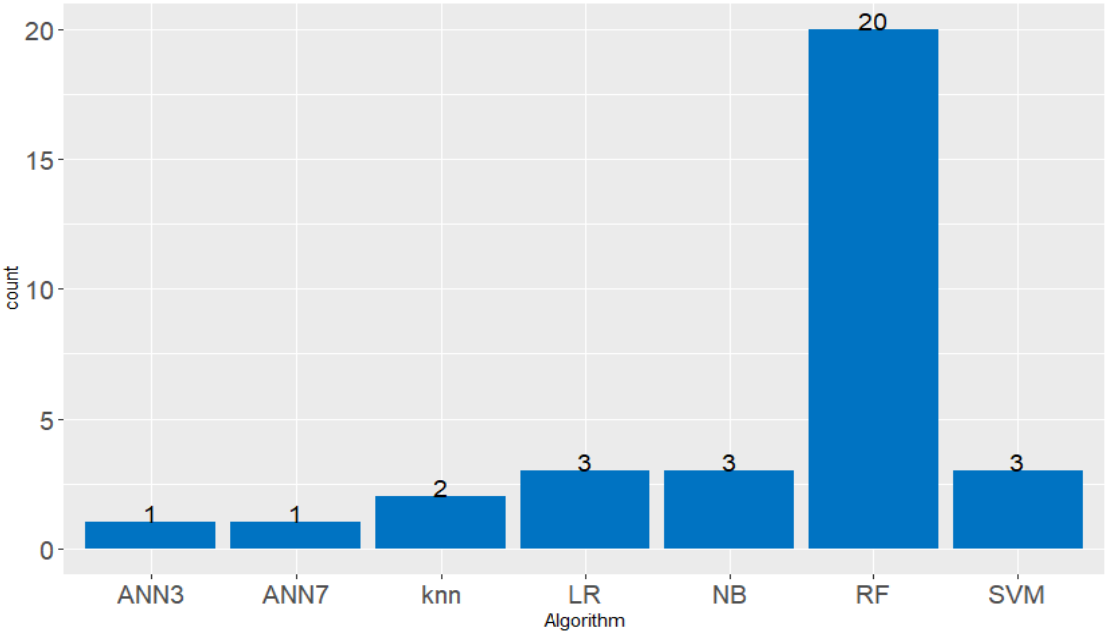
로 설정하였다. 최종적으로 도출된 군집별로 클래스 HHI와 F1을 비교하여 군집의 특징을 정의하고, 마지막으로 정의된 특성에 따라 우수한 알고리즘을 추천한다.

4. 결과

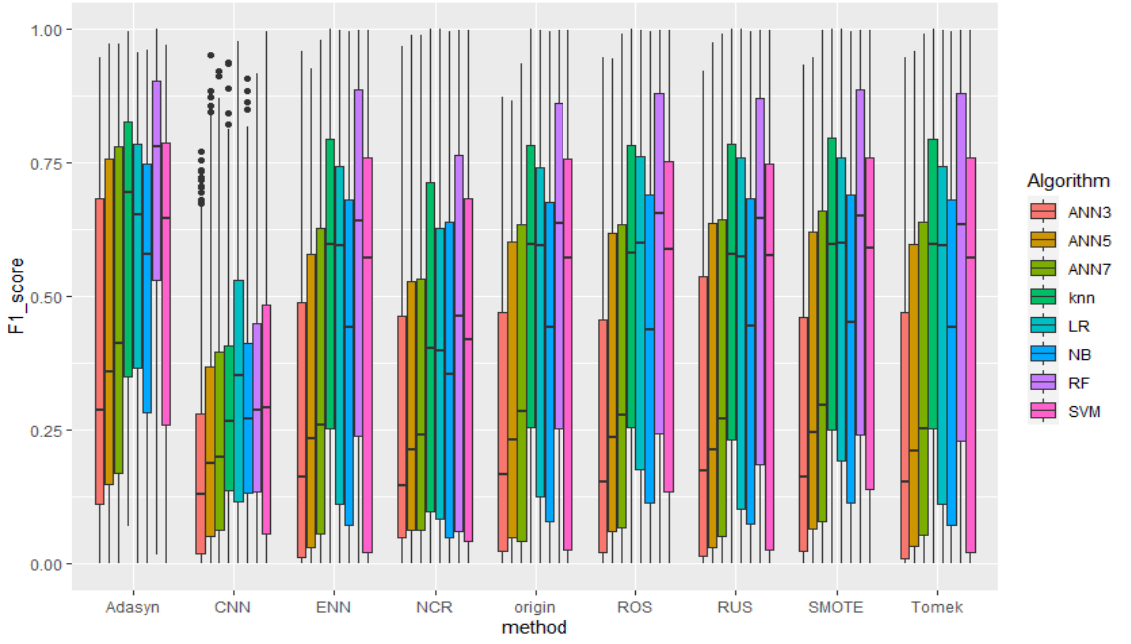
본 연구는 데이터셋 성격을 나누기 위하여 k-평균 군집분석을 사용하였으며, 이외에 분류 알고리즘과 샘플링 방법의 우수성을 평가하기 위해 정량적인 분석을 실시하였다.

