

정 규 세 션 4 주 차

ToBig's 11기 김유민

Decision Tree

의사결정나무

Content

Unit 01 | Decision Tree Overview

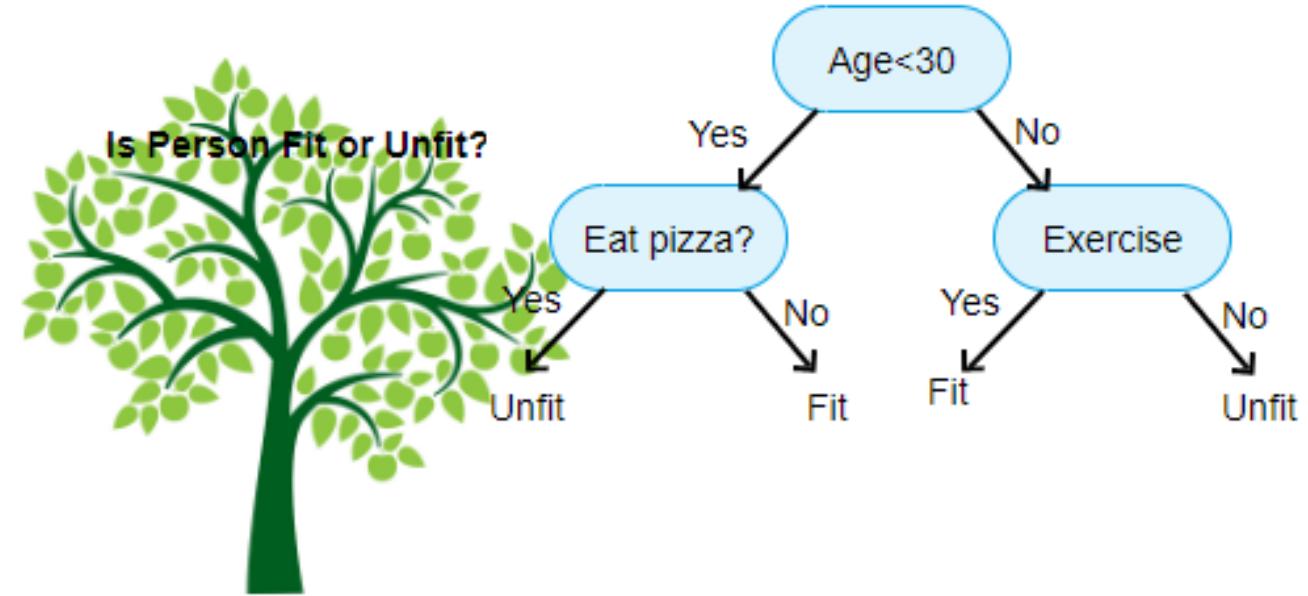
Unit 02 | The algorithm of growing DT

Unit 03 | Tree pruning

Unit 04 | Decision Tree with Sklearn

Unit 01 | Decision Tree Overview

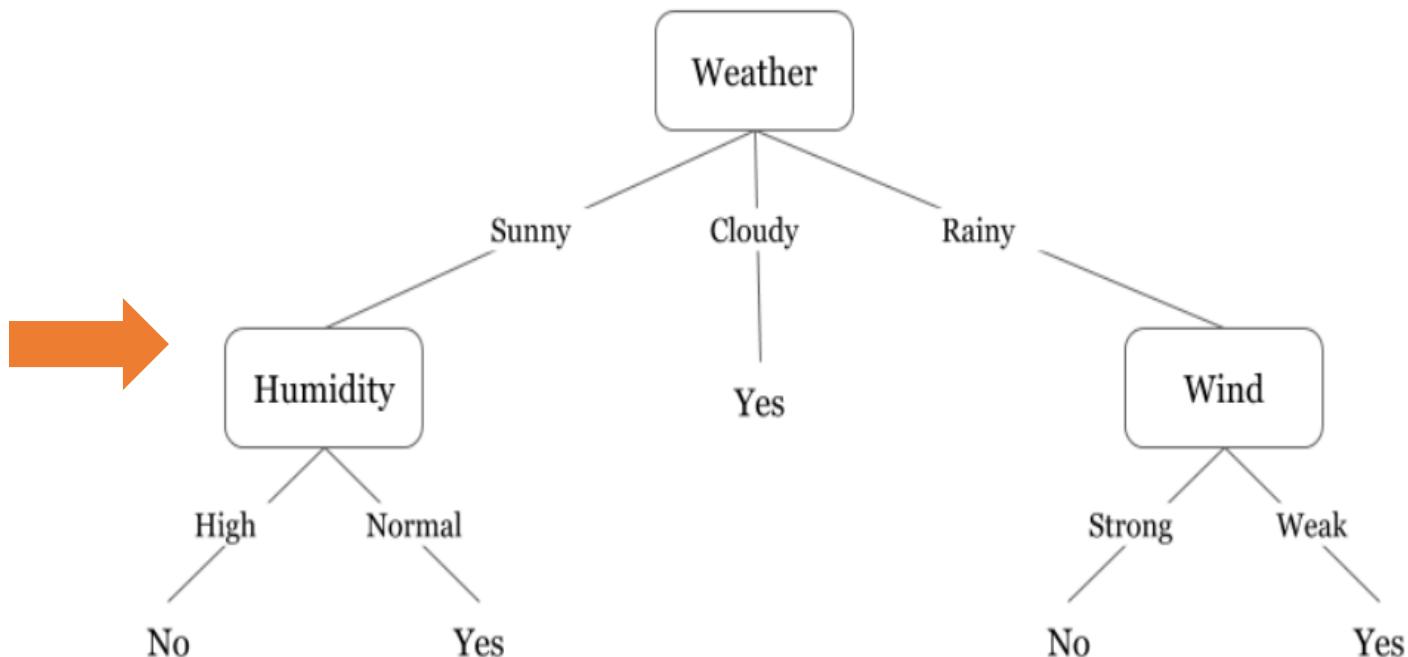
Decision
의사결정을 하는
Tree
나무모양 모델



Unit 01 | Decision Tree Overview

Day	Weather	Temperature	Humidity	Wind	Play?
1	Sunny	80	High	Weak	No
2	Cloudy	66	High	Weak	Yes
3	Sunny	43	Normal	Strong	Yes
4	Cloudy	82	High	Strong	Yes
5	Rainy	65	High	Strong	No
6	Rainy	42	Normal	Strong	No
7	Rainy	70	High	Weak	Yes
8	Sunny	81	High	Strong	No
9	Cloudy	69	Normal	Weak	Yes
10	Rainy	67	High	Strong	No

