DS 05주차 수업 팀과제

Decision Tree

팀 AOA

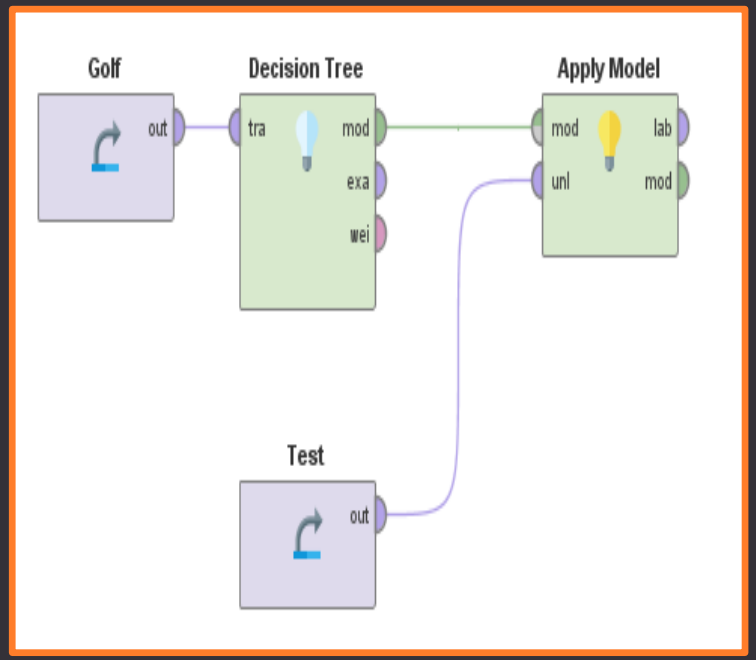
경영학과 2014109050 박한솔
경영학과 2015126062 신지수

00
서 론

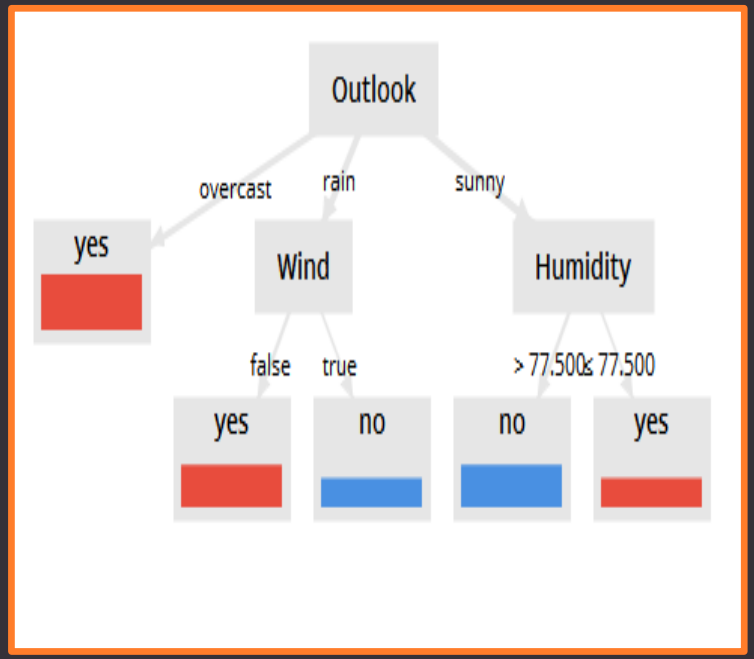
데이터 마이닝(decision tree) 작업의 최종 목적은?

