

• 12/12

Decision Tree (의사결정 Tree) ④ 회귀학습 (Regression) Classification → 이 블록이 조금 더 적합 → 사람들이 일반적으로 의사결정하는 방식과 유사 어떤 사람이 청소년, 노부인, 대학생으로 알아보아요! Model

④ 사람이 사용하는 프로그램과 아주 유사한

↳ 주로 앱, 웹사이트에 일반적으로 사람이 사용하는
경우와 유사한 경우)

* 간결 → 속도가 빠르고 간단해요!

데이터를 파악 상태적으로 다른 model로

넘쳐 속도가 좋아요!

* 도구 → 독점변수인 이전데이터의 경우에 쓰임
(특성)
Class 추가 많은 경우에 사용이 헛들어요.

데이터의 추가 쪽으면 좋지 않아요

* 예 스포츠경기 (예전과
현재와 함께) →
 놀이 → 힙, 큐, 맵
 속도 → 높다, 보통
 비용 → 많다, 적다

X

날씨 ✓

습도 ✓

바람✓

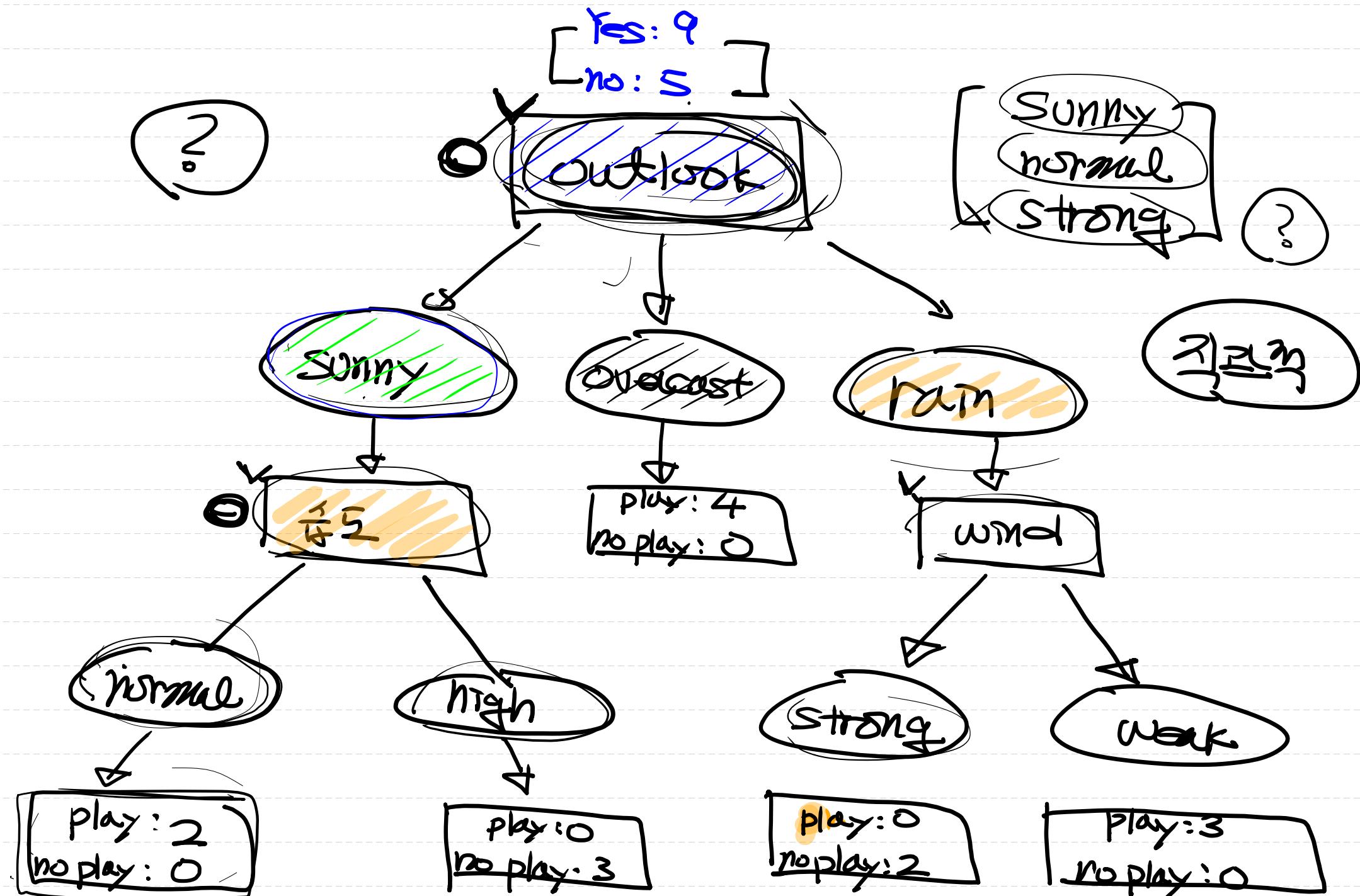
天气

4

天气预报

DAY	OUTLOOK	HUMIDITY	WIND	PLAY
