

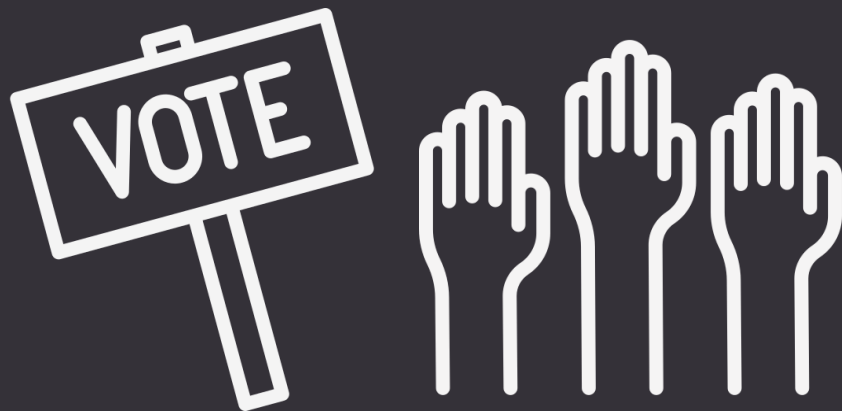
# DS 10주차 수업 팀과제

Ensemble Method

팀 AOA

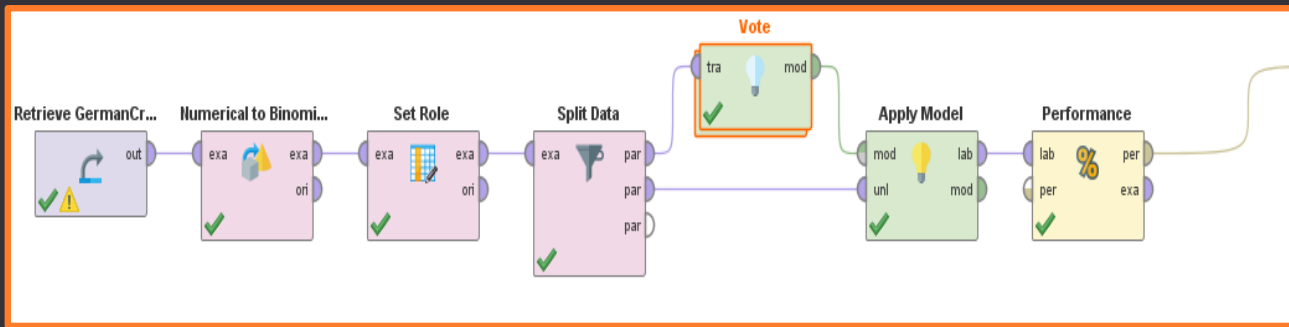
00

# Ensemble



다수의 모델들의 성능을 종합해볼 수 있을까?

# 01 German Credit



## 〈양상블 학습기를 시험하기 위한 process〉

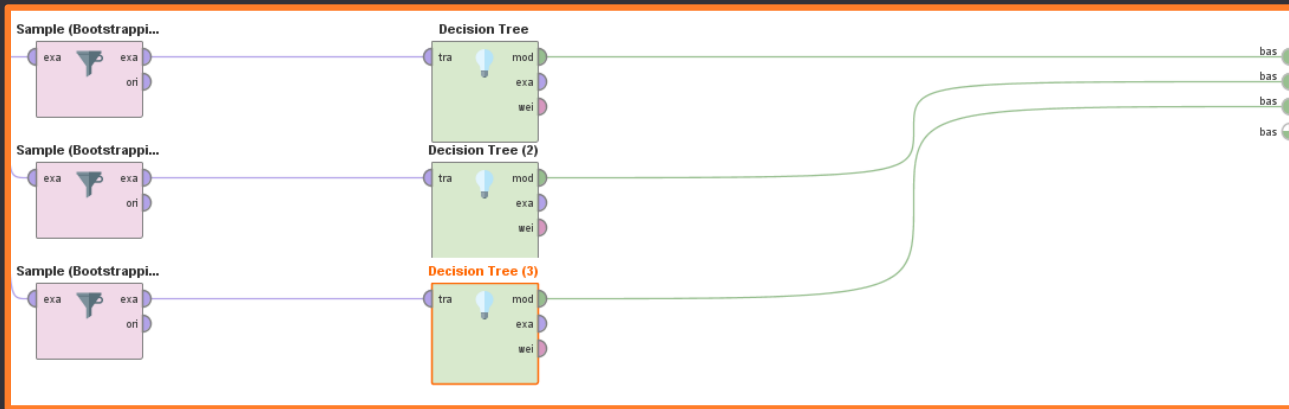
양상블 학습기의 성능을 시험하기 위해 German Credit 데이터셋을 사용하였다.