<Training Data>



<Decision Tree Model>

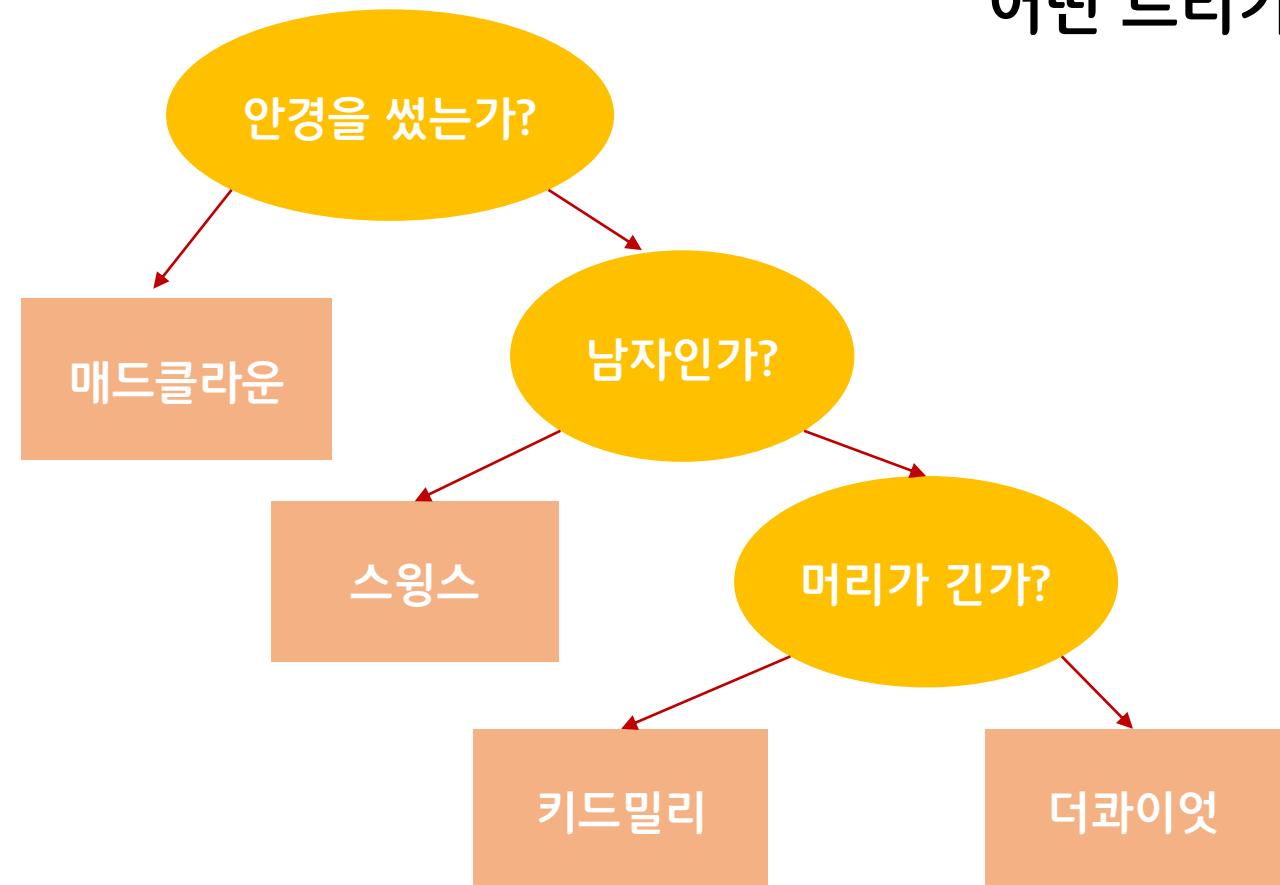
Unit 01 | Decision Tree Overview

Decision Tree Classifier

남자	긴 머리	안경	이름
예	아니오	예	매드클라운
예	아니오	아니오	스윙스
아니오	예	아니오	키드밀리
아니오	아니오	아니오	더콰이엇

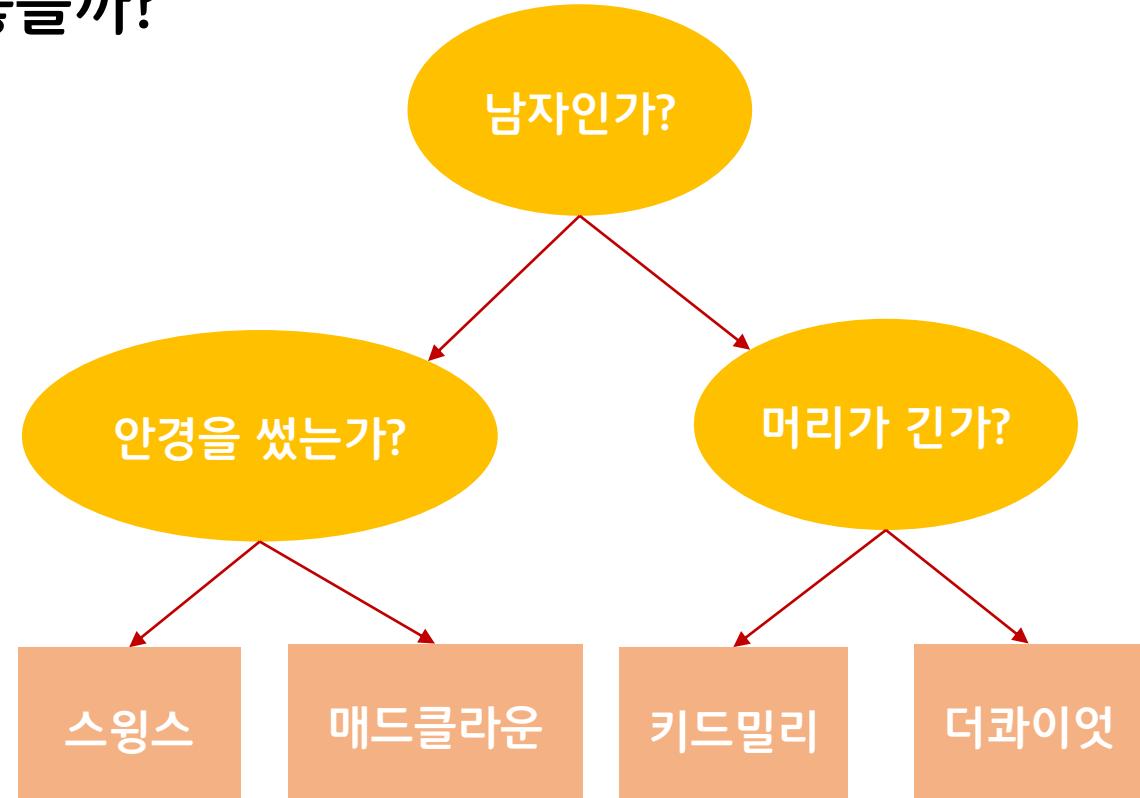
