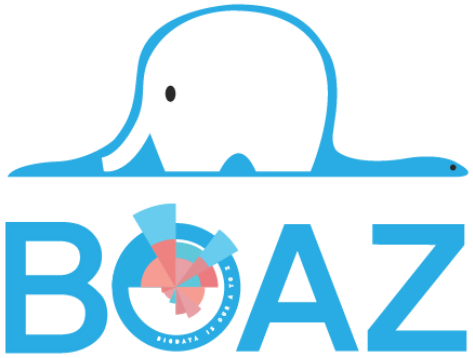
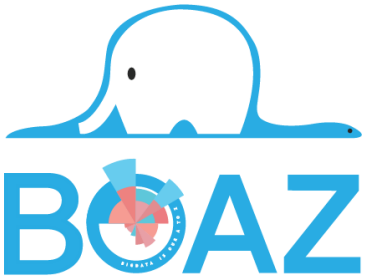


2016.11.17 Base session

BOAZ - D조

Improving Model Performance

< 발표: 정지원, 이다영, 곽현빈 김대규>



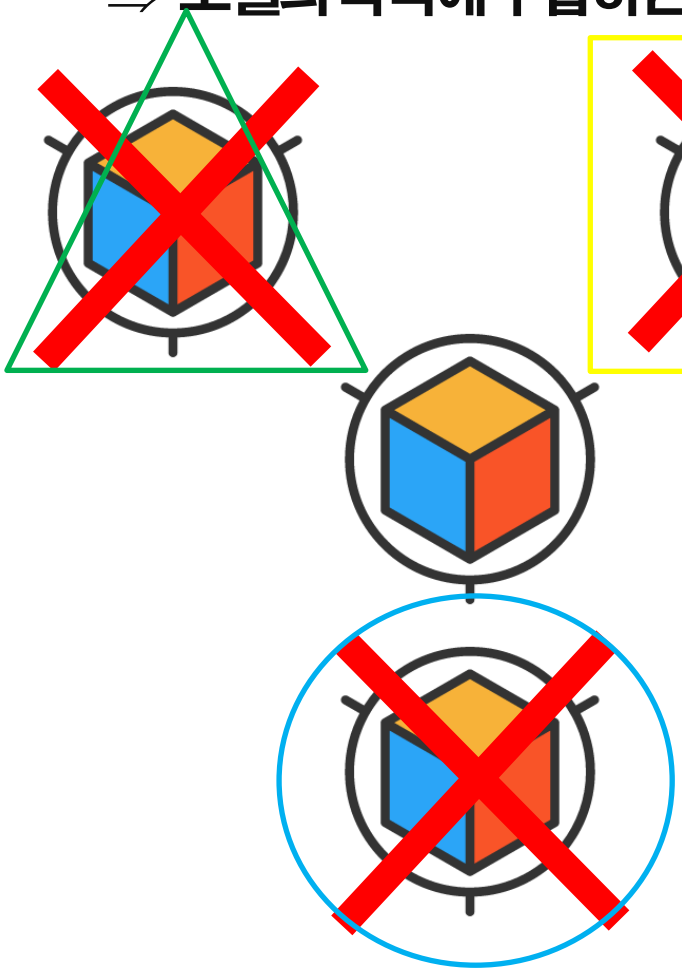
목차

Improving Model Performance

앙상블 기법
① Bagging : bootstrap aggregating
② Boosting
③ Random forest

Improving model performance

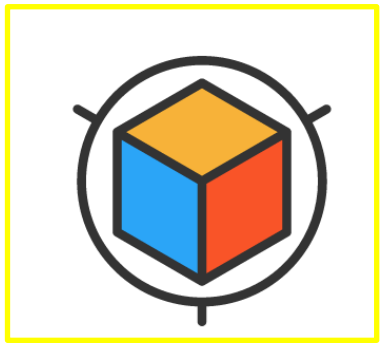
최상의 성능을 만족하려면?
⇒ 모델의 목적에 부합하는가



다른 모델이 고려 대상이 되지 않음
한 특정작업에 최적화된 모델
다른 분야에 성능하락 및 어려운 배치

Improving model performance

모델 성능 향상의 의미



각 특정작업에 대한 모델 찾기

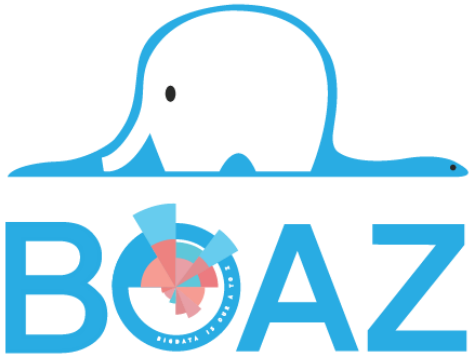
반복, 정제, 학습 알고리즘을 혼합하는 방법

Improving model performance

선수의 포지션 및 역할
선수의 특징
선수의 장,단점
모두 다르다!