예측 모델의 정확도 향상 및 규칙 발견

01
Golf Dataset decision tree

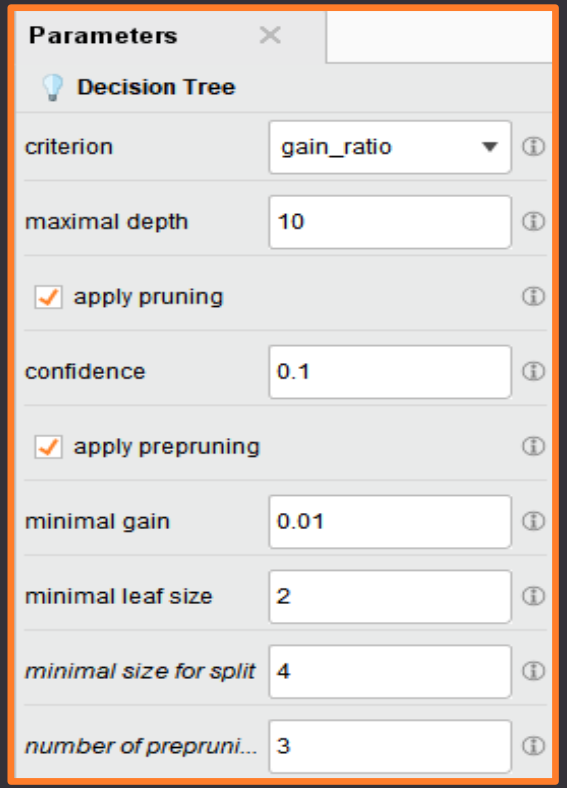


〈process〉



〈decision tree〉

02
Decision tree-Paramter



Criterion: 최선의 split을 위한 기준(방법)을 정하는 파라미터

maximal depth: 트리의 크기를 조정. 즉, 부모 노드로부터 분할을 언제 멈출 것인지 결정. 만약 1일 경우에는 부모 노드만 존재.

Pruning: 과적합을 예방하기 위해 tree의 성장을 제한

Confidence: pruning의 신뢰도를 지정.

Prepruning: tree가 성장하기 전 미리 pruning

Minimal gain: 노드의 gain을 분할하기 전 미리 계산. 그 gain이 최소 gain보다 크다면 분할됨. Minimal gain 값이 높을수록 분할이 적고 tree가 작아짐.

Minimal leaf size: pruning의 최소 잎 사이즈

Minimal size for split: pruning의 최소 스플릿 사이즈

Number of prepruning alternative: 특정 노드에서 분할이 방지된 경우에

매개변수가 분할 테스트된 대체 노드 수를 조정.

03
Accuracy

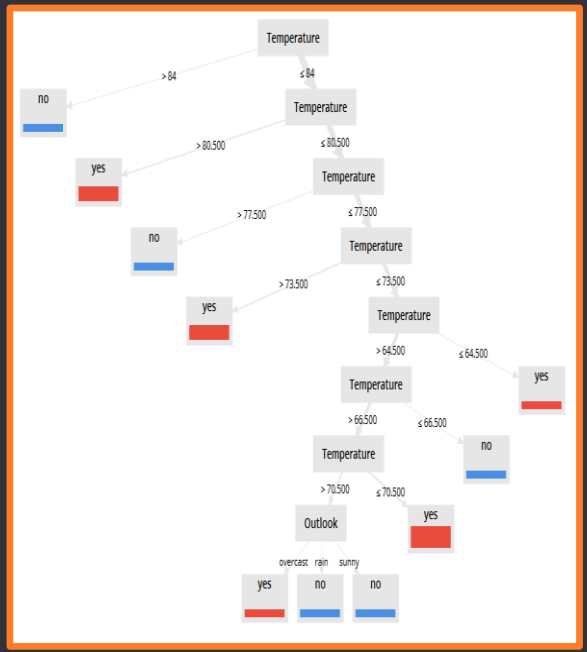
accuracy: 64.29%

	true no	true yes	class precision
pred. no	3	3	50.00%
pred. yes	2	6	75.00%
class recall	60.00%	66.67%	

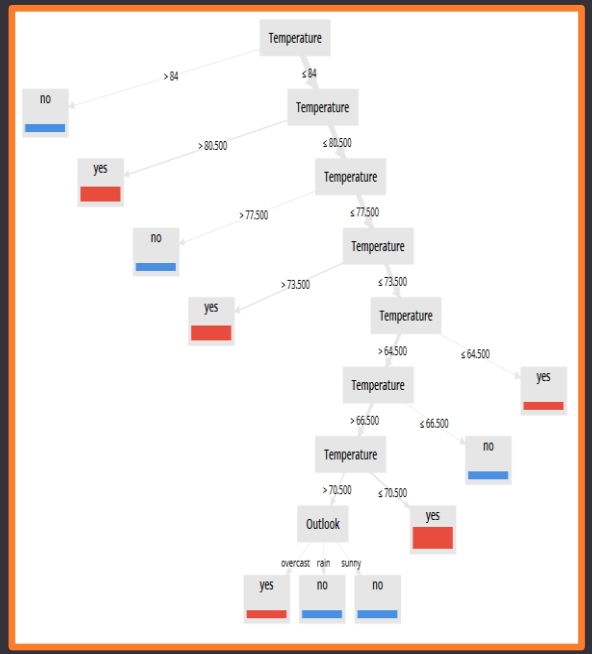
Precision: 정밀도(양의 예측값), 예측 모델이 true라고 분류한 것 중에서 실제 true인 것의 비율을 보여준다.
ex. true yes/(true no + true yes) 정밀도는 검색된 데이터가 원하는 정보와 관련이 있을 확률을 알려준다.