가장 대표적인 방법인 투표(vote)를 사용하여 출력값을 선택하기 위해 **vote** 오퍼레이터를 사용하였다.

분류 모델이므로 German Credit의 이진명목형 속성인 'Response'가 목표변수가 된다.

## 02

# Bootstrap



〈기본 모델의 학습모델을 bootstrap 방법으로 성장시킨 프로세스〉

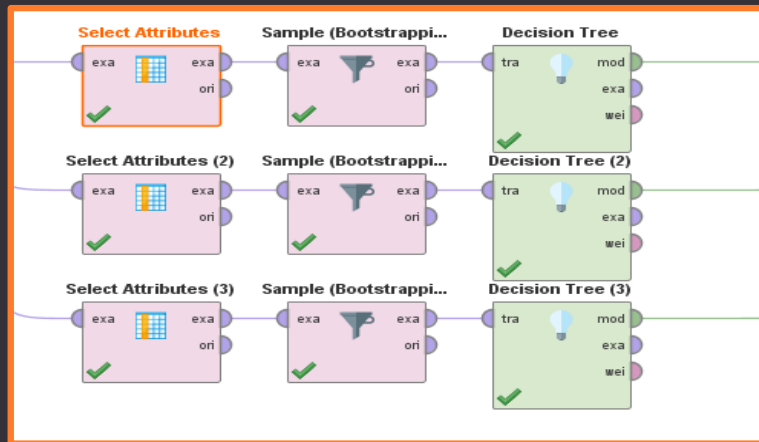
배깅, 즉 학습용 데이터셋을 변환시켜서 각각의 기본 모델을 만드는 방법이다.

복원추출(bootstrapping)을 하기 때문에 각각의 기본 모델들은 중복을 허용하는 서로 다른 데이터셋을 갖는다.

복원 추출의 파라미터는 relative 1.0(원래 크기), 다수결의 결과를 보는 과제이므로 랜덤 시드에 체크하지 않았다.

# 03

## 속성 집합 변경



각 기본 모델에 사용할 속성 집합은 무작위로 적정한 수를 선택한다.

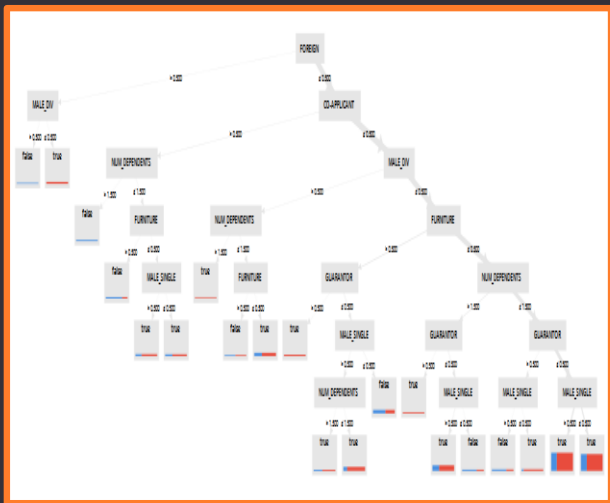
Select attribute 오퍼레이터를 사용하여 각 기본모델의 데이터셋의 속성을 무작위로 선택한다.

이 때 Label 속성인 response는 정상적인 분류를 위해 당연히 모든 모델에 포함되어야한다.