Unit 01 | Decision Tree Overview

[후보 1]



어떤 트리가 좋을까?

[후보 2]



Unit 01 | Decision Tree Overview

[후보 1]

어떤 트리가 좋을까?

[후보 2]

[판단 기준]

어떤 질문이 가장 많은 해답을 줄 수 있는가?

⇒ 어떤 질문이 답의 모호성을 줄여줄 수 있는가?

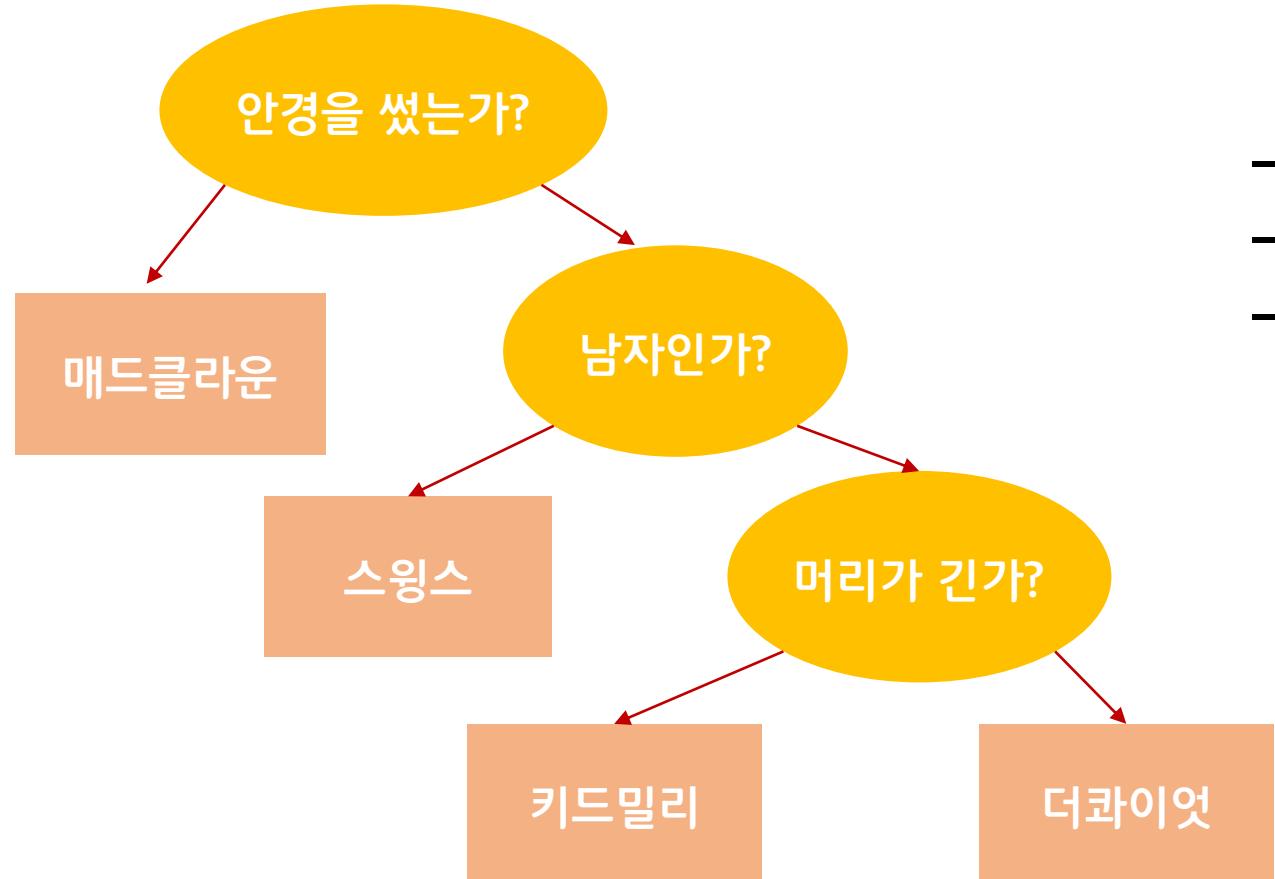
⇒ 정답과 오답을 얼마나 잘 필터링하는가!