Recall: 재현율, 전체 데이터 중 관련 데이터의 비중을 뜻하며 실제 true인 것 중에서 예측 모델이 true라고 예측한 것의 비율을 말한다.
ex. pred no/true no 재현율은 원하는 정보가 검색될 확률이라 할 수 있다.

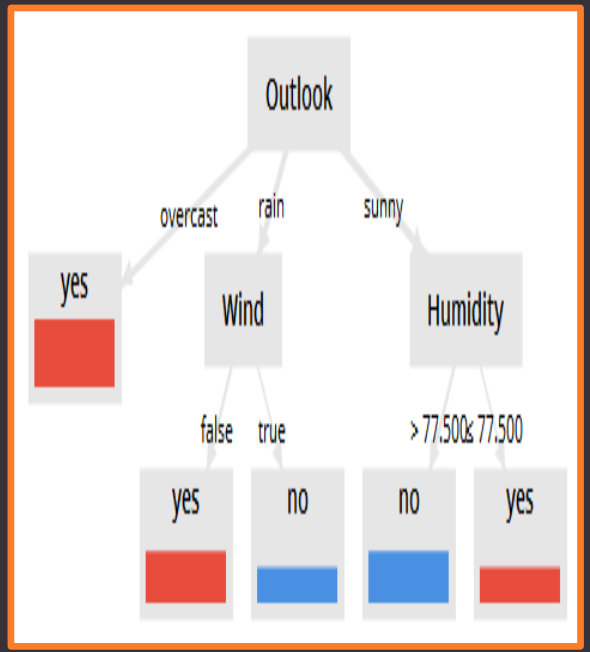
04
Split criterion과 Pruning여부에 따른 tree



〈no pruning〉
Accuracy: 71.43%



〈pruning〉
Accuracy: 71.43%



〈pre pruning〉
Accuracy: 64.29%

05
GINI index (pre pruning)

=B3/(B3+B4+B5)*E3+B4/(B3+B4+B5)*E4+B5/(B3+B4+B5)*E5

A	B	C	D	E	F	G
	node_total	node_yes	node_no	GINI_index		GINI_Child
root	14	10	4	0.40816327		0.34285714
node 1	4	4	0	0		
node 2	5	3	2	0.48		
node 3	5	2	3	0.48		
node 4(2-1)	3	3	0	0		
node 5(2-2)	2	0	2	0		
node 6(3-1)	3	0	3	0		
node 6(3-2)	2	2	0	0		

각 GINI index를 이용하여 자식 노드들의 전체 GINI를 구하면 0.342가 나온다.

부모 노드의 GINI 0.408과 비교해 볼 때, 값이 0.06정도 줄어든 것을 확인할 수 있다.

그러므로 Split 이후 정확도가 높아졌음을 알 수 있다.

06
예측 규칙 생성

Rule 1 : if(Outlook = overcast) then Play = yes

Rule 2 : if(Outlook = rain) and (Wind = false) then Play = yes

Rule 3 : if(Outlook = rain) and (Wind = true) then Play = no

Rule 4 : if(Outlook = sunny) and (Humidity 〉 77.5) then Play = no

Rule 5 : if(Outlook = sunny) and (Humidity 〈= 77.5) then Play = yes

위 규칙은 각 예측 변수가 루트에서부터 단말 노드까지 테스트를 거치며 내려오는 과정이 담겨있다.