마치 코치처럼
많은 전략을 통계 학습기의 성능을
향상하는데 기계 알고리즘을 사용

훈련 기술과 최대한 성능을
발휘할 수 있게 팀워크를 조합



앙상블

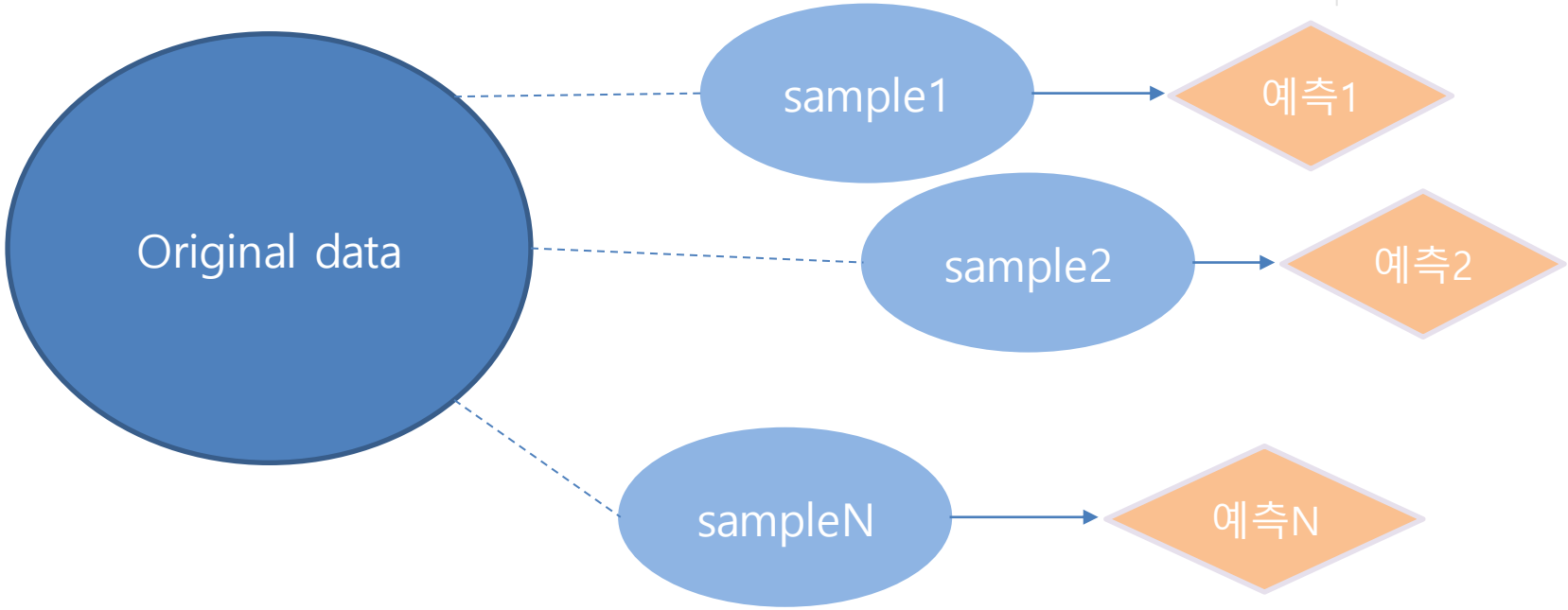
메타학습모델 : 다수 모델의 예측을 관리하고 조합하는 기술

앙상블

데이터 집합에 대해 성능이 가장 좋은 학습 모델을 하나만 고르는 것이 아니라, 여러 가지 다양한 모델들을 모두 이용해서 얻은 결과를 조합하는 방식

Ex> Regression Tree, Simple Regression, Classification Tree…
이들 중 하나만 선택하여 진행하였던 기존의 예측 방식.

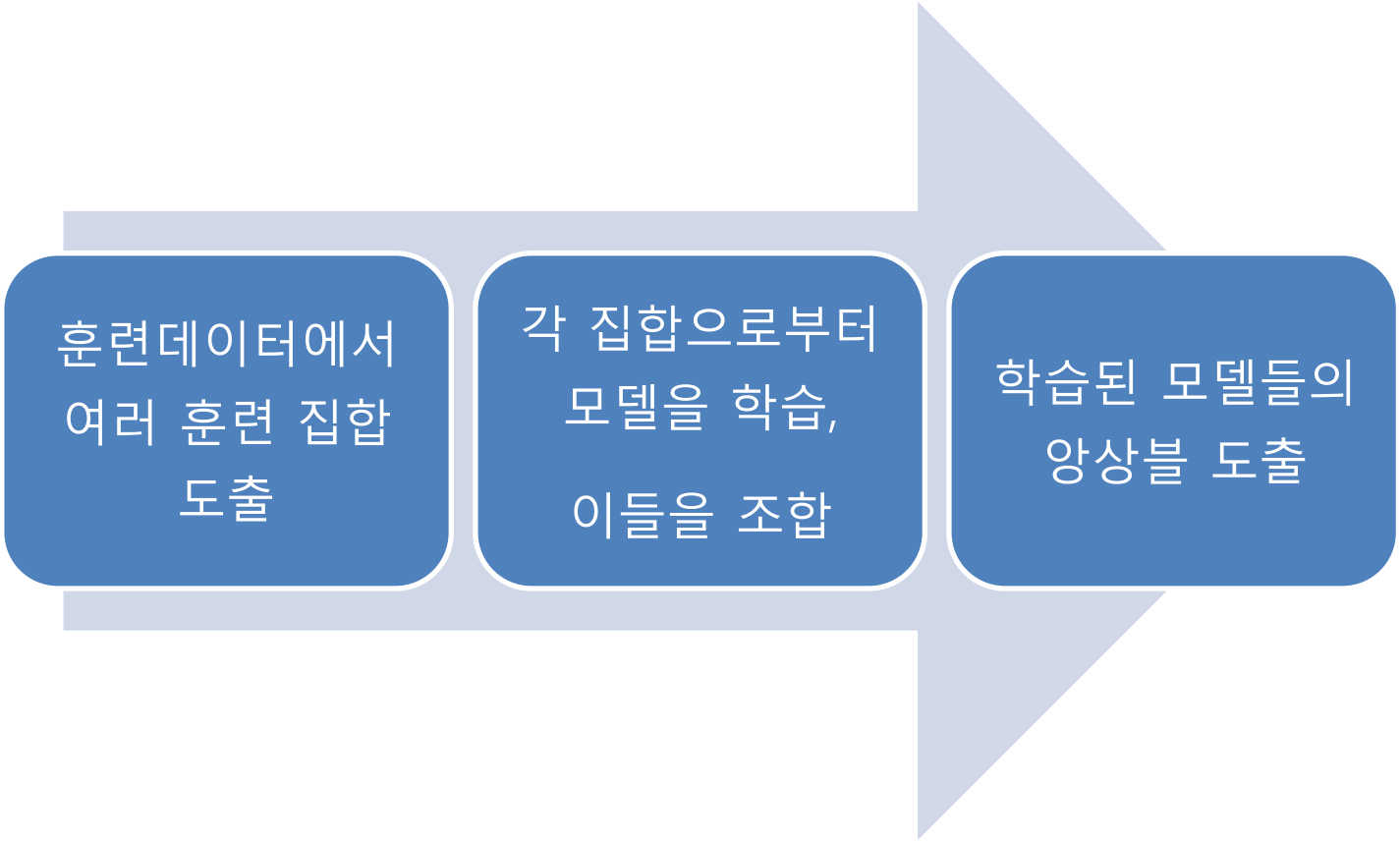
앙상블



Original train data 에서 Sample train data를 여러 번 뽑은 후 여러 모델을 각각 적용, 조합해 최종 앙상블 예측치를 구함.