4.1. 통계 분석

<Figure 6>와 <Figure 7>은 각 데이터셋에서 최고의 성능을 나타낸 클래스 불균형 해소 방법과 분류 알고리즘의 빈도를 그래프로 나타낸 결과이다. 먼저, 클래스 불균형 해소 방법의 빈도 그래프를 보면, Adasyn(adaptive synthetic sampling)방법이 가장 많이 선택된 것을 볼 수 있다. 다음으로 많이 선택된 방법은 RUS(Random Under Sampling)이다. <Figure 7>에서 가장 많이 선택된 분류 알고리즘은 랜덤포레스트이다. 그 밖에 로지스틱 회귀분석, SVM이 선택되었다. <Figure 8>은 샘플링 방법과 분류 알고리즘의 조합에 따른 성능을 상자도표로 나타낸 것이다. Adasyn과 랜덤포레스트의 조합이 가장 좋았으며, CNN과 분류 알고리즘의 조합은 샘플링 방법을 적용하지 않은 것(origin)보다 성능이 저조한 모습을 보였다. 이러한 이유는 CNN의 원리 때문이다. CNN방법은 최소한의 부분 집합을 찾고 클



〈Figure 7〉 Frequency of Best Classification Algorithms



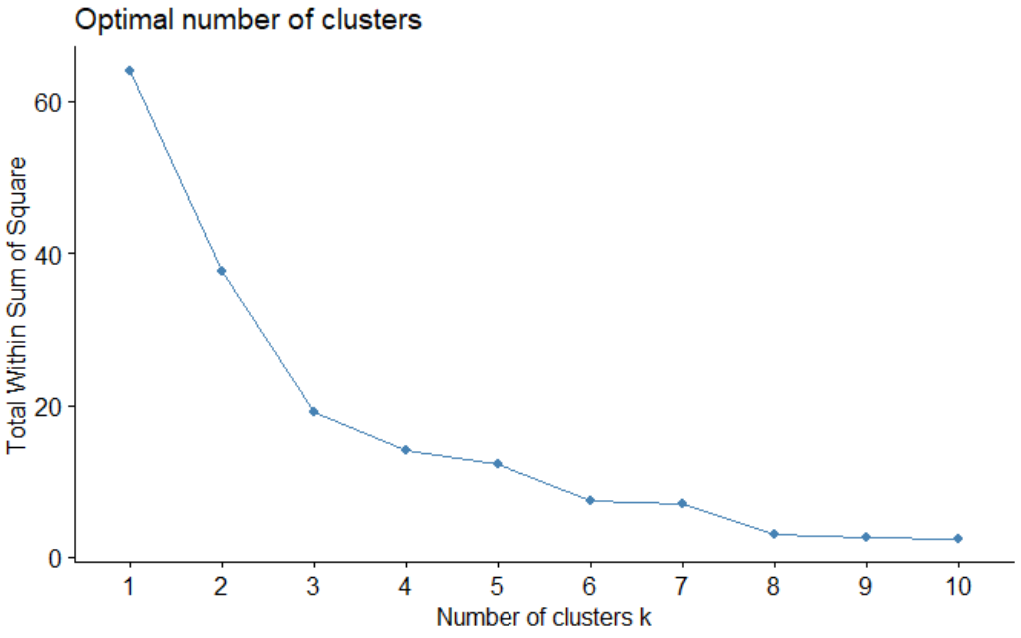
〈Figure 8〉 Boxplot of Classification Performance

래스 경계를 구하는데 소요되는 비용을 줄이는 것이 목적이지만 오히려 경계값이 겹쳐 분류가 쉽지 않은 상황을 만들기 때문이다(박근우와 정인경, 2019).