07
규칙 분석

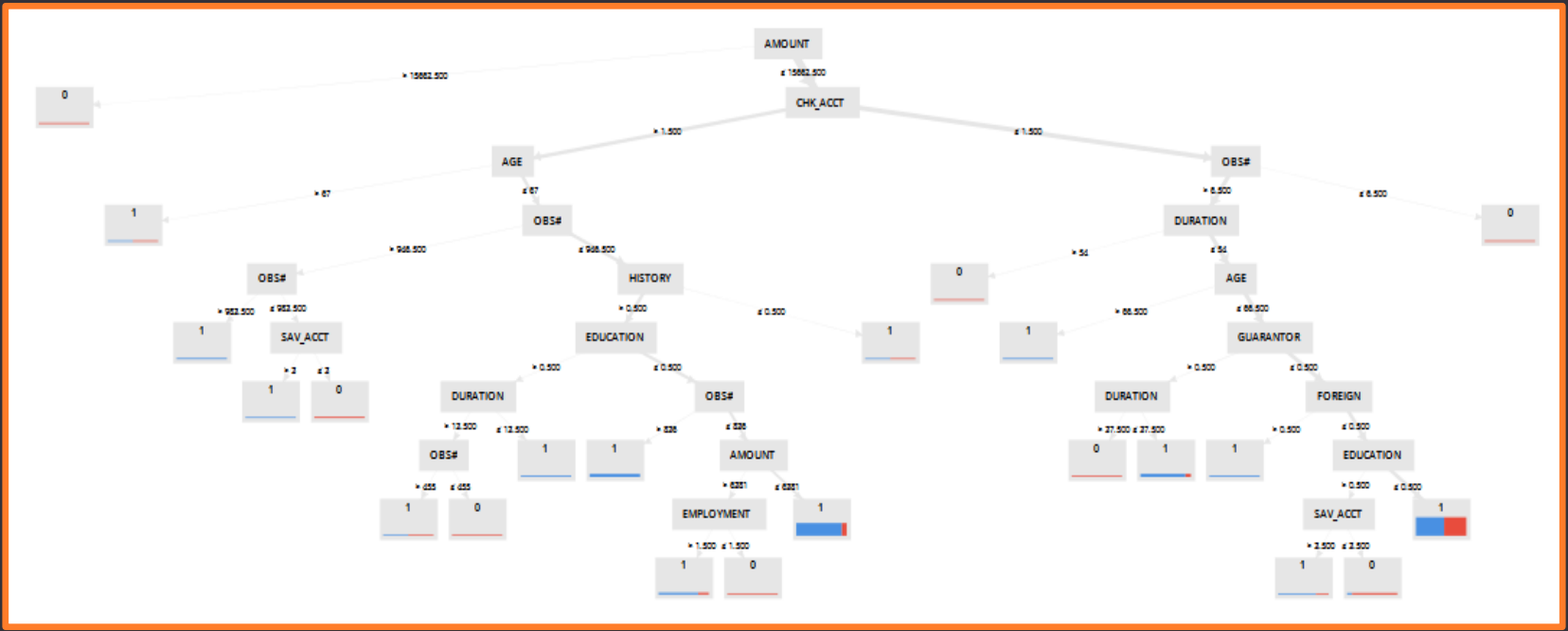
ⅰ. 규칙 중 가장 중요한 규칙은 무엇인가?

여기서 말하는 가장 중요한 규칙이란 가장 많은 사례, 사람에게 적용할 수 있는 규칙이라 해석할 수 있다.
Golf 학습 모델에서 가장 중요한 규칙은 Rule 1 : if(Outlook = overcast) then Play = yes이다.
1번 규칙은 14개 사례 중 4개 사례에 적용 가능하며, 이는 규칙 중 최다 사례에 적용되는 규칙이기 때문이다.

ⅱ. 가장 정확한 규칙은 무엇인가?

가장 정확한 규칙이라 함은 잎노드 즉 단말 노드의 분산이 0이 되는, 단말 노드의 순수성이 가장 높은 규칙이라 할 수 있다.
Golf 학습 모델에서는 단말 노드 모두가 분산이 0이 된 상태, 즉 불순도가 0인 상태이므로 모든 규칙이 정확하다.

