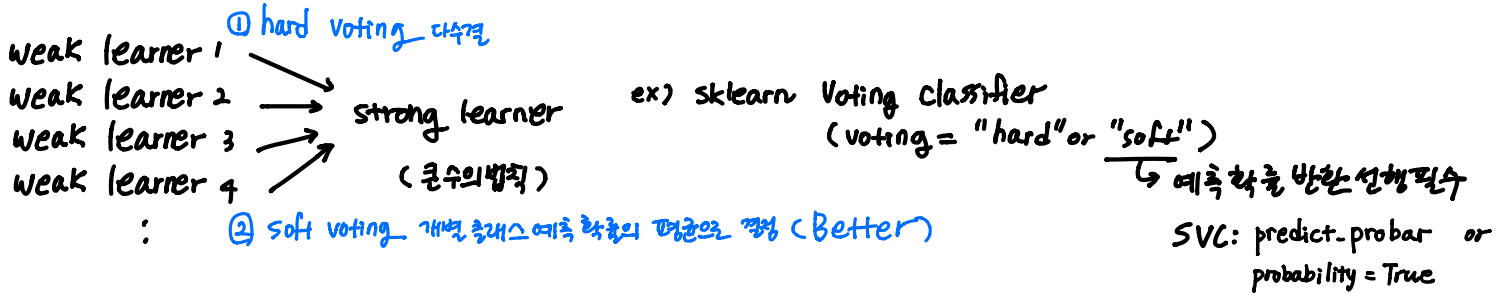
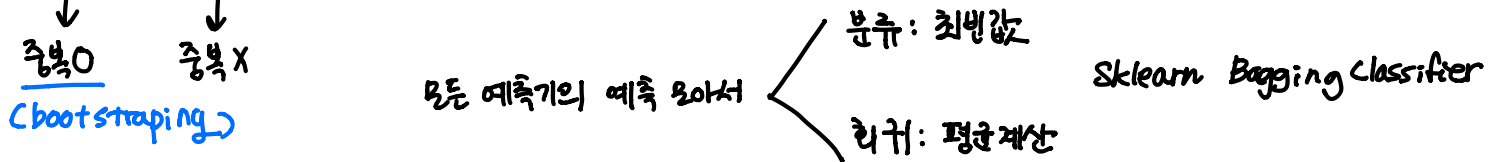


# 해즈온 ML CH7 앙상블 & 랜덤포레스트

## 7.1 투표기반 분류기



## 7.2 배깅 & 페이스팅 : 같은 알고리즘, 훈련세트 무작위 변경 수행 가능, 분산 ↓ 편향 ↓



배깅: m개 샘플 중복하여 random 선택

약 63%만 샘플링 됨 → 나머지 37% : "oob 샘플"

테스트 데이터로 이용

각 예측기의 oob 평가 평균으로 앙상블 평가

oob\_decision\_function에 예측 확률 저장됨

## 7.3 랜덤 패치와 랜덤 서브스페이스 분산 ↓ 편향 ↑

특성 샘플링 : 무작위로 특성을 선택  
bootstrap\_features = True  
max\_features < 1

이미지와 같은 2차원 dataset에서 유용

① 특성, 샘플 모두 샘플링하는 것 → 랜덤 패치 방식

② 특성만 샘플링하는 것 → 랜덤 서브스페이스

## 7.4 랜덤포레스트

배깅 (or 페이스팅) 적용한 결정트리 앙상블 (RandomForestClassifier, RandomForestRegressor)  
(Bagging Classifier + Decision Tree Classifier)

\* 익스트림 랜덤 트리 (엑스트라 트리) 앙상블 분산 ↓ 편향 ↑

: 트리에서 매 노드마다 최적의 임계값 찾는 대신

무작위로 분할한 뒤 최상의 분할 선택.

ExtraTreesClassifier

\* 특성 중요도 측정

각 노드에서 얼마나 불순도가 감소되었는가? (= 가중치 평균 = 연산된 훈련 샘플 수)

feature\_importances\_ 에 저장됨

## 7.5 부스팅 boosting □ → □ → □

① AdaBoost : 이전 모델에서 라소적합된 샘플의 가중치 높여가며... / 연속적 알고리즘. 병렬 실행 불가

$$w^{(i)} = \begin{cases} w^{(i)} & \hat{y}_j^{(i)} = y^{(i)} \\ w^{(i)} \exp(\alpha_j) & \hat{y}_j^{(i)} \neq y^{(i)} \end{cases}$$

$$\hat{y}(x) = \underset{k}{\operatorname{argmax}} \sum_{j=1}^N \alpha_j \quad \begin{matrix} N: \text{예측기 수} \\ \rightarrow \text{가중치 합이 가장 큰 클래스가 예측 결과가 됨} \end{matrix}$$

sklearn AdaBoostClassifier

② Gradient Boosting : 이전 예측기가 만든 잔여 오차 (residual error)에 새로운 예측기 학습  
 $y_2 = y - \text{predict}$

sklearn GradientBoostingRegressor (learning\_rate)

+ SGB

XGBoost 이해하기..

낮게(0.1) 설정하면 트리가 많이 필요하나 성능 ↑

∴ 축소 규제

7.6 스태킹 : 블렌더 or 메타학습기 생성