D1	SUNNY	HIGH	WEAK	NO
D2	SUNNY	HIGH	STRONG	NO
D3	OVERCAST	HIGH	WEAK	YES
D4	RAIN	HIGH	WEAK	YES
D5	RAIN	NORMAL	WEAK	YES
D6	RAIN	NORMAL	STRONG	NO
D7	OVERCAST	NORMAL	STRONG	YES
D8	SUNNY	HIGH	WEAK	NO
D9	SUNNY	NORMAL	WEAK	YES
D10	RAIN	NORMAL	WEAK	YES
D11	SUNNY	NORMAL	STRONG	YES
D12	OVERCAST	HIGH	STRONG	YES
D13	OVERCAST	NORMAL	WEAK	YES
D14	RAIN	HIGH	STRONG	NO

* model (Decision Tree)을 어떻게 만들면 좋을까요??



④ 주어진 Data3 Tree를 구성하려해요

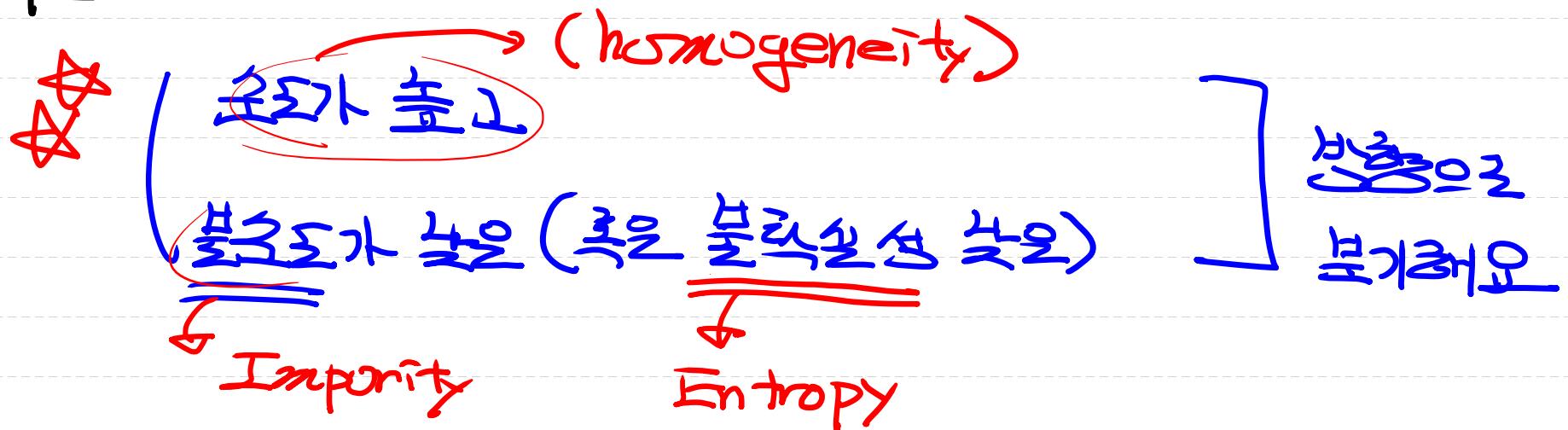
→ 어떤 기준(feature)을 가지고 Tree를 만들어 각 줄지를 결정해야해요..!!