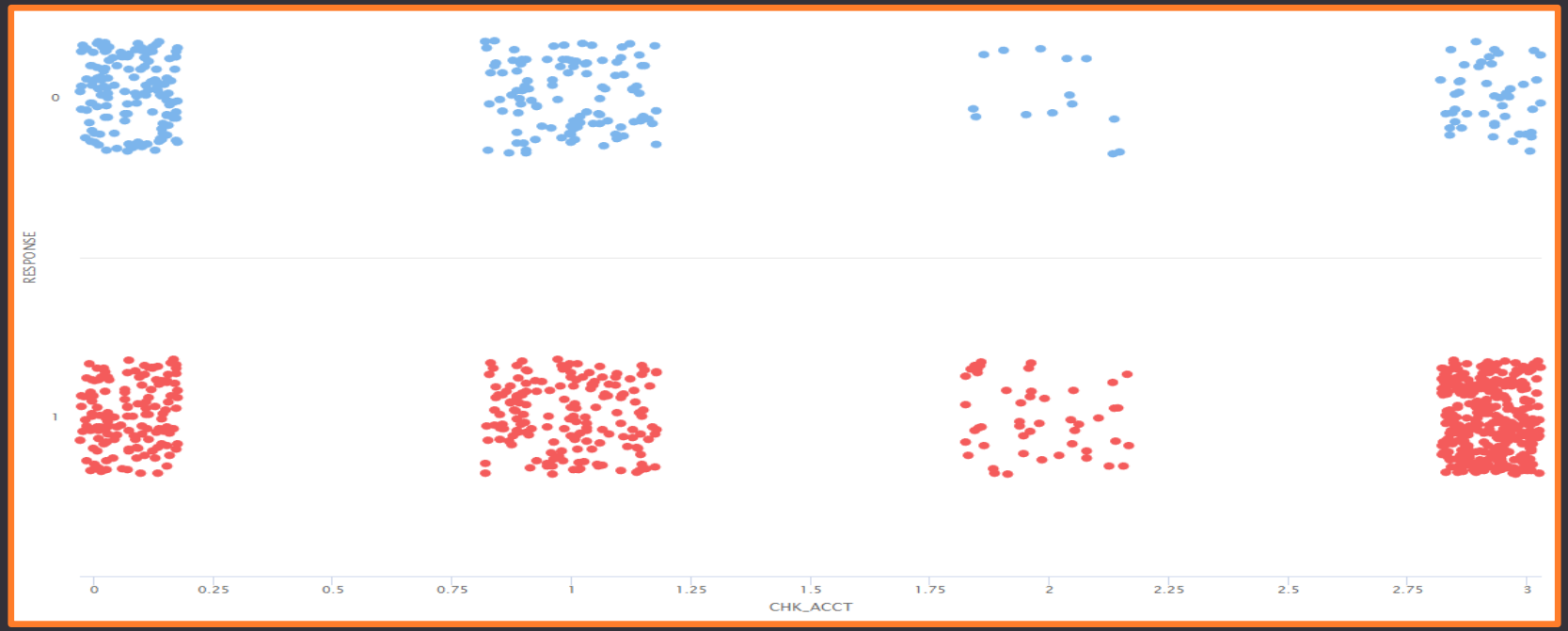
08
German Credit Dataset decision tree



〈decision tree〉

목표 변수 Response는 신용이 좋은지 안 좋은지를 뜻한다.