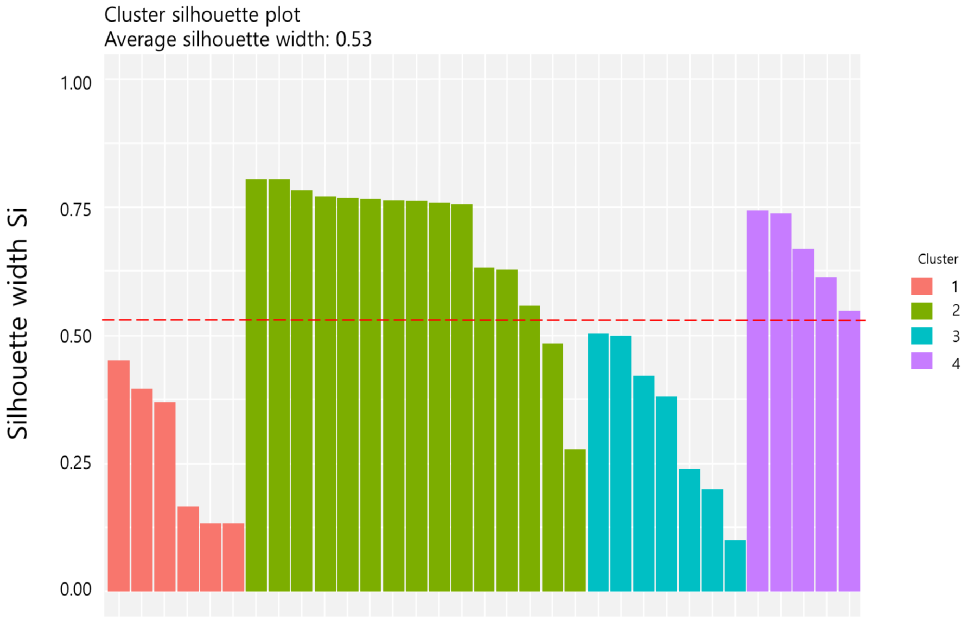
4.2. 데이터셋 유형 분류

군집에 의해 설명되는 분산을 확인한 엘보우 도표(elbow plot) 결과는 <Figure 9>과 같다. 군집에 의해 설명되는 분산을 통해 최적의 k가 4임을 확인할 수 있다. <Figure 10>은 실루엣 점수를 표현한 그래프이다. 그래프를 살펴보면 군집별로 확실한 차이가 있는 것을 볼 수 있다. <Table 2>와 <Table 3>는 군집 간의 평균 차이를 확인하기 위하여 ANOVA를 수행한 결과이다. <Table 2>와 <Table 3>를 살펴보면 유의수준 0.01에서 군집간의 F1과 HHI 평균차이가 있는 것을 알 수 있다. 추가로 사후분석을 통해 개별 군집 간의 F1과 HHI의 평균차이를 살펴보았다. F1의 평균차이에서 군집1과 군집3이 유사한 경향을 보였으며, 군집2와 군집4의 평균 차이가 거의 없는 것으로 나타났다. 또한 HHI는 군집1과 군집2가 유사하였으며, 군집3과 군집4가 유사한 것으로 나타났다. 따라서 군집은 대체로 잘 식별되었다고 볼 수 있다.