장점> 단일 모델보다 훨씬 더 좋은 성능을 보일 수 있다.
단점> 직관적이지 않다. 해석하기 어렵다.

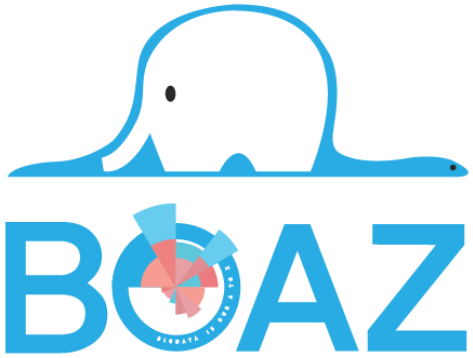
앙상블



조작

훈련 집합 조작	입력 특징 조작
• Bagging • boosting	• Random forest

이 밖에도 클래스 레이블 조작, 학습 알고리즘 조작 등이 있음.
본 세션에서 다루는 세 가지 앙상블 기법은
이와 같은 단계별 조작법에 의해 만들어 짐



Bagging이란?

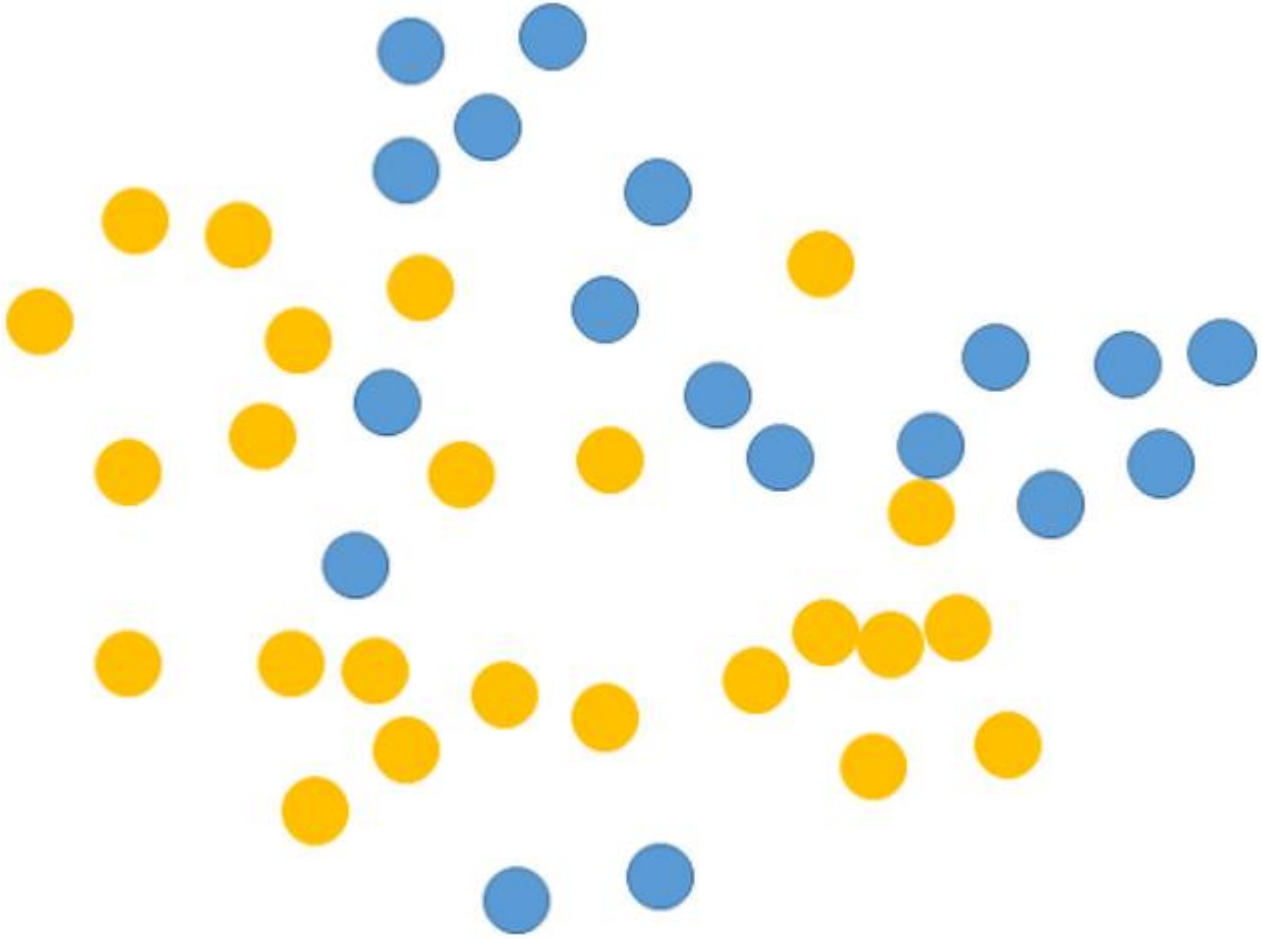
Bagging

- Bootstrap aggregating의 준말

1) original 데이터에서 여러 개의 Bootstrap 데이터 생성
   이 때 Bootstrap 데이터는 동일한 크기, 랜덤복원추출한 표본임
2) 각 Bootstrap data를 모델링
3) 결합(목표변수가 연속형일 때는 평균, 범주형일 때는 투표로 결합)
4) 최종의 예측 모형을 산출

언제 사용할까?
- 단일 모형으로 사용했을 때 예측 모형의 변동성이 클 때
  예측모형의 변동성을 감소시키기 위해 사용
- 여러 번 복원 샘플링하면 예측 모형의 분산이 감소하는 특징 이용