키드밀리

더콰이엇

Unit 01 | Decision Tree Overview

[후보 1]



- > 질문 하나하나마다 답이 명확하게 구분됨!
- > [후보 2] 보다 모호하지 않게 필터링함!
- > Better Tree!

Unit 01 | Decision Tree Overview

Unit 02 | The algorithm of growing DT

Unit 03 | Tree pruning

Unit 04 | Decision Tree with Sklearn

Unit 02 | The algorithm of growing DT

Q1. '모호함(impurity)'을 측정하는 지표?

- A. Entropy, Gini Index, Classification error 등

Q2. 어떤 기준으로 노드를 놓아야 하며, 어떤 노드를 가장 위에 놓아야 할까?

- A. ID3 & CART 알고리즘으로 판단해!

Unit 02 | The algorithm of growing DT

Q1. '모호함(impurity)'을 측정하는 지표?

- A. Entropy, Gini Index, Classification error 등

Q2. 어떤 기준으로 노드를 놓아야 하며, 어떤 노드를 가장 위에 놓아야 할까?

- A. ID3 & CART 알고리즘으로 판단해!

용어에 겁먹지 말고 하나하나 차근차근 알아보아요!

Unit 02 | The algorithm of growing DT

ID3 Algorithm

✓ Entropy란?

A. '무질서도'를 정량화해서 표현한 값



High Entropy (messy)



Low Entropy (Clean)

Unit 02 | The algorithm of growing DT

ID3 Algorithm

✓ Entropy란?

A. '무질서도'를 정량화해서 표현한 값

- > 어떤 집합의 Entropy가 높을수록 그 집단의 특징을 찾는 것이 어려움
- > Decision Tree 잎 노드의 Entropy가 최소가 되는 방향으로 분류하는 것이 최적

$$\text{Entropy} = \sum_{i=1}^n -p(c_i) \log_2(p(c_i))$$

where $p(c_i)$ is the probability/percentage of class c_i in a node.

Unit 02 | The algorithm of growing DT

age	income	student	credit_rating	Class: buys_computer
youth	high	no	fair	no
youth	high	no	excellent	no
middle_aged	high	no	fair	yes
senior	medium	no	fair	yes
senior	low	yes	fair	yes
senior	low	yes	excellent	no
middle_aged	low	yes	excellent	yes
youth	medium	no	fair	no
youth	low	yes	fair	yes
senior	medium	yes	fair	yes
youth	medium	yes	excellent	yes
middle_aged	medium	no	excellent	yes
middle_aged	high	yes	fair	yes
senior	medium	no	excellent	no

ID3 Algorithm

Buys_computer에 대한 Entropy를 구해보자!



Unit 02 | The algorithm of growing DT

ID3 Algorithm

1. ID3

- Entropy를 도입하여 Branch Split을 해보자!
- Information Gain = 전체 Entropy – 속성별 Entropy
- Information Gain이 높을수록 명확한 정보를 얻을 수 있음.

Feature “A”에 대해
Information Gain을 구한다면!
(A의 class는 3개)

$$Gain(A) = Info(D) - Info_A(D_i)$$

$$Info(D) = Entropy_{label}$$

$$Info_A(D_i) = -\sum_{j=1}^3 \frac{|D_j|}{|D|} * Entropy_{label_j}$$

Unit 02 | The algorithm of growing DT

ID3 Algorithm



Cartoon? Winter? Many people?