<Table 4>는 k-평균 군집분석을 수행하여 도출된 군집의 특성이다. <Table 4>에 나타난 값들은 각 군집의 평균값을 의미하며, 괄호안의 값은 각 군집의 표준편차를 의미한다. 군집의 특성을 토대로 다양한 분류 알고리즘과 클래스 불균형 방법이 제안되었다. 데이터셋의 특성에 따라 적합한 분류 알고리즘과 클래스 불균형 해소 방법이 다른 것으로 나타났다. 먼저 군집1을 살펴보



〈Figure 9〉 Optimal Number of Clusters



〈Figure 10〉 Silhouette Score Plot

〈Table 2〉 Result of ANOVA(F1)

N = 33							
		F1					
		n	Mean	Standard Deviation	F	p	Games- Howell
Cluster	Cluster 1[a]	6	0.529	0.188	22.14	0.000***	b<c, a<d
	Cluster 2[b]	15	0.921	0.068			
	Cluster 3[c]	7	0.704	0.131			
	Cluster 4[d]	5	0.930	0.043			
*p<.05, **p<.01, ***p<.001							

〈Table 3〉 Results of ANOVA(HHI)

N = 33							
		HHI					
		n	Mean	Standard Deviation	F	p	Games-Howell
Cluster	Cluster 1[a]	6	1.15	0.097	70.1	0.000***	a, b<c, d
	Cluster 2[b]	15	1.11	0.95			
	Cluster 3[c]	7	1.66	0.2			
	Cluster 4[d]	5	1.93	0.12			
*p<.05, **p<.01, ***p<.001							

면 F1이 0.529로 변수겹침이 심하고 HHI가1.153으로 클래스 불균형이 거의 없는 상태이다. 군집1은 클래스 불균형이 심하지 않고 변수겹침이 심해 클래스의 분류가 어려운 상태이다. 군집1에 적절한 분류 알고리즘으로는 knn, ANN7, LR, SVM, RF 등이 있었으며 클래스 불균형 해소 방법으로는 RUS, Adasyn, SMOTE등이 적합한 것으로 나타났다. 다음으로 군집2는 F1이 0.921로 변수겹침이 거의 없고 HHI가 1.113으로 클래스 불균형이 거의 없는 상태이다. 따라서 군집2는 가장 이상적인 데이터셋이 모여 있는 군집이다. 군집2에서는 SVM, RF, ANN3, LR 등의 분류 알고리즘과 SMOTE, Adasyn, RUS, CNN, ROS 등의 클래스 불균형 해소 방법이 우수하였다. 군집3은 변수겹침이 0.704로 변수겹침이 약간 존재하고, 클래스의 불균형이 HHI 1.66으로 심한 상태이다. 군집3은 클래스 불균형이 심하고 변수겹침도 강한 상태이기 때문에 분류가 가장 어려운 조건을 갖고 있다. 이러한 조건에서는 NB, knn, RF, LR 등의 분류 알고리즘과 CNN, Adasyn, RUS, ENN 등의 클래스 불균형 해소 방법이 가장 효과적이었다. 군집4의 경우 F1이 0.930으로

〈Table 4〉 Cluster Characteristics

	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Mean of F1	0.529 (0.026)	0.921 (0.235)	0.704 (0.184)	0.93 (0.199)
Mean of HHI	1.153 (0.026)	1.113 (0.235)	1.66 (0.184)	1.926 (0.199)
Mean of Number of Features	12.833 (8.4)	18.000 (12.989)	12.000 (7.937)	15.600 (8.591)
Mean of Number of Instance	2782.333 (5340.840)	4588 (11389.336)	4146.286 (6150.810)	6862.2 (11886.529)
Mean of Number of Classes	4.5 (1.643)	4.867 (2.295)	10.571 (8.059)	8 (5.874)
Suggested Classification Algorithms	knn, ANN7, LR, SVM, RF	SVM, RF, ANN3, LR	NB, knn, RF, LR	RF, NB
Suggested Class Imbalance Resolution Method	RUS, Adasyn, SMOTE	SMOTE, Adasyn, RUS, CNN, ROS	CNN, Adasyn, RUS, ENN	ROS, Adasyn

〈Table 5〉 Paired t-test(F1-Score)

(N=33)

		기술통계량			t(*p*)
		N	평균	표준편차	
F1-Score	클래스 불균형 해소방법 적용 전	33	0.615	0.307	-5.977(0.000)***
	클래스 불균형 해소방법 적용 후	33	0.772	0.219	

*p<.05, **p<.01, ***p<.001

거의 없는 상태이고 클래스 불균형은 HHI 1.926으로 매우 높아 소수 클래스를 분류하기 어려운 군집이다. 군집4에 최적인 방법은 RF와 NB 분류 알고리즘과 ROS와 Adasyn 클래스 불균형 해소 방법이었다. <Figure 11>은 군집분석 결과를 도식화한 것이다. <Table 5>는 클래스 불균형 해소 방법을 적용한 결과와 적용하기 전의 대응 표본 t-검정을 수행한 결과이다. 그 결과 유의수준 0.001에서 유의하게 차이가 있다. 따라서 본 연구에서 고려한 클래스 불균형 해소 방법이 유의하게 효과가 있는 것으로 나타났다.