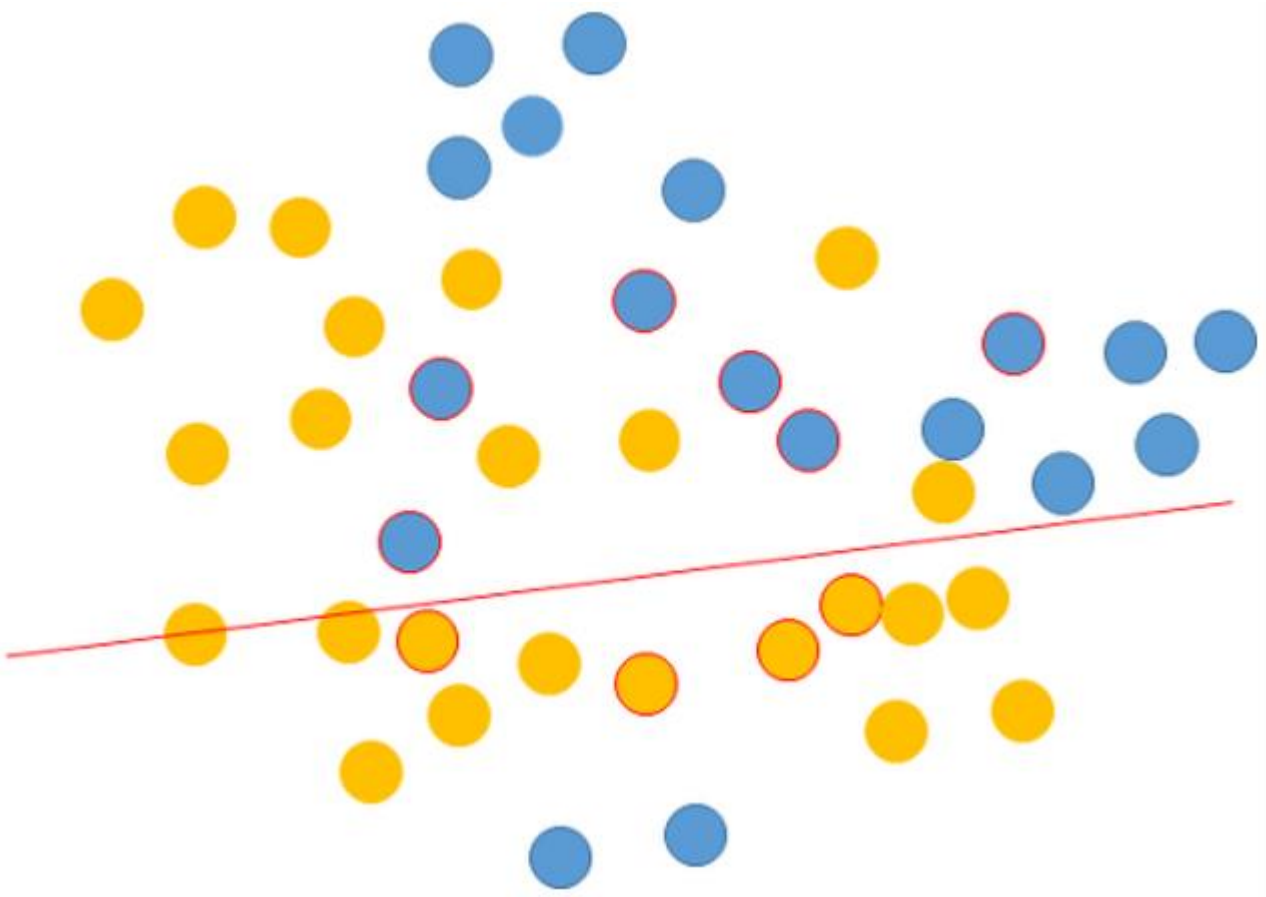
Bagging



전체 data가 아닌 Sampling된 data로 classification

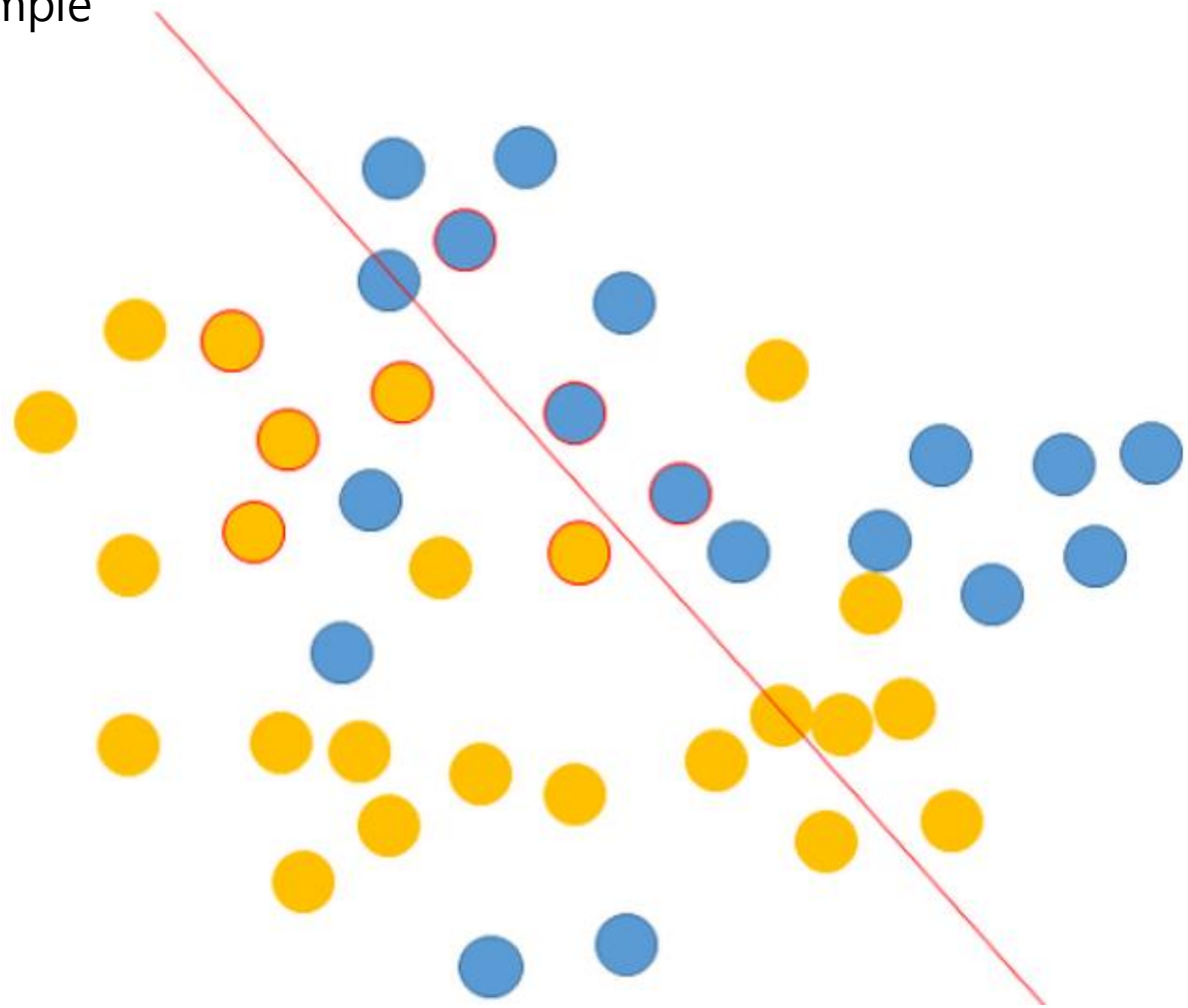
Bagging

#1 sample



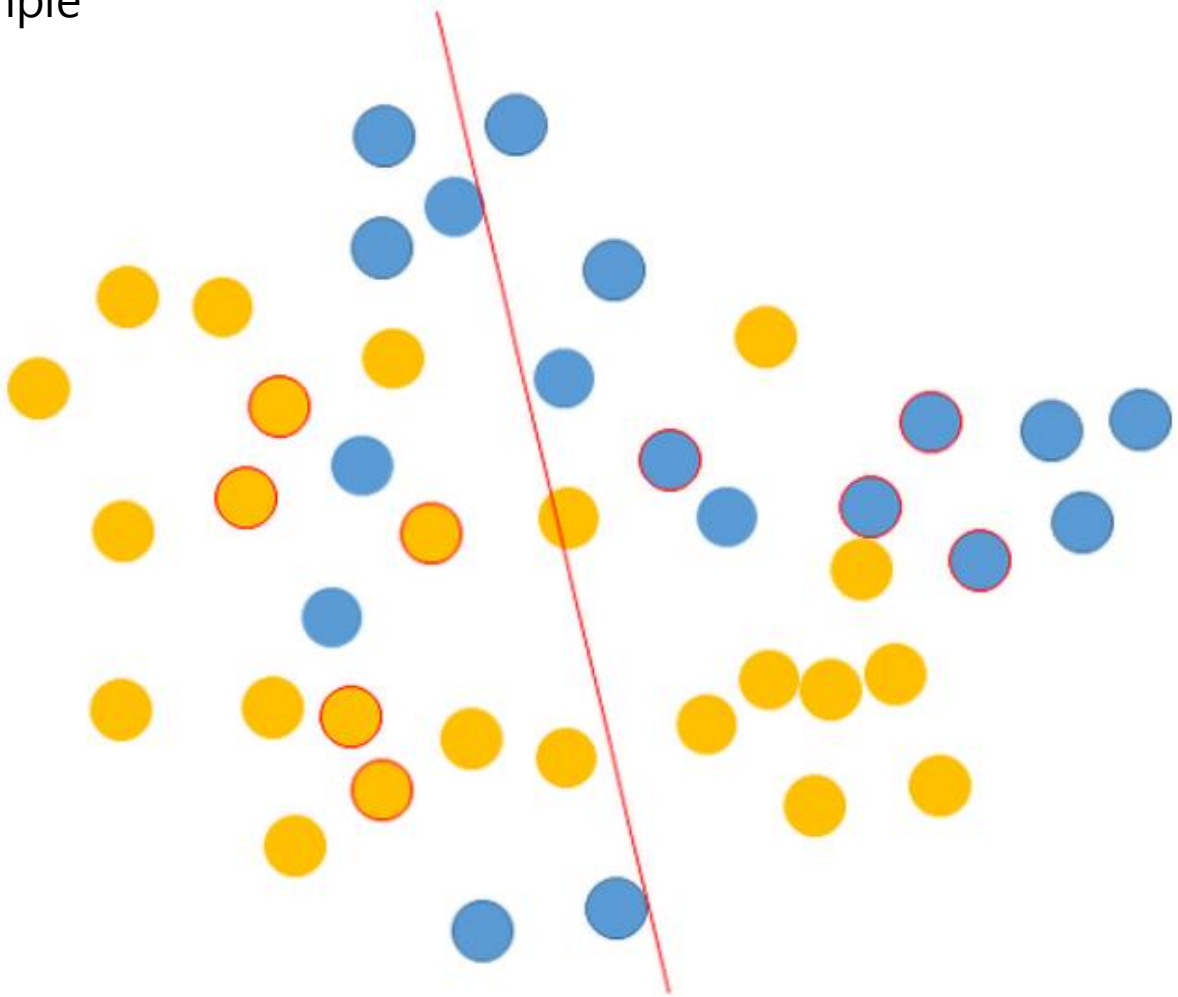
Bagging

#2 sample