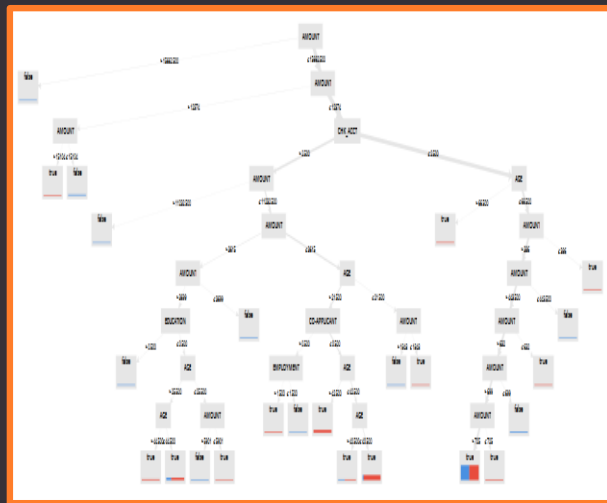
German Credit은 상당히 많은 속성을 포함하므로, 앙상블 학습기의 좋은 조건이 될 수 있는 data set이다.

# 04

## Decision tree



〈기본 모델1의 의사결정나무〉










〈기본 모델2의 의사결정나무〉

기본 모델인 나무 모델들은 가지치기를 하지 않고 (GINI 기준) 완전 나무로 성장시킨다.

이렇게 한다면 과적합을 포함하고 편향이 심해지며, 각 모델의 형태가 다양해진다.

## 05 Aggregation

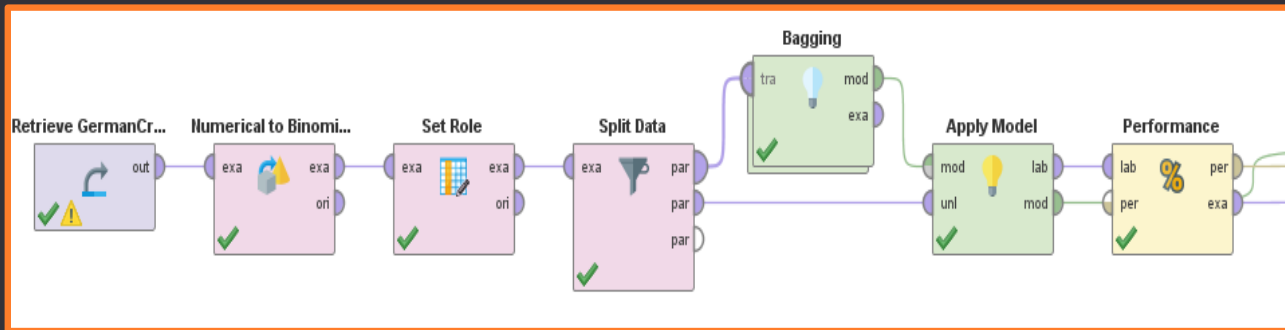
 Vote Model  Model 1  Model 2  Model 3  Stacking Model	 Description	<h3>AttributeBasedVoting</h3> <p>Using the majority of the following attributes for prediction:</p> <pre>base_prediction0 base_prediction1 base_prediction2</pre>
	 Annotations	<p>The default value is true</p>

〈투표에 기초한 앙상블 모델의 출력〉

앙상블의 결과는 기본 모델들의 결과를 다수결의 방법으로 통합한다.

Rapid Miner에서는 Stacking model을 뜻하며, 복원추출부터 통합까지 기법이 **배깅(Bagging)**이다.

# 06 Bagging



## 〈Bagging으로 표현한 process〉

앞에서 설명한 과정은 Bagging 오퍼레이터를 통해 다르게 표현할 수 있다.

