사례기반추론 및 의사결정나무 실습 보고서

Y2023011 윤요섭

Ⅰ. 자료 및 분석 방법

본 보고서는 bankruptcy.xlsx 데이터를 이용하여 IBM SPSS Statistics를 통해 사례기반추론 및 의사결정나무 실습과제를 수행하였다. bankruptcy.xlsx 데이터 변수와 이에 대한 설명은 다음과 같다.

변수명	변수설명
CLASS	데이터셋 구분 (1=학습, 0=검증)
X13_1	금융비용대부채비율
X18_1	매출원가비율
X36_1	자기자본비율
X42_1	금융비용부담율증가분
X43_1	매입채무회전율
X74_1	지급여력도
X76_1	분식계수2
X98_1	기업경상이익율
X107_1	현금흐름7대전기총부채
X118_1	총자산변동계수
X129_1	자산대비금융비용증가율
X130_1	자산대비영업외비용증가율
X133_1	매출원가율*매출원가평균증가비
X149_1	매출대비재고자산증가율
X157_1	총자본회전율*매출액증가비
Y	부실여부(1=부실, 0=건전)

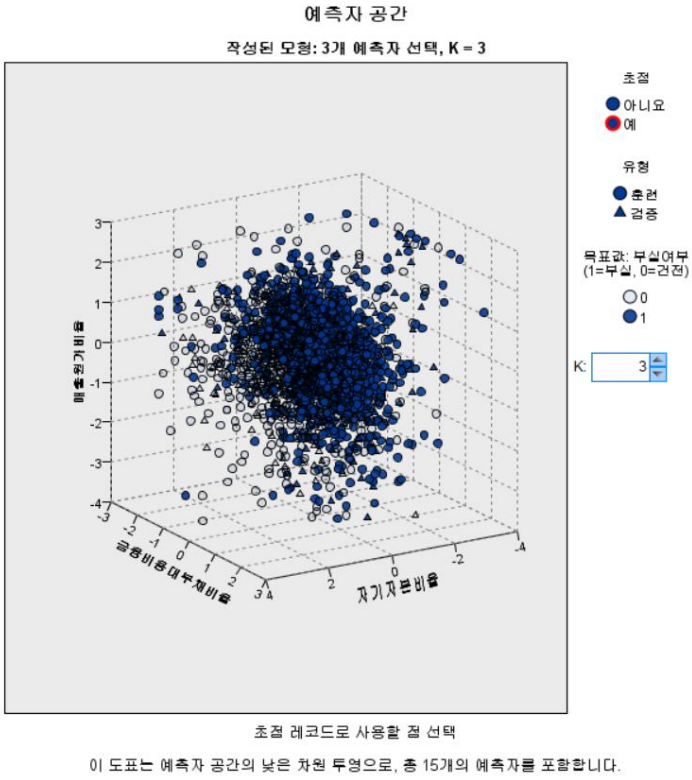
bankruptcy.xlsx 데이터를 통해 사례기반추론 및 의사결정나무 실습을 실시하였으며, 이 때, CLASS 변수가 1인 경우 학습용 데이터로 0인 경우 테스트용 데이터로 활용하였다. 종속변수로는 Y(부실여부)로 설정하였으며, CLASS변수와 Y변수 외의 변수를 독립변수로 설정한 뒤, 학습을 진행하였다.

Ⅱ. 분석 과정과 결과 및 해석

Ⅱ-가 사례기반추론 최근접 이웃을 통한 분석

K-Nearest Neighbor 알고리즘은 사례기반추론에서 사용되는 가장 대표적인 알고리즘으로, 새로운 입력 데이터를 분류하기 위해, 가장 가까운 k개의 데이터 포인트를 찾아서 그들의 다수결을 통해 분류를 결정하는 알고리즘이다. 실습에서는 K를 Default Value인 3으로 결정하고 실습을 진행하였다.

K-최근접 이웃 알고리즘에서 유사한 이웃 사례를 선택하기 위해, 적절한 유사도 평가함수(거리 함수)를 결정해야 하는데, 두 점 사이의 직선 거리를 계산하는 가장 기본적인 거리 측정방법인 유클리드 거리로 결정하고 실습을 진행하였다.



분류 테이블

분할	관측빈도	예측 0	예측 1	정확도 비율
훈련	0	871	196	81.6%
	1	180	887	83.1%
	전체 비율	49.3%	50.7%	82.4%
검증	0	224	44	83.6%
	1	47	221	82.5%
	결측	0	0	
	전체 비율	50.6%	49.4%	83.0%

실습 결과, 훈련데이터의 경우, 82.4%의 정확도를 보였으며, 검증데이터의 경우, 83.0%의 정확도를 보인 것을 알 수 있다.