→ 조건들면 축소되는 예측이 높아져요!

Decision Tree는 Tree를 만들어 나가는 과정이 흐름과 같이예요

↳ Tree를 분기해 나가는 과정

어떤 기준으로 분기해나요?



문도가 증가하고 / 물도도

불확실성

감소하는 경향

Information gain (정보량)

불확실성이 감소

이 값이 크게 발생하는
상황으로 학습이 진행

불확실성 (entropy)

무질서도를 측정하는 개념

entropy > 높아질 때

정보량을
측정하는
법

$$I(2) = \log_2 \frac{1}{P(A)}$$

→ ↑

무질서도가 ↑

↳ 토양을 죽이기 힘들어짐

아름에 해가 통증이면 폐요 → 학률 ① ↑ 진통제 ↑

// " 불족이면 " → 학률 0이자 진통제 ↑

★ A 상황에서 B 상황에 전이할 때 ✓

A

전이
할 때

불리한 상황

* Entropy가 높다

전보증이 많다

B

불리한 상황

* Entropy가 낮다

전보증이 적어요

전보증이 많아져요.



결국은 Decision Tree의 뿌리는

전보증이 많을 경향으로 진행!!

전체 Entropy에서 본래의 Entropy를 뺀 값

A영역에 대한 Entropy는 다음의 식으로 표현!

(도입영역
있으므로)

~~*~~

$$\overline{\text{Entropy}}(A) = - \sum_{k=1}^m P_k \log_2(P_k)$$

개념 (class)

A영역에 속하는 도입영역(기준)