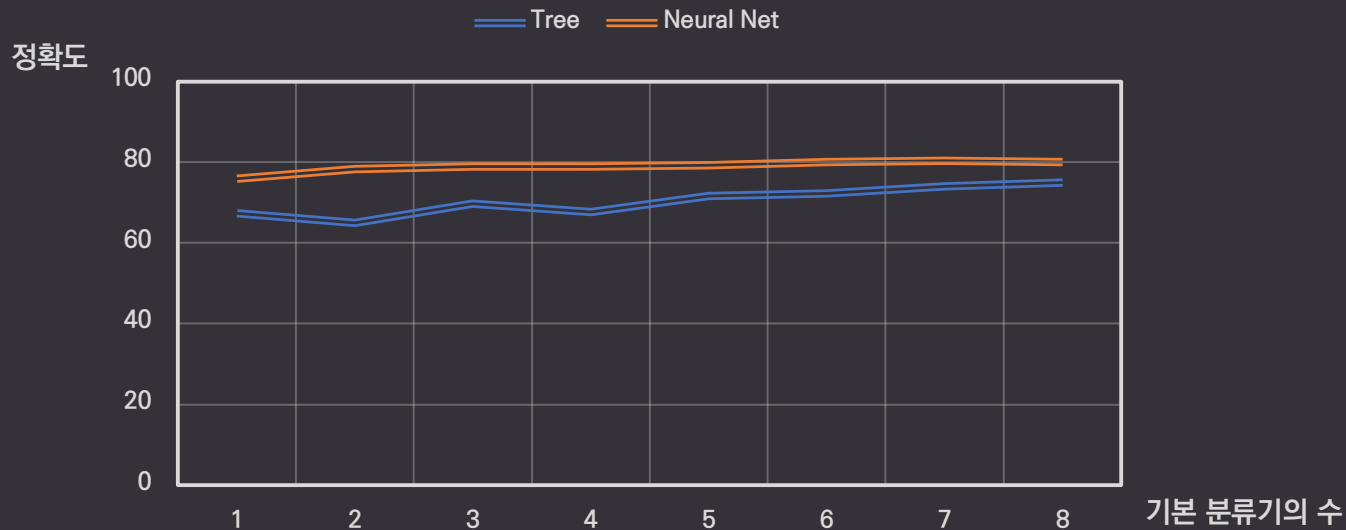
앞 페이지의 Vote 오퍼레이터의 서브 프로세스에 들어간 과정을 합친 것이 Bagging 오퍼레이터이다.

위 오퍼레이터가 다수의 기본 모델 성능을 측정하는데 더 간편하므로 학습 곡선 측정엔 Bagging을 사용한다.



## 07

# Learning curve



기본 분류기의 수와 정확도에 대한 학습곡선을 plot해 보았다.

그래프를 보면 신경망 모델이 정확도는 높지만, 의사결정나무보다 성능향상은 덜하다는 것을 확인할 수 있다.

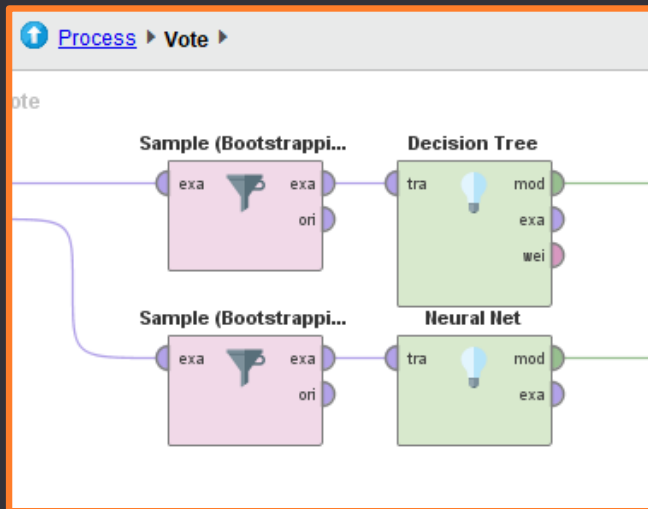
## 08

# 성능 향상의 차이

Q. 나무모델이 경계 기반 분류모델에 비해 성능향상이 좋은 이유?

나무 모델은 루트부터 시작해서 단말 노드로 만들어진다. 중간 노드 하나가 바뀌게 되면 그 아래 부분에는 큰 변화가 생긴다. 따라서 데이터 셋에 약간만 변화를 줘도 모델 전체가 크게 달라진다.(다양성과 독립성 향상)  
그러므로 경계기반 등 다른 모델보다 나무 모델을 기본 모델로 한 앙상블 모델이 더 큰 성능향상을 보인다.

# 나무 모델과 다른 모델을 기반으로 한 앙상블



〈나무 모델과 신경망 모델을 기반으로 한 앙상블〉

다른 두 모델을 기반으로 앙상블에 의한 성능 향상을 측정해 보았다.

측정 결과 기본 모델들의 정확도 평균은 66.33이며, 앙상블 모델의 정확도는 67.67이었다.