〈Figure 11〉 Results of Dataset Personality Segmentation

5. 결론

5.1. 기여점

본 연구는 몇 가지 의의를 갖는다. 첫째, 데이터셋 특성별로 적합한 클래스 불균형 해소 방법과 분류 알고리즘이 존재함을 발견하였다. 변수겹침이 강하고 클래스 불균형이 거의 없는 경우는 knn, ANN, LR, SVM, RF와 같은 분류 알고리즘을 사용해야 하며, 클래스 불균형 해소 방법으로는 RUS, Adasyn, SMOTE 방법 등이 우수하다. 변수겹침이 거의 없고 클래스 불균형이 심한 경우에는 RF, NB와 ROS, Adasyn의 조합이 가장 이상적이다. 그리고 클래스 불균형이 심하고 변수겹침이 조금 있는 경우에는 NB, knn, RF, LR과 CNN, Adasyn, RUS, ENN 조합이 가장 적합한 것으로 나타났다. 또한 변수겹침이 없고 클래스 불균형이 거의 없는 경우는 SVM, RF, ANN, LR 등과 같은 분류 알고리즘이 적합하고 SMOTE, Adasyn, RUS, CNN, ROS 등과 같은 클래스 불균형 해소 방법이 적합하다. 둘째, 기존의 메타학습 방법은 머신러닝을 통해 추천되기 때문에 블랙박스라는 한계점이 있어 알고리즘 선정에 대한 이론적 이해가 부족하였다(Muñoz et al., 2015; Merz, 2004). 하지만 본 연구에서는 군집분

석 방법을 사용하여 데이터셋의 유형을 분류하고, 군집의 특성을 분석함으로써 해석 가능성을 높였다. 그 결과 네 가지 특성을 도출하였다. 군집의 특성을 파악할 수 있다면 보다 심층적인 분석을 하여 알고리즘을 선정할 수 있을 것으로 기대된다. 셋째, 성능에 긍정적인 영향을 미치는 분류 알고리즘과 클래스 불균형 해소 방법의 조합을 실험을 통해 보여주었다. 그 결과 Adasyn과 RF의 조합이 가장 우수한 성능을 나타냈다. 경험이 부족하거나 데이터셋에 대한 사전 조사가 부족한 경우 우선적으로 사용이 가능한 조합일 것이다.

5.2. 한계점과 향후 연구

본 연구는 몇 가지 기여점이 존재하지만 한계점 또한 존재한다. 첫째, 이미지, 텍스트와 같은 비정형 데이터셋에 대한 비교를 하지 못하였다. 비정형 데이터셋은 전처리 방식에 따라 분류성능의 차이가 많이 나기 때문에 본 연구에서는 제외하였다. 하지만 존재하는 데이터셋의 약 80%~90%가 비정형 데이터임을 감안할 때 추후 연구에서 꼭 고려해야한다(Qureshi and Gupta, 2014). 둘째, 최근 활발하게 연구가 진행중인 딥러닝 알고리즘을 다양하게 고려하지 못하였다. 본 연구에서 딥러닝 알고리즘을 다양하게 고려하지 못한 이유는 딥러닝 알고리즘들은 고려해야할 패러미터가 많고, 패러미터에 따라 성능의 차이가 존재하기 때문이다. 추후 연구에서는 딥러닝의 패러미터를 고려한 연구가 진행되어야 할 것이다. 셋째, 분류 알고리즘의 패러미터를 거의 조정하지 않았다. 패러미터를 조정하는 것 역시 데이터 과학자의 경험과 지식이 매우 필요한 영역이다. 하지만 본 연구에서는 디폴트 값을 사용하여 연구를 진행하였기 때문에 실제 실험에서는 패러미터에 의해 성능차이가 나타날 것으로 예상된다. 추후 연구에서는 패러미터를 고려한 메타학습 방법이 필요할 것이다. 넷째, 본 연구는 지도학습기반 추천에서 벗어나 비지도학습을 통해 이론적 가이드를 제시하였다는 점에서 기여점이 있다. 하지만 비지도학습방법은 성능의 측정이나 우수성을 객관적으로 증명하는 것이 어렵기 때문에 본 연구의 결과가 사용자의 경험에 따라 주관적으로 판단될 수 있다. 따라서 추후 연구에서는 관련 연구자들의 의견을 확보하여 제시하거나 준지도학습방법이 사용되어야 할 것이다.