(개념)에 속하는 도입영역

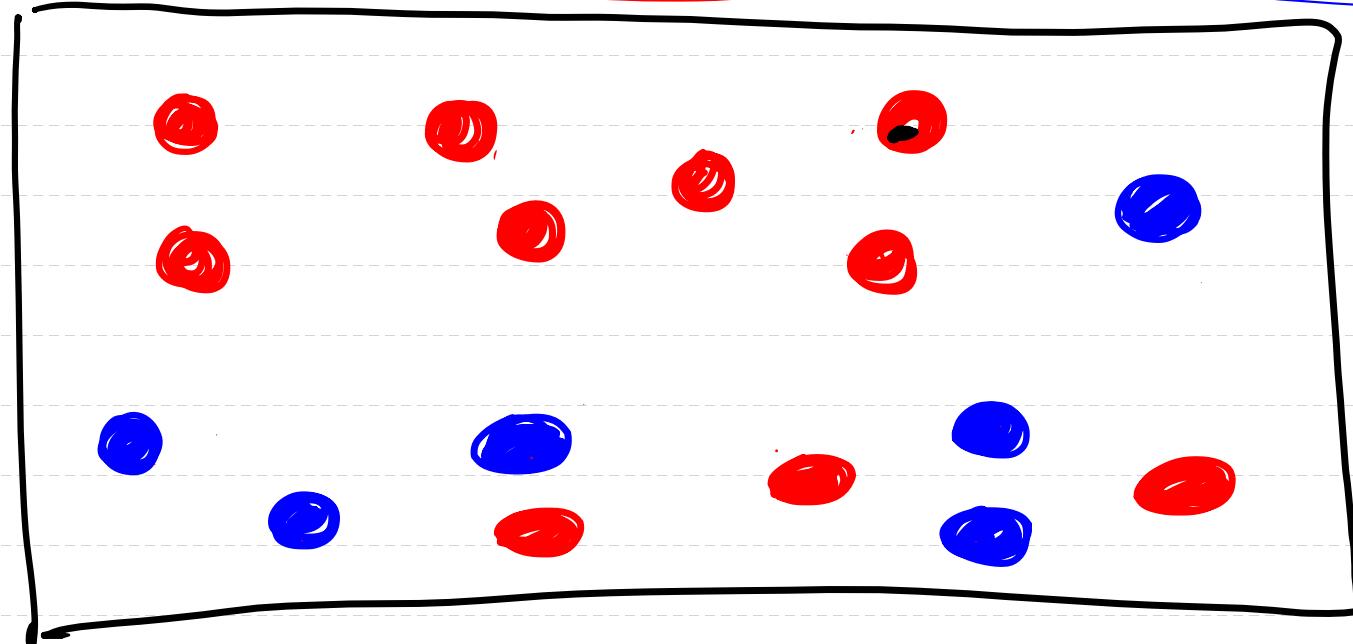
비율

$$- \left(\frac{10}{16}\right) \log_2\left(\frac{10}{16}\right) - \left(\frac{6}{16}\right) \log_2\left(\frac{6}{16}\right)$$

k 개로

예)

A영역
(B영역
포함)



entropy는
무엇인가?

0.95

10개

정보특성

(선행 전이의 가능도)

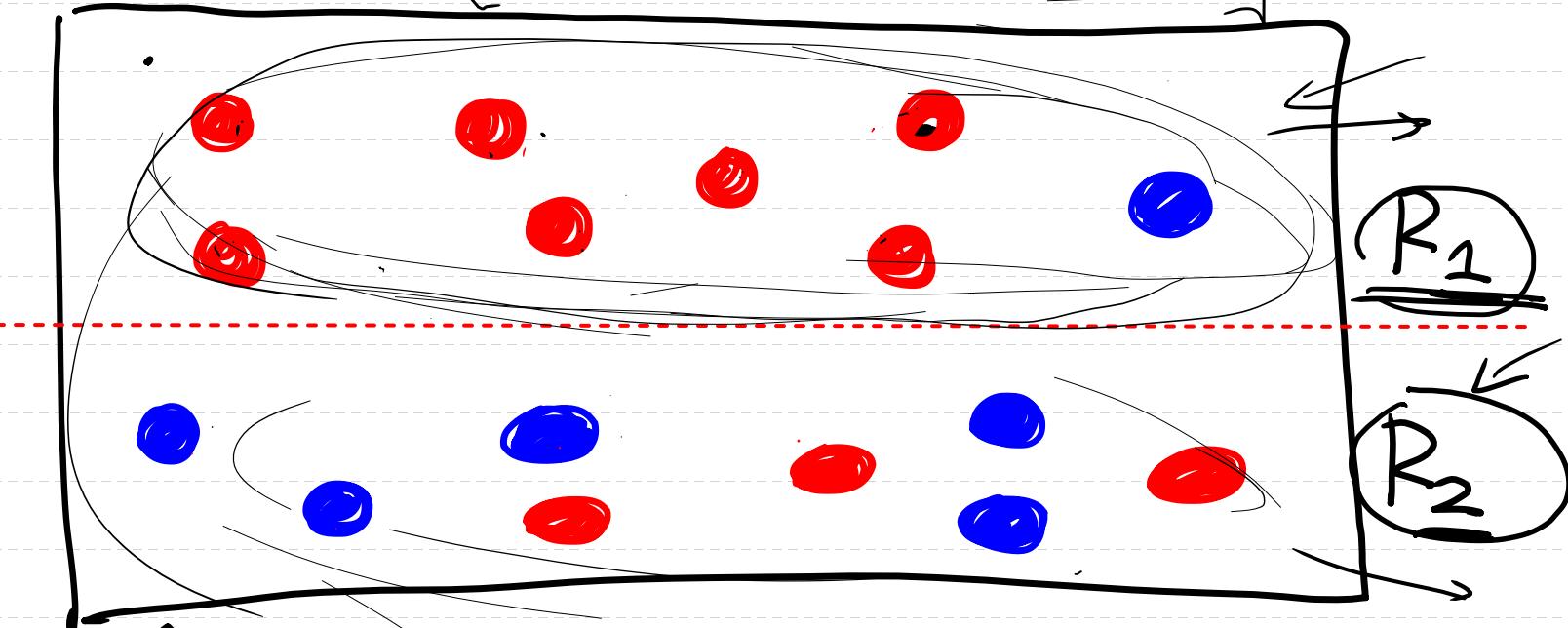
→ 전체 entropy - 분자熵 entropy

정보특성이 깊은 분포로 상세한 전이

여기까지 tree를 보기

$$\text{Entropy}(F) = \sum_{j=1}^d R_j \left(- \sum_{k=1}^m P_k \log(P_k) \right)$$