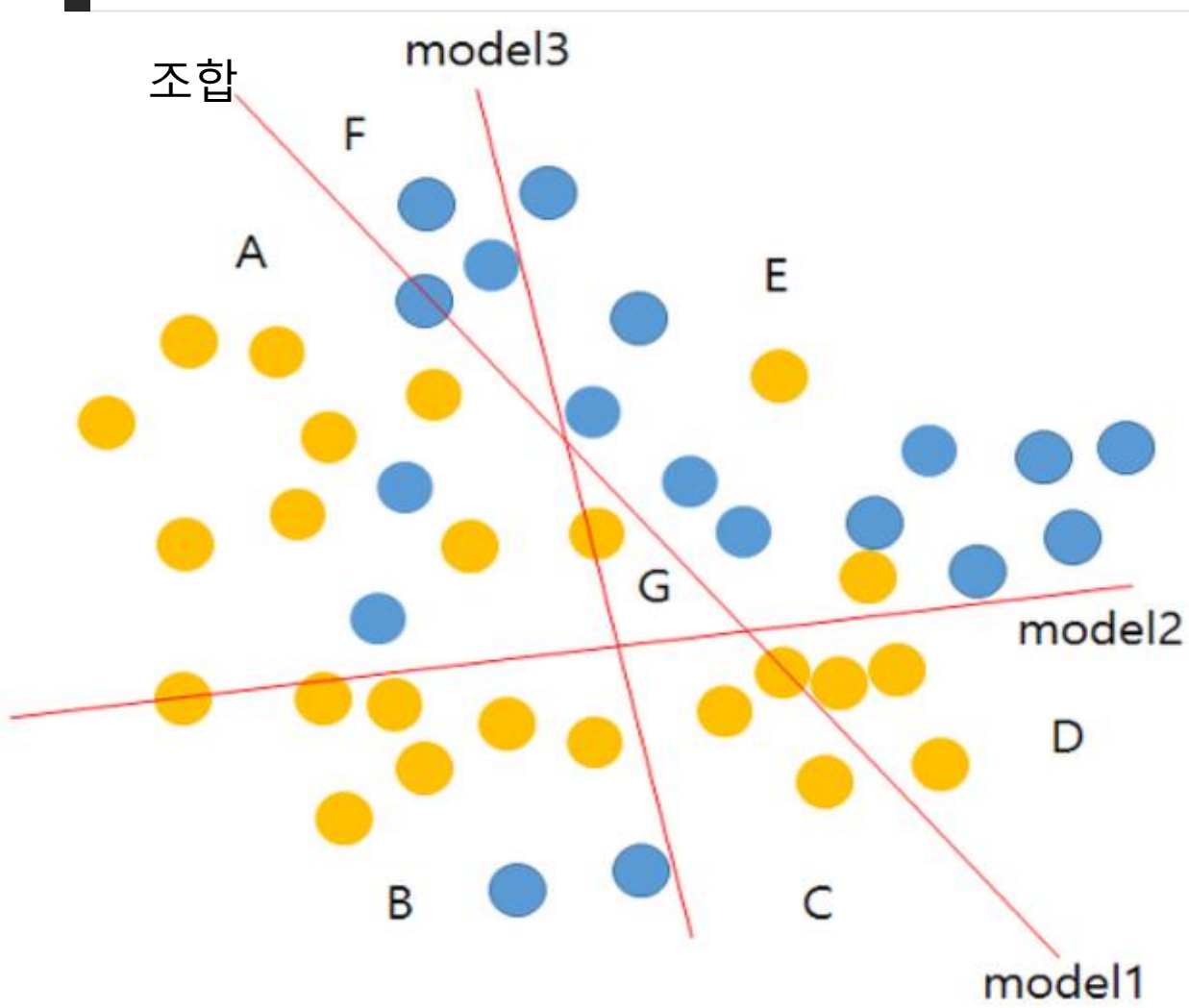


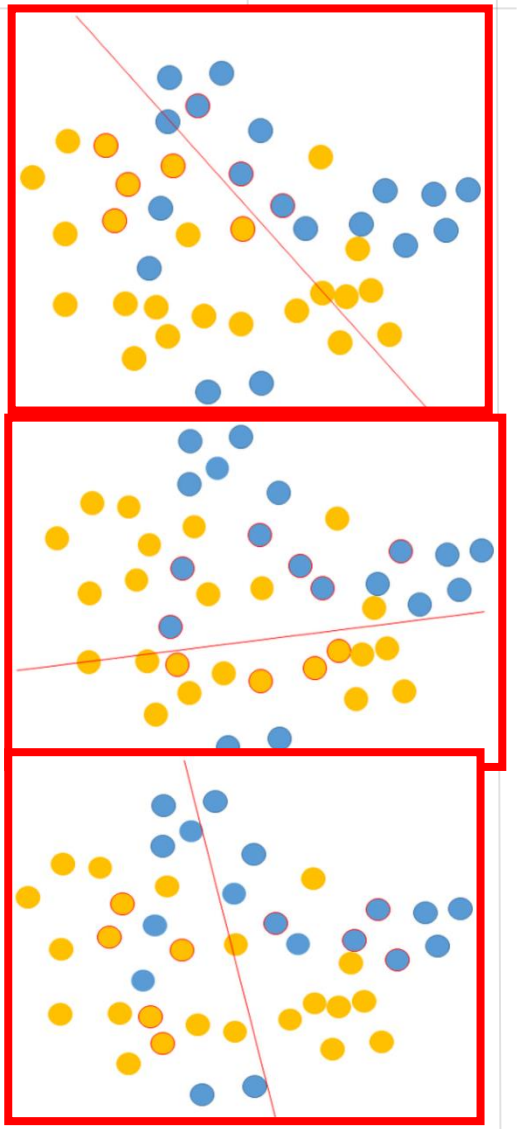
Bagging

#3 sample



Bagging





3개의 sample로 구역을 나누고 모델을 정한다!

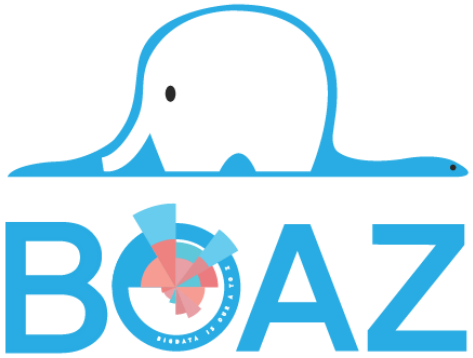
Bagging

조합

	A	B	C	D	E	F	G
model1	Yellow	Yellow	Yellow	Blue	Blue	Blue	Yellow
model2	Blue	Yellow	Yellow	Yellow	Blue	Blue	Blue
model3	Yellow	Yellow	Blue	Blue	Blue	Yellow	Blue
Vote	Yellow win	Yellow win	Yellow win	Blue win	Blue win	Blue win	Blue win