5.3. 결론

본 연구는 기존의 메타학습 방법의 한계점인 블랙박스 모형을 벗어나 군집분석 방법을 채택하여 데이터셋 성격을 유형화하고 해석 가능하도록 하였다. 그 결과 네 가지 유형의 데이터셋 성격을 제시하였고 성격에 따라 어떠한 특성을 갖는지 파악하였다. 기존의 메타학습 방법들은 머신러닝 알고리즘을 활용하여 추천하는 것이 대부분이었다. 따라서 어떠한 유형의 데이터셋이 어떠한 기법에 적절한지 이론적인 근거를 제시하기 어려웠다.

이러한 결과는 실제 데이터 분석이나 인공지능 개발에 소요되는 비용과 시간을 줄여 줄 수 있을 것으로 예상된다. 사실 인공지능 개발에서 예측 정확도는 매우 중요한 요소이다. 많은 데이터 과학자들은 예측 정확도를 높이기 위해 많은 반복적 실험을 시도하고 있다. 그 이유는 데이터셋이 갖고 있는 특성이 서로 상이하고 데이터셋 생성의 원천이 다르기 때문이다. 이에 따라 적합

한 예측 알고리즘과 문제 해결방법을 찾기 위해 다양한 알고리즘을 적용한다. 이때 많은 자원들이 소모되고 있다. 특히 알고리즘의 수행속도를 높이기 위해 GPU(Graphics Processing Unit)와 TPU(Tensor Processing Unit)와 같은 하드웨어를 통해 학습 속도를 높이고 있다. 데이터 과학자들은 반복 실험을 하면서 많은 양의 탄소가 발생되고 있다(Strubell et al., 2019). 탄소 중립 시대를 맞이한 현재, 본 연구의 결과는 그린 AI(Green AI)를 실천할 수 있는 좋은 근거가 될 것이다.

현재의 연구에서는 일부 머신러닝 기법만을 사용하여 연구를 진행하였으며 분류 알고리즘의 패러미터에 대한 연구는 진행하지 않았다. 분류 알고리즘을 선정하는 일도 사용자의 경험과 전문적 지식을 필요로 한다. 따라서 추후 연구에서는 데이터셋의 특성에 따라 분류 알고리즘의 패러미터를 조절할 수 있는 메타학습이 필요할 것이다. 또한 정형 데이터셋 이외에 이미지 또는 텍스트와 같은 비정형 데이터셋에 대한 연구도 진행되어야 할 것이다.

참고문헌(References)

Amin, A., Anwar, S., Adnan, A., Nawaz, M., Howard, N., Qadir, J., ... & Hussain, A. (2016). Comparing oversampling techniques to handle the class imbalance problem: A customer churn prediction case study. *IEEE Access*, *4*, 7940-7957.

Anwar, N., Jones, G., & Ganesh, S. (2014). Measurement of data complexity for classification problems with unbalanced data. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *7*(3), 194-211.

Blagus, R., & Lusa, L. (2013). Improved shrunken centroid classifiers for high-dimensional class-imbalanced data. *BMC bioinformatics*, *14*(1), 1-13.

Cano, J. R. (2013). Analysis of data complexity measures for classification. *Expert systems with applications*, *40*(12), 4820-4831.

Dogan, N., & Tanrikulu, Z. (2013). A comparative analysis of classification algorithms in data mining for accuracy, speed and robustness. *Information Technology and Management*, *14*(2), 105-124.