R_j 는 분할한 드리븐
가운데 분할후 영역이
족히는 드리븐 비율



$$\frac{5}{8} \left(-\frac{5}{8} \log_2 \left(\frac{5}{8} \right) \right) - \frac{3}{8} \log_2 \left(\frac{3}{8} \right) + \frac{3}{8} \times \left(-\frac{3}{8} \log_2 \left(\frac{3}{8} \right) \right) -$$

특성을
도입하는
부록

● 정리!!

Decision Tree

- * 상수의 조이가 (초보가 추가하는 방향으로 진행)
(entropy가 최대한 감소하는 방향으로 진행)
- * 하나의 열의 통으로는 끝의 data를 판별 → 불확실성 → 0
- " 두 병중의 디자이너가 봄비씩 판별 → 불확실성 ↑ → 1
entropy

날씨

온도

비습

DAY	OUTLOOK	HUMIDITY	WIND	PLAY
D1	SUNNY	HIGH	WEAK	NO
D2	SUNNY	HIGH	STRONG	NO
D3	OVERCAST	HIGH	✓ WEAK	YES
D4	RAIN	HIGH	✓ WEAK	YES
D5	RAIN	NORMAL	✓ WEAK	YES
D6	RAIN	NORMAL	STRONG	NO
D7	OVERCAST	NORMAL	STRONG	YES
D8	SUNNY	HIGH	WEAK	NO
D9	SUNNY	NORMAL	✓ WEAK	YES
D10	RAIN	NORMAL	✓ WEAK	YES
D11	SUNNY	NORMAL	STRONG	YES
D12	OVERCAST	HIGH	STRONG	YES
D13	OVERCAST	NORMAL	✓ WEAK	YES
D14	RAIN	HIGH	STRONG	NO