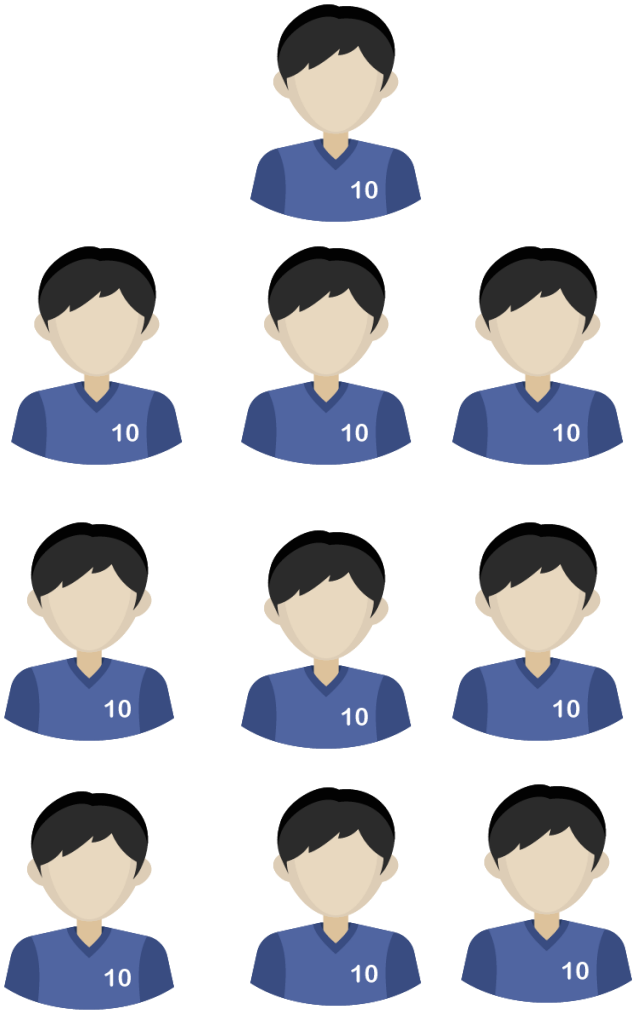
에러율

	model1	model2	model3	ensemble
error 개수	9	13	15	9



Boosting이란?

Boosting



정말정말 쉬운 weak learner 에러구하기

$$\text{error(t)} = \sum_{i=1}^{n} w(t)_i I(h_t(x_i)! = y_i)$$

where n is the number of data and t is the index of weak learners
where $h_t(x_i)$ is the predicted value of x_i with tth weak learner
where I(A) is 1 if A is true, and 0 if A is false

Boosting

예제로 알아봅시다…

x	0	1	2	3	4	5	6	7	8	9
y	1	1	1	-1	-1	-1	1	1	1	-1
d1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
h1(x<thr), thr = 2.5	1	1	1	-1	-1	-1	-1	-1	-1	-1
I(h1(x) != y)	0	0	0	0	0	0	1	1	1	0
e1 = sum(d1*I(h1(x) != y))	0.3									

X = weak learner
Y = d1
D1 = Weight 각각 10개니까 0.1
Thresold 값은 error rate를 제일 작게하는 thresold값을 선택하는 것
0.5부터 8.5까지 0.5단위로 할 것
h1(x<thr)값 = x<thr라는 조건이 참이면 1 거짓이면 -1이다. 단순하다.
그래서 Weak learner이다.

Boosting

x	0	1	2	3	4	5	6	7	8	9
y	1	1	1	-1	-1	-1	1	1	1	-1
d1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
h1(x<thr), thr = 2.5	1	1	1	-1	-1	-1	-1	-1	-1	-1
I(h1(x) != y)	0	0	0	0	0	0	1	1	1	0
e1 = sum(d1*I(h1(x) != y))	0.3									
a1	0.42364893									
z1 = sum(d1*exp(-1*a1*y*h1(x)))	0.916515139									
d1*I(h1(x) != y)	0	0	0	0	0	0	0.1	0.1	0.1	0
d1*exp(-1*a1*y*h1(x))	0.065465	0.065465	0.065465	0.065465	0.065465	0.065465	0.152753	0.152753	0.152753	0.065465
d2	0.071429	0.071429	0.071429	0.071429	0.071429	0.071429	0.166667	0.166667	0.166667	0.071429
h2(x<thr), thr = 8.5	1	1	1	1	1	1	1	1	1	-1
I(h2(x) != y)	0	0	0	1	1	1	0	0	0	0
e2 = sum(d2*I(h2(x) != y))	0.214285714									
a2	0.649641492									
z2 = sum(d2*exp(-1*a2*y*h2(x)))	0.820651807									
d2*I(h2(x) != y)	0	0	0	0.071429	0.071429	0.071429	0	0	0	0
d2*exp(-1*a2*y*h2(x))	0.037302	0.037302	0.037302	0.136775	0.136775	0.136775	0.087039	0.087039	0.087039	0.037302
d3	0.045455	0.045455	0.045455	0.166667	0.166667	0.166667	0.106061	0.106061	0.106061	0.045455
h3(x>thr), thr = 5.5	-1	-1	-1	-1	-1	-1	1	1	1	1
I(h3(x) != y)	1	1	1	0	0	0	0	0	0	1
e3 = sum(d3*I(h3(x) != y))	0.181818182									
a3	0.752038698									
z3 = sum(d3*exp(-1*a3*y*h3(x)))	0.771389216									
d3*I(h3(x) != y)	0.045455	0.045455	0.045455	0	0	0	0	0	0	0.045455
d3*exp(-1*a3*y*h3(x))	0.096424	0.096424	0.096424	0.078567	0.078567	0.078567	0.049997	0.049997	0.049997	0.096424