Feng, S., Keung, J., Yu, X., Xiao, Y., Bennin, K. E., Kabir, M. A., & Zhang, M. (2021). COSTE: Complexity-based OverSampling TEchnique to alleviate the class imbalance problem in software defect prediction. *Information and Software Technology*, *129*, 106432.

George, G., Haas, M. R., & Pentland, A. (2014). Big data and management. *Academy of management Journal*, *57*(2), 321-326.

Ho, T. K. (2002). A data complexity analysis of comparative advantages of decision forest constructors. *Pattern Analysis & Applications*, *5*(2), 102-112.

Ho, T. K., & Basu, M. (2002). Complexity measures of supervised classification problems. *IEEE transactions on pattern analysis and machine intelligence*, *24*(3), 289-300.

Huang, Y. M., Hung, C. M., & Jiau, H. C. (2006). Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem. *Nonlinear Analysis: Real World Applications*, *7*(4), 720-747.

Jo, T., & Japkowicz, N. (2004). Class imbalances

versus small disjuncts. *ACM Sigkdd Explorations Newsletter*, *6*(1), 40-49.

Khan, I., Zhang, X., Rehman, M., & Ali, R. (2020). A literature survey and empirical study of meta-learning for classifier selection. *IEEE Access*, *8*, 10262-10281.

Khoshgoftaar, T. M., Van Hulse, J., & Napolitano, A. (2010). Comparing boosting and bagging techniques with noisy and imbalanced data. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *41*(3), 552-568.

Kim, J., & Kwon, O. (2021). A model for rapid selection and covid-19 prediction with dynamic and imbalanced data. *Sustainability*, *13*(6), 3099.

Kim, E., & Hong, T. (2015). Response Modeling for the Marketing Promotion with Weighted Case Based Reasoning Under Imbalanced Data Distribution. *Journal of Intelligence and Information Systems*, *21*(1), 29-45.

Kim, J., Kim, M. Y., & Kwon, O. (2020). The Effect of Meta-Features of Multiclass Datasets on the Performance of Classification Algorithms. *Journal of Intelligence and Information Systems*, *26*(1), 23-45.

Kotsiantis, S., & Kanellopoulos, D. (2006). Discretization techniques: A recent survey. *GESTS International Transactions on Computer Science and Engineering*, *32*(1), 47-58.

Krawczyk, B. (2016). Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, *5*(4), 221-232.

Lee, S., & Shin, T. (2018). Development and application of prediction model of hyperlipidemia using SVM and meta-learning algorithm. *Journal of Intelligence and Information Systems*, *24*(2), 111-124.

Leyva, E., González, A., & Perez, R. (2014). A set of complexity measures designed for applying meta-learning to instance selection. *IEEE Transactions on Knowledge and Data Engineering*, *27*(2), 354-367.

López, V., Fernández, A., García, S., Palade, V., & 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.

Lorena, A. C., Maciel, A. I., de Miranda, P. B., Costa, I. G., & Prudêncio, R. B. (2018). Data complexity meta-features for regression problems. *Machine Learning*, *107*(1), 209-246.

Lu, W. Z., & Wang, D. (2008). Ground-level ozone prediction by support vector machine approach with a cost-sensitive classification scheme. *Science of the total environment*, *395*(2-3), 109-116.

Matsumoto, A., Merlone, U., & Szidarovszky, F. (2012). Some notes on applying the Herfindahl–Hirschman Index. *Applied Economics Letters*, *19*(2), 181-184.

Merz, P. (2004). Advanced fitness landscape analysis and the performance of memetic algorithms. *Evolutionary Computation*, *12*(3), 303-325.

Muñoz, M. A., Sun, Y., Kirley, M., & Halgamuge, S. K. (2015). Algorithm selection for black-box continuous optimization problems: A survey on methods and challenges. *Information Sciences*, *317*, 224-245.

Park, G. U., & Jung, I. (2019). Comparison of resampling methods for dealing with

imbalanced data in binary classification problem. *The Korean Journal of Applied Statistics*, *32*(3), 349-374.

Pascual-Triana, J. D., Charte, D., Andrés Arroyo, M., Fernández, A., & Herrera, F. (2021). Revisiting data complexity metrics based on morphology for overlap and imbalance: snapshot, new overlap number of balls metrics and singular problems prospect. *Knowledge and Information Systems*, *63*(7), 1961-1989.