① 만족도 상태 (root node)의 Entropy를 구해보아요

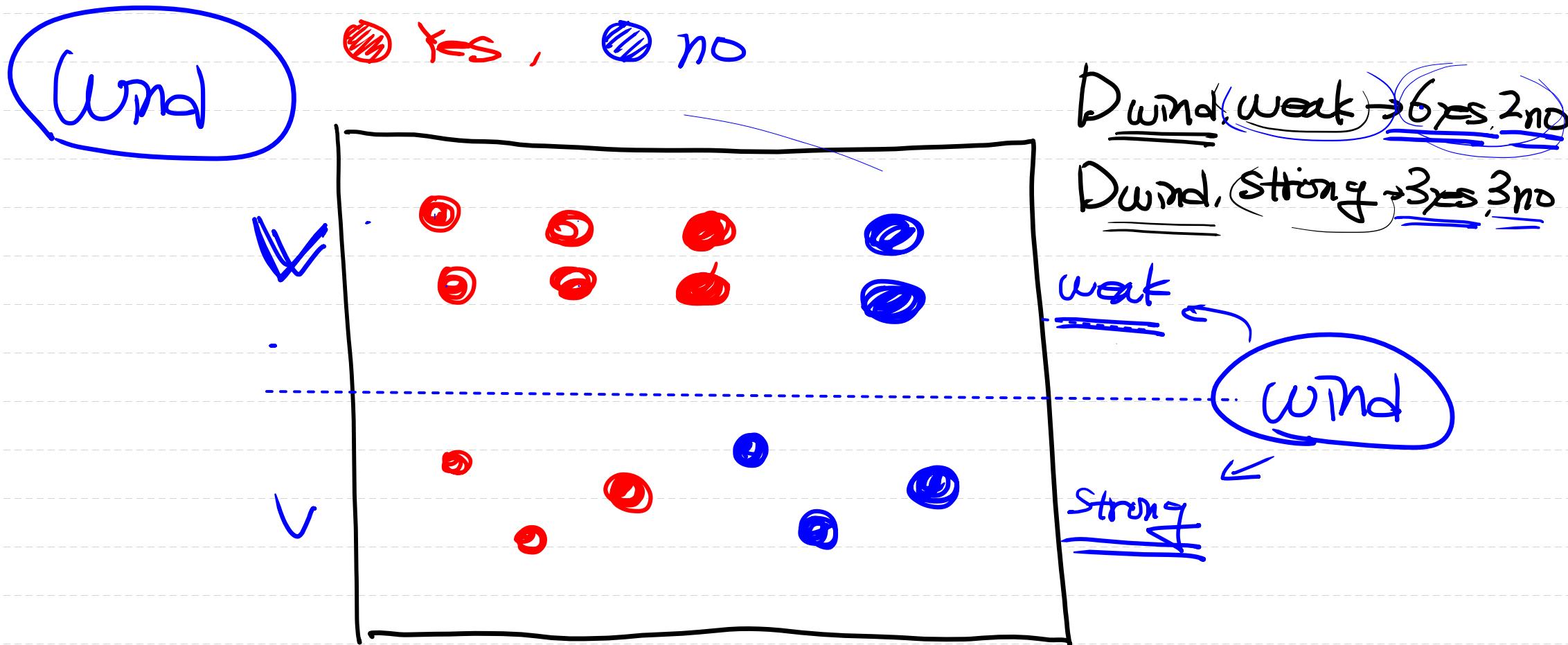
✓

$$\text{Entropy} = [9_{\text{yes}}, 5_{\text{no}}]$$

$$= - \left(\frac{9}{14} \log_2 \left(\frac{9}{14} \right) + \left(\frac{5}{14} \log_2 \left(\frac{5}{14} \right) \right) \right)$$

$$\approx 0.94$$

② Root node의 Wind3 보통화인 entropy를 계산!