Ⅱ-나 의사결정나무를 통한 분석

의사결정나무(Decision Tree) 모델은 일련의 독립변수들을 활용하여 주어진 문제에 답을 찾아가는 나무(Tree) 형태의 모델이다. 분류와 회귀에 모두 적용이 가능한데, CART(Classification and Regression Tree) 알고리즘과 같은 결정 트리 학습 알고리즘의 유연성과 범용성 때문이다. DecisionTreeClassifier와 DecisionTreeRegressor의 차이점은 아래와 같다.

DecisionTreeClassifier와 DecisionTreeRegressor 차이점

- DecisionTreeClassifier는 분류 문제를 다루기 위한 모델이며, DecisionTreeRegressor는 회귀문제를 다루기 위한 모델이다.
- DecisionTreeClassifier는 클래스 레이블을 예측하며 이산적인 값을 가지나, DecisionTreeRegerssor는 타겟 값을 예측하며 연속적인 값을 가진다.

- DecisionTreeClassifier는 분류 문제에서 사용되는 손실 함수인 지니계수와 엔트로피를 최소화하면서 분류하나, DecisionTreeRegressor는 회귀 문제에서 사용되는 손실 함수인 평균 제곱 오차(MSE)나 평균 절대 오차(MAE)를 최소화하면서 예측값을 구한다.

의사결정나무는 최종 결과물이 일련의 IF-THEN 규칙들로 표현되는 특성을 가지고 있다.

의사결정나무의 2가지 핵심 아이디어는 반복적 분할과 가지치기인데, 분할을 반복하여 모든 데이터를 100% 정확히 분류해 낼수 있을 만큼 세분화해나 갈수 있다. 이럴 경우 Overfitting에 빠질 문제가 생기는데 이를 방지하여 불필요한 가지를 제거함으로 써 나무를 단순화하는 작업이 이루어진다.

의사결정나무를 통해 실습한 결과는 아래와 같다.

```
/* Node 4 */.
IF (((자산대비금융비용증가율 NOT MISSING  AND (자산대비금융비용증가율 = 0.0468393595)) OR
자산대비금융비용증가율 IS MISSING  AND ((자산대비영업외비용증가율 NOT MISSING  AND
(자산대비영업외비용증가율 = 0.1216523335)) OR 자산대비영업외비용증가율 IS MISSING  AND
((금융비용부담율증가분 NOT MISSING  AND (금융비용부담율증가분 = 0.24822571)) OR
금융비용부담율증가분 IS MISSING  AND ((총자산변동계수 NOT MISSING  AND (총자산변동계수 =
0.20729694)) OR 총자산변동계수 IS MISSING  AND ((매출원가율*매출원가평균증가비 NOT MISSING  AND
(매출원가율*매출원가평균증가비 = -0.09428412450000001)) OR 매출원가율*매출원가평균증가비 IS MISSING
AND ((자기자본비율 NOT MISSING  AND (자기자본비율 > -0.5812748595)) OR 자기자본비율 IS MISSING
AND ((매입채무회전율 NOT MISSING  AND (매입채무회전율 > -0.337639881)) OR 매입채무회전율 IS
MISSING  AND ((지급여력도 NOT MISSING  AND (지급여력도 > -0.4382566475)) OR 지급여력도 IS
MISSING  AND ((금융비용대부채비율 NOT MISSING  AND (금융비용대부채비율 = 0.5532949094999999)) OR
...
THEN
Node = 4
Prediction = 0
Probability = 0.968553
```

출력 결과를 일부를 가져와서 보면 IF-THEN 규칙에 따라서 변수의 값이 일정값에 해당할 때, Node, Prediction, Probability 값을 확인 할 수 있다. 이를 통해 각 분기점에 대한 정보와 예측된 클래스의 값, 예측된 클래스의 확률의 정보를 확인 할 수 있다.

의사결정나무로 분류하였을 때, 분류표는 아래와 같다.

분류

표본	감시됨	예측		
		0	1	정확도(%)
훈련	0	922	145	86.4%
	1	176	891	83.5%
	전체 퍼센트	51.5%	48.5%	85.0%
검정	0	219	49	81.7%
	1	48	220	82.1%
	전체 퍼센트	49.8%	50.2%	81.9%

성장방법: CRT

종속변수: 부실여부(1=부실, 0=건전)

훈련 데이터의 경우, 85%의 정확도가 나왔으며, 검정 데이터의 경우 81.9%가 나온 것을 확인 할 수 있다. 사례기반 추론을 통한 실습에서는 훈련 데이터의 정확도보다 검증 데이터의 정확도가 높은 반면, 의사결정나무를 통한 실습에서는 훈련 데이터의 정확도보다 검증 데이터의 정확도가 낮은 것을 확인 할 수 있었다.

이를 통해, 반복적 분할 과정에서 가지치기를 더 많이 하여 학습한다면 Overfitting의 위험에서 더 벗어날 수 있을 것이다.