$$\mathrm{a} = \frac{w_{old}}{2e} = \frac{0.1}{2\times 0.3} = 0.16667$$

$$\mathrm{b} = \frac{1}{2(1-e)} = \frac{1}{2(1-0.3)} = 0.07143$$

$$a(t) = \frac{1}{2}\log\frac{1-e_t}{e_t}$$

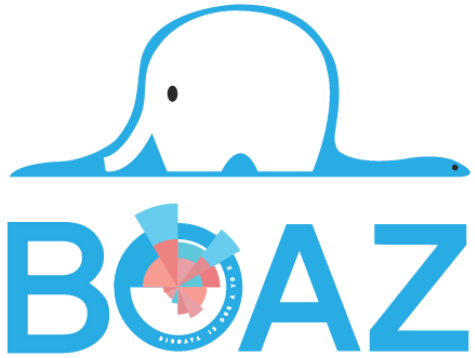
Boosting

x	0	1	2	3	4	5	6	7	8	9
y	1	1	1	-1	-1	-1	1	1	1	-1
h1(x<thr), thr = 2.5	1	1	1	-1	-1	-1	-1	-1	-1	-1
a1	0.42364893									
a1*h1(x<thr)	0.423649	0.423649	0.423649	-0.42365	-0.42365	-0.42365	-0.42365	-0.42365	-0.42365	-0.42365
H(x) = sign(a1h1)	1	1	1	-1	-1	-1	-1	-1	-1	-1
Correct	y	y	y	y	y	y	n	n	n	y
h2(x<thr), thr = 8.5	1	1	1	1	1	1	1	1	1	-1
a2	0.649641492									
a2*h2(x<thr)	0.649641	0.649641	0.649641	0.649641	0.649641	0.649641	0.649641	0.649641	0.649641	-0.64964
H(x) = sign(a1h1+a2h2)	1	1	1	1	1	1	1	1	1	-1
Correct	y	y	y	n	n	n	y	y	y	y
h3(x>thr), thr = 5.5	-1	-1	-1	-1	-1	-1	1	1	1	1
a3	0.752038698									
a3*h3(x<thr)	-0.75204	-0.75204	-0.75204	-0.75204	-0.75204	-0.75204	0.752039	0.752039	0.752039	0.752039
H(x) = sign(a1h1+a2h2 + a3h3)	1	1	1	-1	-1	-1	1	1	1	-1
Correct	y	y	y	y	y	y	y	y	y	y

$$\mathrm{a} = \frac{w_{old}}{2e} = \frac{0.1}{2\times 0.3} = 0.16667$$

$$\mathrm{b} = \frac{1}{2(1-e)} = \frac{1}{2(1-0.3)} = 0.07143$$

$$a(t) = \frac{1}{2}\log\frac{1-e_t}{e_t}$$



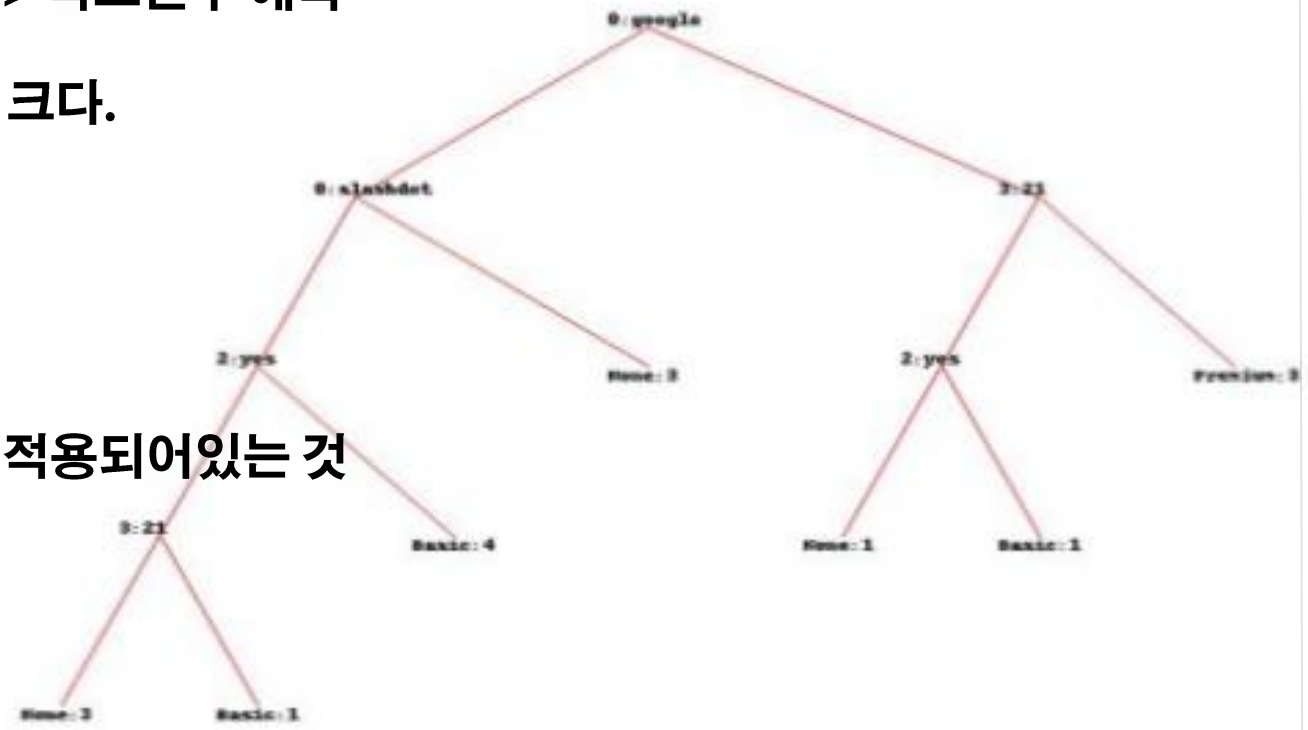
RandomForest란?

랜덤포레스트

의사결정나무

하나의 데이터 집합 -> 한번의 훈련용 데이터 생성
-> 하나의 트리 생성 -> 목표변수 예측

단점> 과적합 위험이 크다.



Tree에 특별히 잘 적용되어있는 것

랜덤포레스트 (배깅방법)

하나의 데이터 집합 ->랜덤복원샘플링으로 여러 개의 훈련용 데이터 생성
-> 여러 개 트리 생성 -> 다수결/평균/확률 등으로 목표변수 예측

장점> 트리들의 편향은 유지되면서 분산 감소