$$\frac{3}{14} \times \left(-\frac{6}{8} \log_2 \left(\frac{6}{8} \right) - \frac{2}{8} \log_2 \left(\frac{2}{8} \right) \right) +$$

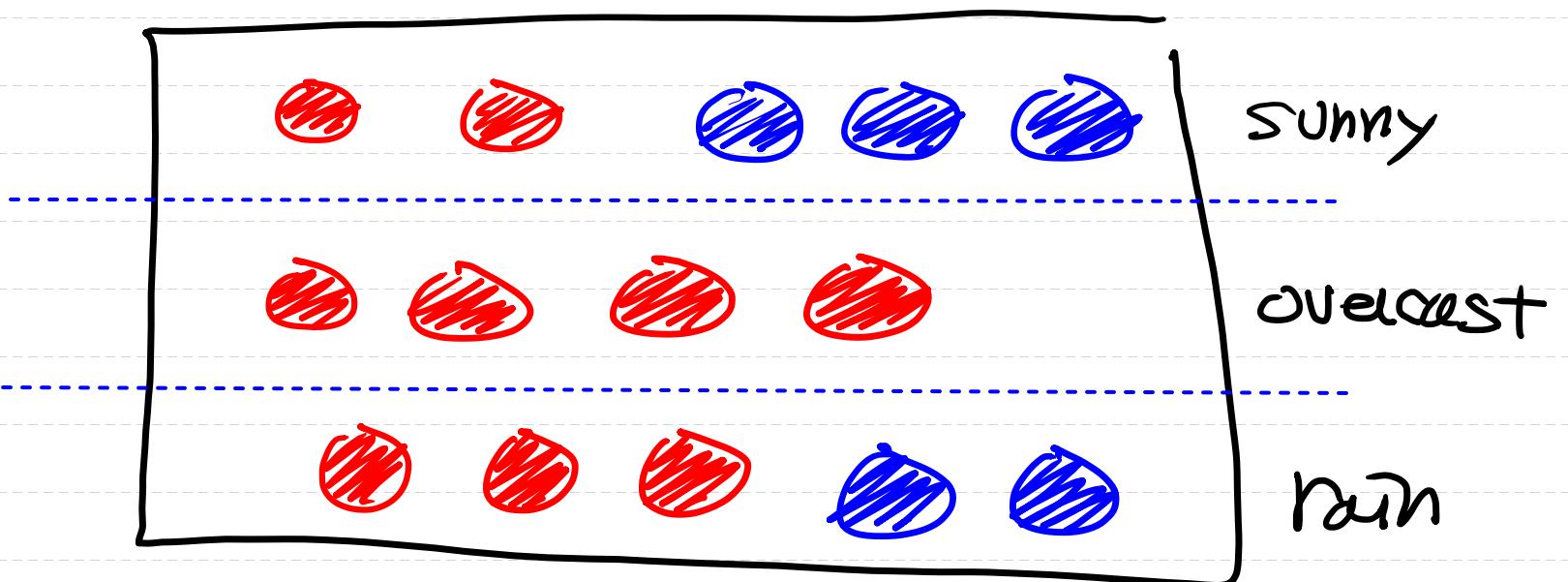
$$\frac{6}{14} \times \left(-\frac{3}{6} \log_2 \left(\frac{3}{6} \right) - \frac{3}{6} \log_2 \left(\frac{3}{6} \right) \right) \Rightarrow$$