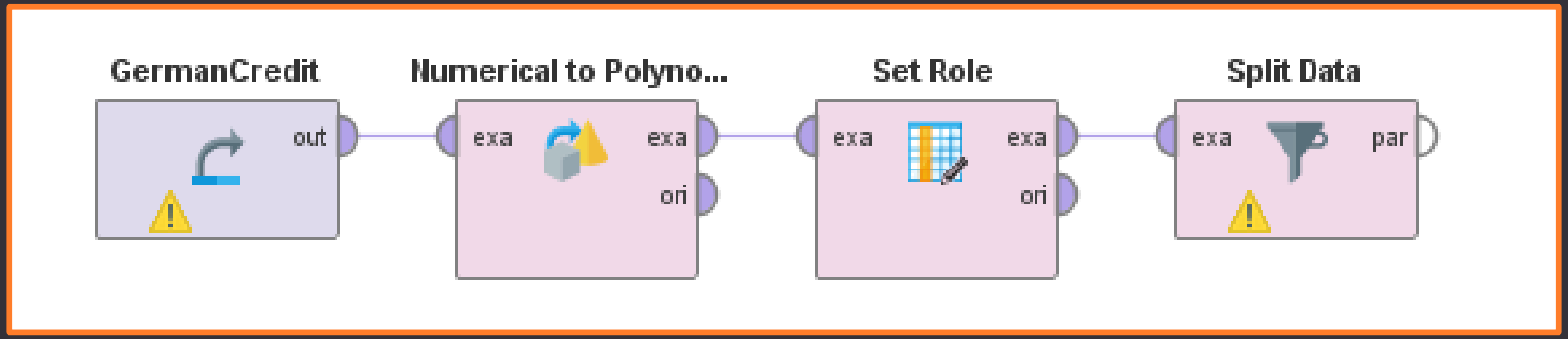
09
예측 변수 분석



〈변수 CHK_ACCT 의 산점도〉

다른 변수와 비교해서 RESPONSE에 가장 큰 영향을 미친다

10
층화 추출

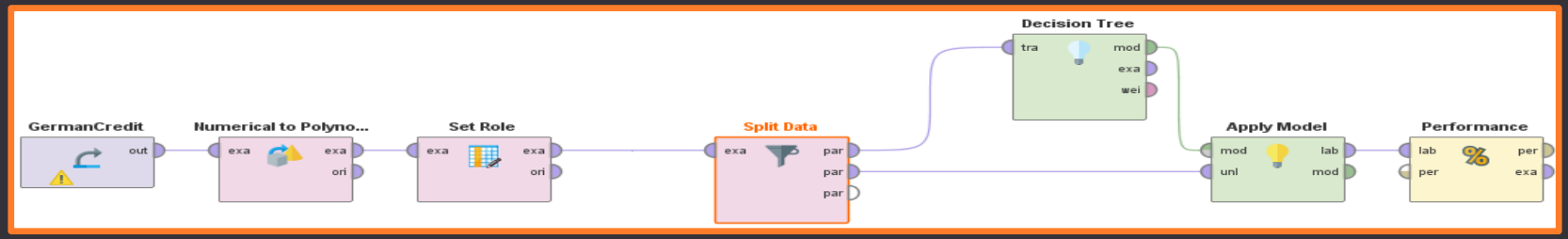


Numerical to Polynominal: 목표 변수 'Response'의 변수 타입을 바꿔주는 오퍼레이터, gini index를 계산하려면 명목형이어야 한다.

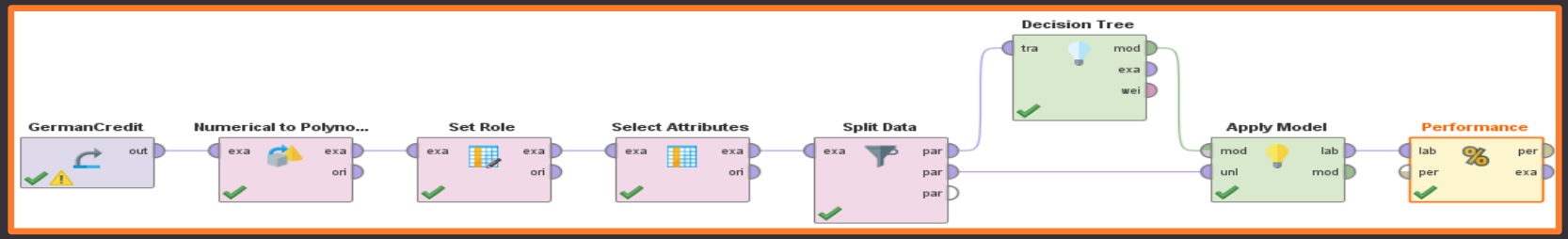
Set Role: 일반 변수인 'Response'를 목표 변수로 지정하는 오퍼레이터, 파라미터에서 label로 설정해주어야 한다.

Split Data: 학습용 데이터셋 800개 시험용 데이터셋 200개로 나누었다. 일관된 정확도를 보고 싶기 때문에 랜덤시드에 체크해야한다.