Unit 02 | The algorithm of growing DT

img	cartoon	winter	> 1	Family winter photo
	No	Yes	Yes	Yes
	No	Yes	No	No
	Yes	No	Yes	No
	Yes	Yes	Yes	No
	No	Yes	No	No
	No	No	Yes	No
	Yes	No	Yes	No
	yes	yes	no	no

ID3 Algorithm

$$Gain(A) = \boxed{Info(D)} - Info_A(D_i)$$

전체 8개 사진

→ 겨울 가족 사진 Yes 1개

→ 겨울 가족 사진 No 7개

Unit 02 | The algorithm of growing DT

img	cartoon	winter	> 1	Family winter photo
	No		Yes	Yes
	No	Yes	No	No
	Yes	No	Yes	No
	Yes	Yes	Yes	No
	No	Yes	No	No
	No	No	Yes	No
	Yes	No	Yes	No
	yes	yes	no	no

ID3 Algorithm

$$0.543 \\ Gain(A) = \boxed{Info(D)} - \boxed{Info_A(D_i)}$$

$$Info_A(D_i) = - \sum_{j=1}^2 \frac{|D_j|}{|D|} * Entropy_{label_j}$$

$$Info(\text{cartoon}) = - \sum_{i=1}^2 \frac{|D_i|}{|D|} Entropy_{label_i}$$

$$\text{Yes} = 1, \text{No} = 2$$

- $\frac{|D_1|}{|D|} Entropy_{label_1} = \frac{4}{8} E([0, 4^-]) = \frac{4}{8} (-\frac{1}{4} \log_2 \frac{1}{4} - \frac{3}{4} \log_2 \frac{3}{4})$
- $\frac{|D_2|}{|D|} Entropy_{label_2} = \frac{4}{8} E([1, 3^-]) = \frac{4}{8} (-\frac{1}{4} \log_2 \frac{1}{4} - \frac{3}{4} \log_2 \frac{3}{4})$

$$\therefore Info(\text{cartoon}) = 0.408$$

$$\therefore Gain(\text{cartoon}) = \underline{0.543} - \underline{0.408} \\ = 0.138$$

Unit 02 | The algorithm of growing DT

img	cartoon	winter	> 1	Family winter photo
	No	Yes	Yes	Yes
	No	Yes	No	No
	Yes	No	Yes	No
	Yes	Yes	Yes	No
	No	Yes	No	No
	No	No	Yes	No
	Yes	No	Yes	No
	yes	yes	no	no

ID3 Algorithm

0.543

$$Gain(A) = \boxed{Info(D)} - \boxed{Info_A(D_i)}$$

$$Info_A(D_i) = -\sum_{j=1}^3 \frac{|D_j|}{|D|} * Entropy_{label_j}$$

$$Info(winter) = -\sum_{i=1}^2 \frac{|D_{xi}|}{|D|} Entropy_{label_i}$$

$$\text{Yes} = 1, \text{No} = 2$$

- $\frac{|D_1|}{|D|} Entropy_{label_1} = \frac{5}{8} E[1 + .4 -] = \frac{5}{8} (-\frac{1}{5} \log_2 \frac{1}{5} - \frac{4}{5} \log_2 \frac{4}{5}) = 0.45$
- $\frac{|D_2|}{|D|} Entropy_{label_2} = \frac{3}{8} E[0 + .3 -] = \frac{3}{8} (-\frac{0}{3} \log_2 \frac{0}{3} - \frac{3}{3} \log_2 \frac{3}{3}) = 0$

$$\therefore Info(winter) = 0.45$$

$$\therefore Gain(winter) = \underline{\underline{0.543}} - \underline{\underline{0.45}}$$

$$= 0.093$$

Unit 02 | The algorithm of growing DT

img	cartoon	winter	> 1	Family winter photo
	No	Yes	Yes	Yes
	No	Yes	No	No
	Yes	No	Yes	No
	Yes	Yes	Yes	No
	No	Yes	No	No
	No	No	Yes	No
	Yes	No	Yes	No
	yes	yes	no	no

ID3 Algorithm

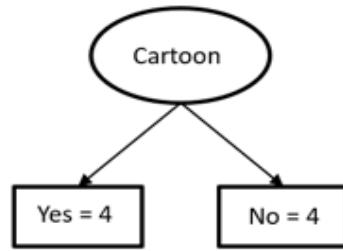
0.543

$$Gain(A) = \boxed{Info(D)} - \boxed{Info_A(D_i)}$$

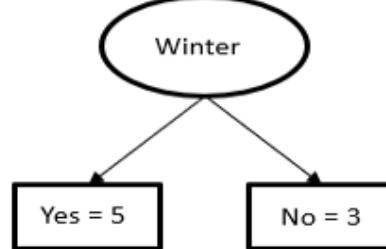
$$\boxed{Info_A(D_i)} = -\sum_{j=1}^3 \frac{|D_j|}{|D|} * Entropy_{label_j}$$

Unit 02 | The algorithm of growing DT

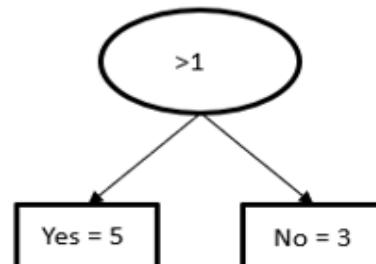
ID3 Algorithm



$$\begin{aligned}\text{Information Gain(winter family photo, cartoon)} \\ &= 0.543 - (4/8 * E([0+, 4-]) + 4/8 * E([1+, 3-])) \\ &= 0.138\end{aligned}$$



$$\begin{aligned}\text{Information Gain(winter family photo, winter)} \\ &= 0.543 - (5/8 * E([1+, 4-]) + 3/8 * E([0+, 3-])) \\ &= 0.093\end{aligned}$$



$$\begin{aligned}\text{Information Gain(winter family photo, >1)} \\ &= 0.543 - (5/8 * E([1+, 4-]) + 3/8 * E([0+, 3-])) \\ &= 0.093\end{aligned}$$