Yes no

Outlook, sunny = [2 3]

Outlook, overcast = [4 0]

Outlook, rain = [3 2]



$$\frac{5}{14} \times \left(-\frac{2}{5} \log_2 \left(\frac{2}{5}\right) - \frac{3}{5} \log_2 \left(\frac{3}{5}\right) \right) +$$

$$\frac{4}{14} \times \left(-\frac{4}{4} \log_2 \left(\frac{4}{4}\right) - \frac{0}{4} \log_2 \left(\frac{0}{4}\right) \right) +$$

$$\frac{5}{14} \times \left(-\frac{3}{5} \log_2 \left(\frac{3}{5}\right) - \frac{2}{5} \log_2 \left(\frac{2}{5}\right) \right) =$$

★

KNN (K-Nearest Neighbors)

[K-최근접(이웃)]

모델

점대비로 근접

가장 간단한 알고리즘.

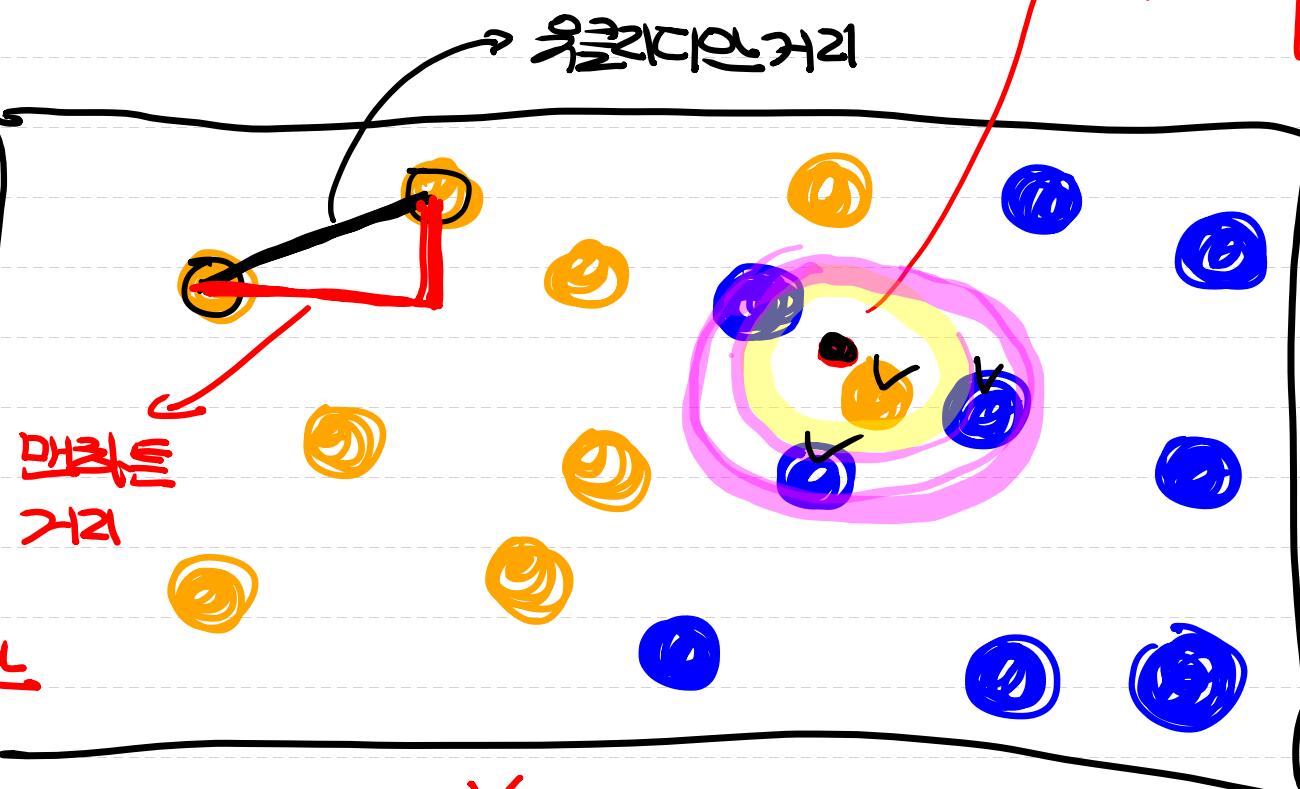
hyperparameters

K 몇개의
최근접이웃을
선택할지

거리를 정복

일본자로
유전

거리



If K=1
제일 가까운 노란!

If K=3
제일 가까운 블루!

학습과정이 없어요

→ Lazy Model

Instance-based Learning

* 일반적으로 $k=1$ 일정우

구현으로 어느정도 성과 보장률을 증명

→ 구현해 보아요!!

기계학습
알고리즘



linear Regression
Logistic Regression

Sklearn
Tensorflow

Sklearn
구현

기계학습
알고리즘

K-Means (기각기준 clustering)

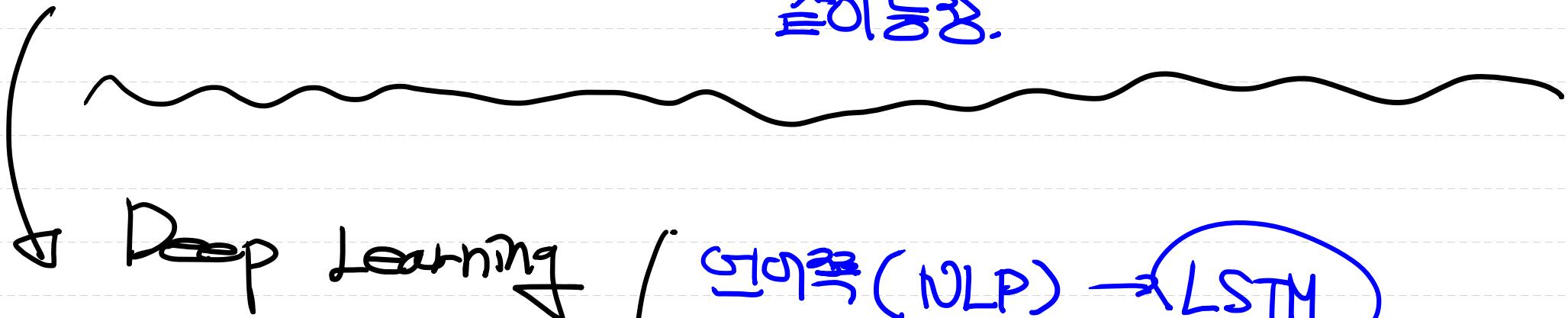
DBSCAN (밀도기준 clustering) → 속도가 느릴수 있어요

(최초축소) (PCA - 주성분분석)

그럼 모델 1개만 쓰는게 아니라

여러개의 model을 사용해서 ~~분류하면 더 좋을거~~
간단해요!

* → 랜덤 포레스트 (Random Forest)
脾이 좋음.



언어학 (NLP) → LSTM

이미지학 (VISION) → CNN