

트리 구조를 기반으로 데이터에서 규칙 학습하는 알고리즘 명:

- 해결 가능한 문제: ,
- 복잡한 데이터에서도 () 동작

결정트리 첫 기준:

결정트리 중간 기준:

마지막 기준:

결정트리는 다른 알고리즘에 비하여 전처리가 많이 필요하다 O, X

결정트리에서 과대적합 방지를 위하여:

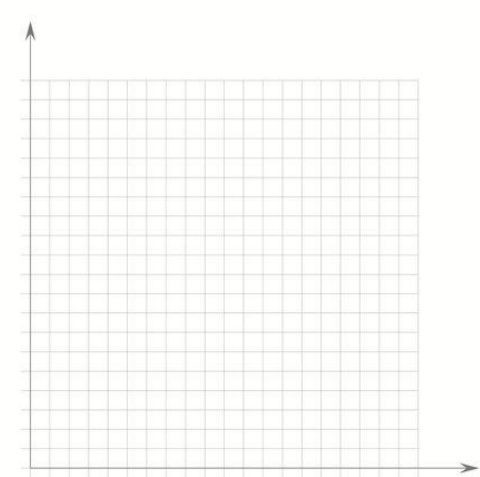
Parameter	Description
<div></div>	트리의 최대 깊이 제한, 감소시킬수록 과대적합의 위험이 줄어든다.
<div></div>	각 노드에서 분할에 사용할 features 개수
<div></div>	분할되기 위한 최소 샘플 수 조건
<div></div>	리프 노드가 가져야할 최소 샘플 수
<div></div>	min_sample_leaf 와 동일 (가중치가 부여된 전체 샘플에서의 비율)
<div></div>	리프 노드의 최대 수
<div></div>	분할로 얻어질 최소한의 불순도 감소량

지니 불순도에서 완벽한 분류:

지니 불순도에서 완벽한 랜덤:

엔트로피에서 완벽한 랜덤:

지니 불순도 및 엔트로피 그래프:



결정트리 단점 2개:

화이트박스 모델 예:

블랙박스 모델 예:

```
tree_clf1 = DecisionTreeClassifier(random_state=42)
tree_clf2 = DecisionTreeClassifier( , random_state=42)
tree_clf1.fit(X_moons, y_moons)
tree_clf2.fit(X_moons, y_moons)

tree_reg = DecisionTreeRegressor( =2, random_state=42)
tree_reg.fit(X_quad, y_quad)
```

다양한 모델을 합쳐 학습:

차원을 줄여 필요한 부분의 성능을 강화하는 방법:

직접 투표 분류기 종류:

- 참/거짓:
- 확률:

훈련 세트에서 중복을 허용하여 샘플링:

- 학습 되지 못한 데이터로 테스트:

중복 허용 없이 샘플링:

처음에는 과소적합, 오분류에 가중치를 강하게 주는 기법:

- 대표적 1개:

결정 트리 쌓기:

1차원식 감소하는 법:

말린거 펼치기:

모든 것은 차원 축소 가능하다 가정:

- 항상 유효한지: O, X

```
pca = PCA(n_components=2) n_components의 뜻:
```

```
pca = PCA(n_components=0.95) n_components의 뜻:
```