Information Gain이 가장 높음!
→ cartoon이냐 아니냐로
branch split했을 때
가장 정보 획득량이 큼
→ 최초 split “cartoon” column

Unit 02 | The algorithm of growing DT

CART Algorithm

2. CART

- Gini index를 도입하여 Branch Split을 해보자!
- 데이터를 split했을 때 불순한 정도
- 데이터의 대상 속성을 얼마나 잘못 분류할지 계산
- Binary split을 전제로 분석함
- Feature의 데이터 분류 개수가 k개일 때 $2^{k-1} - 1$ 개 만큼의 split 생성

Feature “A”에 대해
Gini 계수를 구한다면!
(A의 class는 3개)



$$Gini(A) = \sum_{j=1}^2 \frac{|D_j|}{|D|} * Gini(D_i)$$

$$Gini(D_i) = 1 - \sum_{j=1}^3 P_j$$

Unit 02 | The algorithm of growing DT

<i>age</i>	<i>income</i>	<i>student</i>	<i>credit_rating</i>	<i>Class: buys_computer</i>
youth	high	no	fair	no
youth	high	no	excellent	no
middle_aged	high	no	fair	yes
senior	medium	no	fair	yes
senior	low	yes	fair	yes
senior	low	yes	excellent	no
middle_aged	low	yes	excellent	yes
youth	medium	no	fair	no
youth	low	yes	fair	yes
senior	medium	yes	fair	yes
youth	medium	yes	excellent	yes
middle_aged	medium	no	excellent	yes
middle_aged	high	yes	fair	yes
senior	medium	no	excellent	no

Age	$Gini_{age}(D)$
Credit	$Gini_{credit}(D)$
Income	$Gini_{income}(D)$
Student	$Gini_{student}(D)$

이 중
가장 작은 Gini index값을 가지는 변수가
최초 split이 될거야!

Unit 02 | The algorithm of growing DT

Age에 대한 Gini index를 먼저 구해보자!

CART Algorithm

	RID	age	income	student	credit_rating	class_buys_computer
0	1	youth	high	no	fair	no
1	2	youth	high	no	excellent	no
7	8	youth	medium	no	fair	no
8	9	youth	low	yes	fair	yes
10	11	youth	medium	yes	excellent	yes

	RID	age	income	student	credit_rating	class_buys_computer
2	3	middle_aged	high	no	fair	yes
3	4	senior	medium	no	fair	yes
4	5	senior	low	yes	fair	yes
5	6	senior	low	yes	excellent	no
6	7	middle_aged	low	yes	excellent	yes
9	10	senior	medium	yes	fair	yes
11	12	middle_aged	medium	no	excellent	yes
12	13	middle_aged	high	yes	fair	yes
13	14	senior	medium	no	excellent	no

Unit 02 | The algorithm of growing DT

CART Algorithm

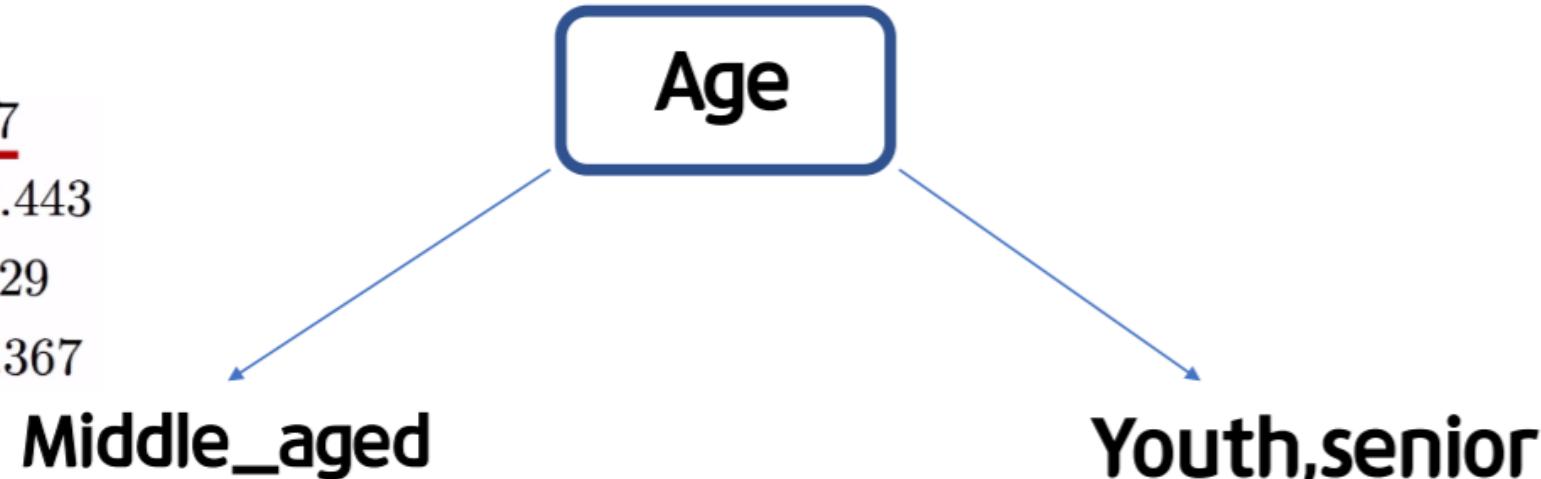
Gini Index

$$\underline{Min(Gini_{age_i})} = 0.357$$

$$Min(Gini_{income_i}) = 0.443$$

$$Min(Gini_{credit}) = 0.429$$

$$Min(Gini_{student}) = 0.367$$



	age	income	student	credit_rating	class_buys_computer
2	middle_aged	high	no	fair	yes
6	middle_aged	low	yes	excellent	yes
11	middle_aged	medium	no	excellent	yes
12	middle_aged	high	yes	fair	yes