Pasupa, K., Vatathanavaro, S., & Tungjitnob, S. (2020). Convolutional neural networks based focal loss for class imbalance problem: a case study of canine red blood cells morphology classification. *Journal of Ambient Intelligence and Humanized Computing*, 1-17.

Pfahringer, B., Bensusan, H., & Giraud-Carrier, C. G. (2000, June). Meta-Learning by Landmarking Various Learning Algorithms. In *ICML* (pp. 743-750).

Pimentel, B. A., & De Carvalho, A. C. (2019). A new data characterization for selecting clustering algorithms using meta-learning. *Information Sciences*, *477*, 203-219.

Qureshi, S. R., & Gupta, A. (2014, March). Towards efficient Big Data and data analytics: A review. In *2014 Conference on IT in Business, Industry and Government (CSIBIG)* (pp. 1-6). IEEE.

Rossi, A. L. D., de Leon Ferreira, A. C. P., Soares, C., & De Souza, B. F. (2014). MetaStream: A meta-learning based method for periodic algorithm selection in time-changing data. *Neurocomputing*, *127*, 52-64.

Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and policy considerations for deep learning in NLP. *arXiv preprint arXiv:1906.02243*.

Sun, A., Lim, E. P., & Liu, Y. (2009). On strategies for imbalanced text classification using SVM: A comparative study. *Decision Support Systems*, *48*(1), 191-201.

Van der Walt, C. M., & Barnard, E. (2007). Data characteristics that determine classifier performance. *SAIEE Africa Research Journal*, *98*(3), 87-93.

Weiss, G. M., & Provost, F. (2003). Learning when training data are costly: The effect of class distribution on tree induction. *Journal of artificial intelligence research*, *19*, 315-354.

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, *1*(1), 67-82.

Zhang, X., Li, R., Zhang, B., Yang, Y., Guo, J., & Ji, X. (2019). An instance-based learning recommendation algorithm of imbalance handling methods. *Applied Mathematics and Computation*, *351*, 204-218.

Abstract

Class Imbalance Resolution Method and Classification Algorithm Suggesting Based on Dataset Type Segmentation

Jeonghun Kim* · Kee-Young Kwahk**

In order to apply AI (Artificial Intelligence) in various industries, interest in algorithm selection is increasing. Algorithm selection is largely determined by the experience of a data scientist. However, in the case of an inexperienced data scientist, an algorithm is selected through meta-learning based on dataset characteristics. However, since the selection process is a black box, it was not possible to know on what basis the existing algorithm recommendation was derived. Accordingly, this study uses k-means cluster analysis to classify types according to data set characteristics, and to explore suitable classification algorithms and methods for resolving class imbalance. As a result of this study, four types were derived, and an appropriate class imbalance resolution method and classification algorithm were recommended according to the data set type.

Key Words : Class Imbalance, Meta Learning, Dataset Type, Clustering Analysis, Data characteristics

Received : June 16, 2022 Revised : June 30, 2022 Accepted : July 4, 2022
Corresponding Author : Kee-Young Kwahk

* Graduate School of Business IT, Kookmin University
** Corresponding author: Kee-Young Kwahk
College of Business Administration / Graduate School of Business IT, Kookmin University
77 Jeongneung-ro, Seongbuk-gu, Seoul 136-702, Korea
Tel: +82-2-910-4738, Fax: +82-2-910-4017, E-mail: kykwahk@kookmin.ac.kr

저 자 소 개

김 정 훈
현재 국민대학교 비즈니스IT전문대학원에서 연구원으로 재직 중이다. 목원대학교에서 경제학사, 경희대학교에서 경영학 석사 및 박사를 취득하였다. 주요 연구 분야는 텍스트 마이닝, 머신러닝, 빅데이터 분석, HRI(Human Robot Interaction) 등이다.

곽 기 영
현재 국민대학교 경영대학과 비즈니스IT전문대학원 교수로 재직 중이다. 서울대학교 경영대학을 졸업하고 KAIST 경영과학과와 테크노경영대학원에서 석사 및 박사학위를 취득하였다. 주요 연구관심분야는 Social network analysis and its application, Data analytics, Users' behavior in social media, Social communication ecology, Knowledge management 등이다.