두 가지 모델을 섞어도 성능은 향상될 수 있지만, 나무모델 베이스들로만 만들어진 앙상블보다는 성능 향상이 적을 것을 예측 가능하다.

# 10

## 독립성

독립성이란 앙상블 모델을 설계하는데, 필요한 조건을 말한다.

각 기본 모델들은 동일한 학습 데이터셋을 참고하기 때문에, 모델의 다양성과 독립성을 성취하기 어렵다.

그러나 우리는 이번 과제 앞 단계에서 모델의 독립성을 촉진하기 위한 몇 가지 방법을 미리 해 보았다.

**다양한 알고리즘:** 서브 프로세스 안의 모델들의 알고리즘이 다양해진다면, 모델들의 고유 특성이 다르기 때문에 다양한 기본 모델들의 집합을 얻을 수 있을 것이다.

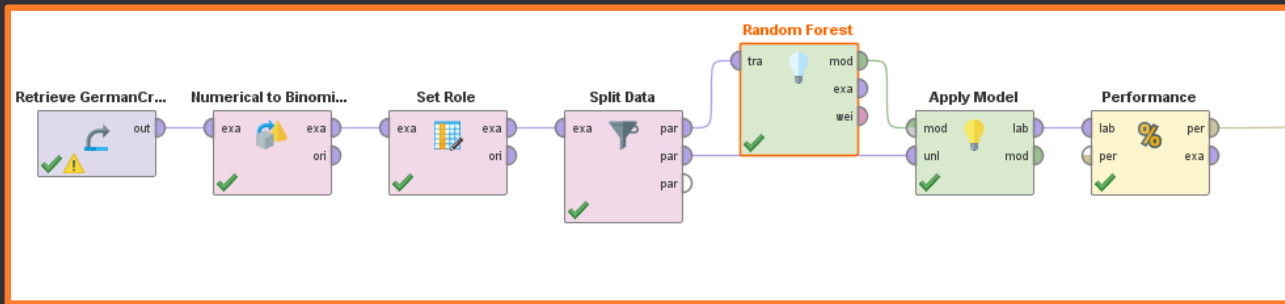
**모델의 파라미터:** 의사결정 나무의 깊이, 신경망 모델의 cycle, k-nn 모델의 k값 등 모델들의 파라미터를 변경함으로써 다양한 모델들을 얻을 수 있다.

**Bootstrapping:** 학습용 데이터셋으로부터 중복을 허용하는 복원추출을 사용한다면 하나의 학습용 데이터셋으로부터 여러 기본모델들의 데이터셋을 구축할 수 있다.

**속성 집합 변경:** 각 기본 모델마다 속성 집합을 무작위로 적정한 수를 선택해서 모델들의 데이터셋을 다양화 할 수 있다.

## 11

## Random Forest



〈Random Forest process〉

Random Forest 기법에서는 오퍼레이터 내부에 의사결정나무가 포함되어 있다.

배깅과 유사하게 학습 데이터셋을 복원추출하며, 랜덤으로 선택한 속성만 고려하여 노드에서 최적 분리를 찾는다.

# 12

## Parameter

number of trees	100	①
criterion	gain_ratio	①
maximal depth	10	①
<input type="checkbox"/> apply pruning		①
<input type="checkbox"/> apply prepruning		①
<input type="checkbox"/> random splits		①
<input checked="" type="checkbox"/> guess subset ratio		①
voting strategy	confidence vote	①
<input type="checkbox"/> use local random seed		①
<input checked="" type="checkbox"/> enable parallel execution		①

Number of trees: 생성할 나무의 수를 결정한다. 각 나무들은 복원추출된 상태로 생성되기 때문에, **모델 간 독립성에 영향을 미친다.**

Criterion, maximal depth: 생성할 나무의 기준, 깊이 등을 결정한다.

Guess subset ratio: 각 모델에 선택될 속성의 비율을 정한다. 입력된 임의값의 비율을 정하고 나무를 분리한다. **모델 간 독립성에 영향을 미친다.**

Voting strategy: 투표 방식을 정할 수 있다.  
(confidence= 정확도 기준, majority=다수결 기준)

Enable parallel execution: 메모리와 관계, 모델 생성의 병렬 처리.