	age	income	student	credit_rating	class_buys_computer
0	youth	high	no	fair	no
1	youth	high	no	excellent	no
3	senior	medium	no	fair	yes
4	senior	low	yes	fair	yes
5	senior	low	yes	excellent	no
7	youth	medium	no	fair	no
8	youth	low	yes	fair	yes
9	senior	medium	yes	fair	yes
10	youth	medium	yes	excellent	yes
13	senior	medium	no	excellent	no

Unit 02 | The algorithm of growing DT

Continuous Feature

Q. 연속형 변수는 어떻게 split하나요?

- 1) 전체 데이터를 모두 기준점으로 하거나,
- 2) 중위수, 사분위수를 기준점으로 하거나,
- 3) Label의 class가 바뀌는 수를 기준점으로!

Unit 02 | The algorithm of growing DT

Continuous Feature

Step 1. Split할 연속형 변수를 sorting한다.

	ID	STREAM	SLOPE	ELEVATION	VEGETATION
0	1	False	steep	3900	chapparal
1	2	True	moderate	300	riparian
2	3	True	steep	1500	riparian
3	4	False	steep	1200	chapparal
4	5	False	flat	4450	conifer
5	6	True	steep	5000	conifer
6	7	True	steep	3000	chapparal

ELEVATION
300
1200
1500
3000
3900
4450
5000

Unit 02 | The algorithm of growing DT

Continuous Feature

Step 2. Label의 class가 바뀌는 지점을 찾는다. Step 3. 경계의 평균값으로 기준값을 잡는다.

	ID	STREAM	SLOPE	ELEVATION	VEGETATION
1	2	True	moderate	300	riparian
3	4	False	steep	1200	chapparal
2	3	True	steep	1500	riparian
6	7	True	steep	3000	chapparal
0	1	False	steep	3900	chapparal
4	5	False	flat	4450	conifer
5	6	True	steep	5000	conifer

(1)
(2)
(3)
(4)

	ID	STREAM	SLOPE	ELEVATION	VEGETATION
1	2	True	moderate	300	riparian
3	4	False	steep	1200	chapparal
2	3	True	steep	1500	riparian
6	7	True	steep	3000	chapparal
0	1	False	steep	3900	chapparal
4	5	False	flat	4450	conifer
5	6	True	steep	5000	conifer

750
1,350
2250
4175

Unit 02 | The algorithm of growing DT

Continuous Feature

Step 4. 구간별 경계값을 기준으로 Entropy 또는 Gini를 산출한다.

$$\text{Gain}(\text{elec}_{750}) = \text{Info}(D) - \text{Info}_{\text{elec}_{750}}(D)$$

$$\text{Gain}(\text{elec}_{1350}) = \text{Info}(D) - \text{Info}_{\text{elec}_{1350}}(D)$$

$$\text{Gain}(\text{elec}_{2250}) = \text{Info}(D) - \text{Info}_{\text{elec}_{2250}}(D)$$

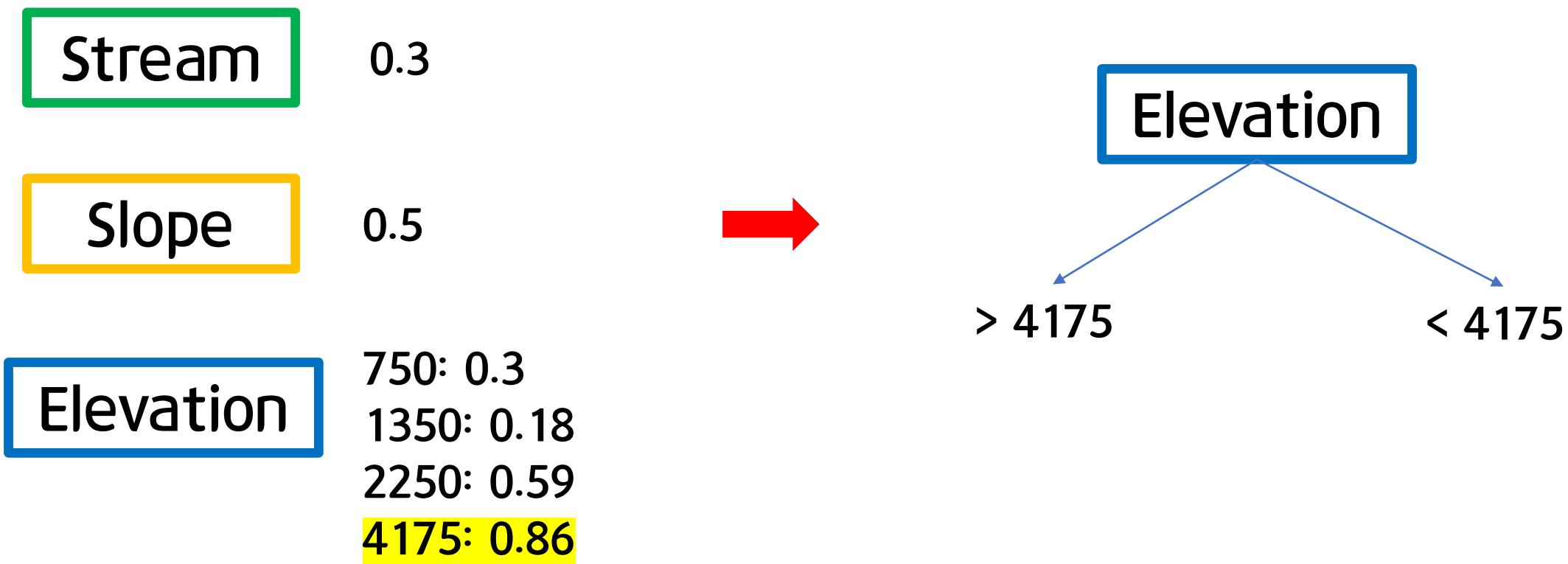
$$\text{Gain}(\text{elec}_{4175}) = \text{Info}(D) - \text{Info}_{\text{elec}_{4175}}(D)$$