11
변수 집합 간 정확도 비교



〈모든 변수의 정확도: 72%〉



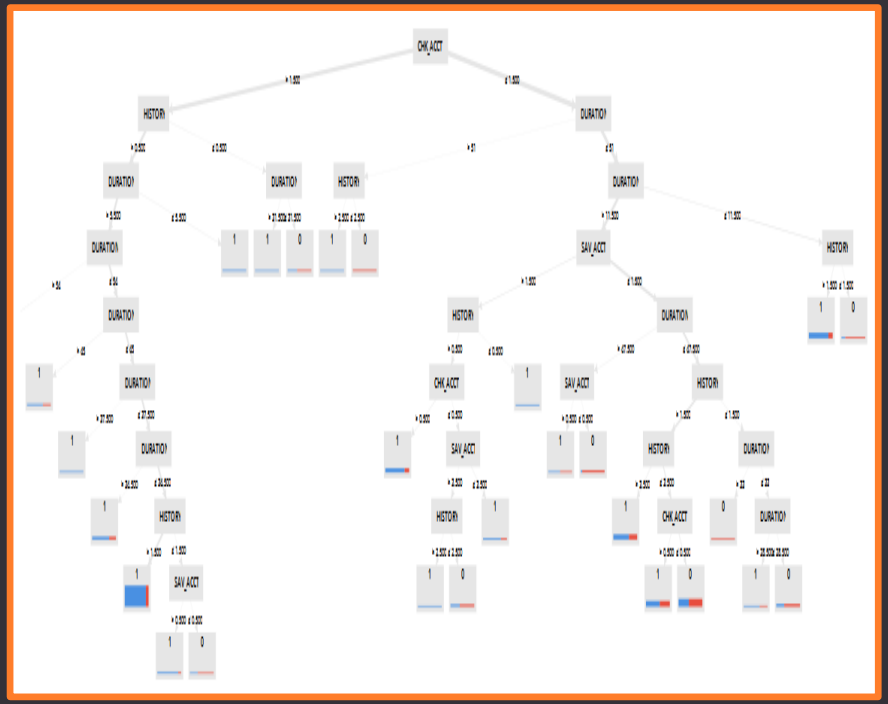
〈선정된 예측 변수 집합의 정확도: 76%〉

CHK_ACCT, DURATION, HISTORY, SAV_ACCT 선정

12
Split criterion과 Pruning에 따른 정확도 분석

gain_ratio	no pruning	75.50%
	pruning	75.50%
	pre pruning	76.00%
information_gain	no pruning	73%
	pruning	73%
	pre pruning	75.50%
gini_index	no pruning	74%
	pruning	74%
	pre pruning	73.50%