$$\text{Max}(\text{Gain}(\text{elec}))$$

Unit 02 | The algorithm of growing DT

Continuous Feature

Step 5. 최종 Split point 선택!



Content

Unit 01 | Decision Tree Overview

Unit 02 | The algorithm of growing DT

Unit 03 | Tree pruning

Unit 04 | Decision Tree with Sklearn

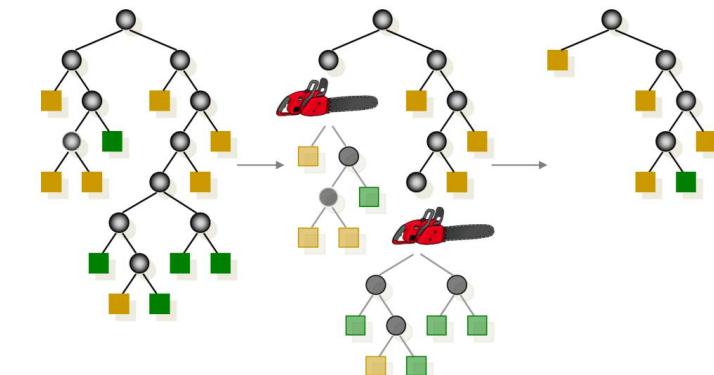
Unit 03 | Tree Pruning

Continuous Feature

✓ DT의 문제점?

계속 분류하다 보면... 마지막에는 순수 node 탄생

- > 제한없이 분기를 허용하면 overfitting 발생함
+ 모델이 너무 복잡해짐
- > 가지치기(pruning) 도입



Unit 03 | Tree Pruning

✓ Pruning

- 1) Pre-pruning(사전 가지치기)
 - 트리의 최대 depth나 분기점의 최소 개수를 미리 지정
- 2) Post-pruning(사후 가지치기 또는 가지치기)
 - 트리를 만든 후 데이터 포인트가 적은 노드를 삭제 or 병합

Unit 03 | Tree Pruning

✓ DT의 장단점

[장점]

1. 모델 시각화가 용이함 -> 비전문가도 쉽게 해석 가능!
2. 각 특성이 개별적으로 처리되어 데이터 스케일 영향X -> 전처리 필요X
3. 특성의 스케일이 서로 다르거나 연속형/범주형 혼합데이터인 경우에도 적용 가능

[단점]

1. 가지치기를 해도 과대적합되는 경향이 있음 -> 일반화 성능 bad
=> 해결방안? Ensemble!

Content

Unit 01 | Decision Tree Overview

Unit 02 | The algorithm of growing DT

Unit 03 | Tree pruning

Unit 04 | Decision Tree with Sklearn

Unit 04 | Decision Tree with Sklearn

✓ Decision Tree with Sklearn

sklearn.tree.DecisionTreeClassifier

```
class sklearn.tree.DecisionTreeClassifier(criterion='gini', splitter='best', max_depth=None, min_samples_split=2,
min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features=None, random_state=None, max_leaf_nodes=None,
min_impurity_decrease=0.0, min_impurity_split=None, class_weight=None, presort=False) [source]
```

Criterion: 트리 성장 알고리즘 ('Entropy' 또는 'gini' 입력)

Max_depth: 트리 깊이 설정(default = None)

Min_samples_split: 노드 분기할 때 최소 데이터 개수 기준

Min_samples_leaf: 리프에 있어야하는 데이터의 최소 개수

Q & A

들어주셔서 감사합니다.

Reference

ToBig's 11기 정규세션 Decision Tree 강의(이준걸님)

Decision Tree 강의(허민석님)

가천대학교 의사결정나무모델 강의

<https://www.youtube.com/watch?v=n0p0120Gxqk>

<https://www.youtube.com/watch?v=UPKugg0fK04>

<https://tensorflow.blog/%ED%8C%8C%EC%9D%B4%EC%8D%AC-%EB%A8%B8%EC%8B%A0%EB%9F%AC%EB%8B%9D/2-3-5-%EA%B2%B0%EC%A0%95-%ED%8A%B8%EB%A6%AC/>