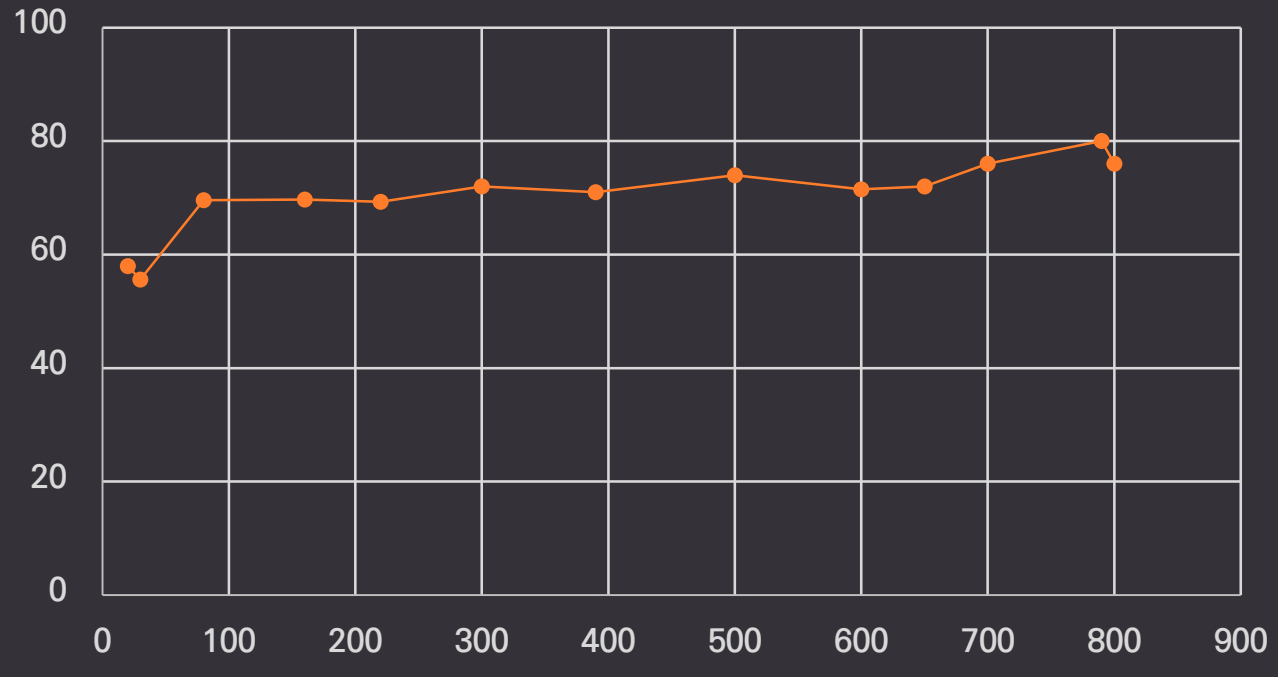
〈각 split criterion, pruning에 따른 정확도〉



〈gain_ratio - pre pruning의 의사결정트리〉

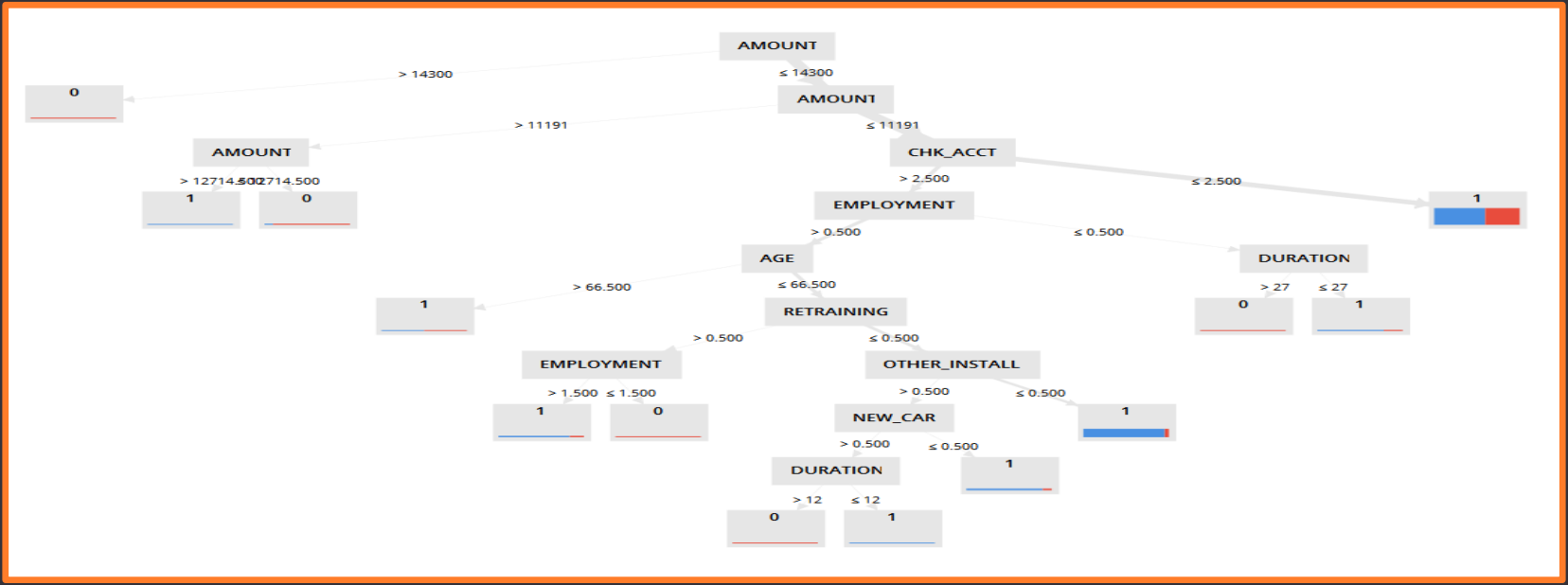
13
Learning curve

X(data size)	Y(accuracy)
20	57.95%
30	55.58%
80	69.58%
160	69.69%
220	69.31%
300	72.00%
390	70.98%
500	74,00%
600	71.50%
650	72.00%
700	76.00%
790	80.00%
800	76.00%



학습용 데이터셋의 크기가 커질수록 정확도가 올라가는 것을 확인 할 수 있다.

14
목표 변수 분포를 바꾼다면?



목표 변수의 분포는 Split Data에서 랜덤 시드 값을 다르게 설정함으로써 바꿔줄 수 있다. 목표 변수의 분포가 바뀐다면 위 사진과 같이 의사결정나무의 모양은 달라지게 된다. 모양이 달라지게 되는 이유는 목표변수의 분포가 바뀐다면, 목표변수와 함께 예측변수들의 개체들도 바뀌게 되는데, 이렇게 된다면 split 방법을 결정짓는 **불순도 척도 값 또한 바뀌기 때문이다.**

15
결 론

이번 decision tree 과제를 마무리하면서 느낀 점은

ⅰ. 래피드마이너의 오퍼레이터를 통해 정말로 간단히 데이터들의 decision tree를 만들기 쉽다는 것

ⅱ. 데이터의 변환이 필요 없이 바로 사용이 가능하다는 것

ⅲ. Performance 오퍼레이터를 통한 정확도 측정과 criterion 파라미터를 사용한 예측 모델 생성 및 규칙 발견이 쉽다는 것이다.

그 밖에 생소한 파라미터들의 매뉴얼을 읽는 것과 criterion 파라미터의 값들을 이해하느라 복습하는데 시간이 걸렸지만

익숙해진다면 매우 빠르고 간편한 예측모델링 알고리즘이라고 생각한다.