

Dong-A Univ. (ISPL)



동아대학교
DONG-A UNIVERSITY

Entropy & Decision Tree & KNN Method

컴퓨터공학과
2024년 2학기 머신러닝



ENTROPY & Decision Tree

▪ Entropy: Measurement for Uncertainty

- 불확실성(Uncertainty)의 정도를 나타내는 수치

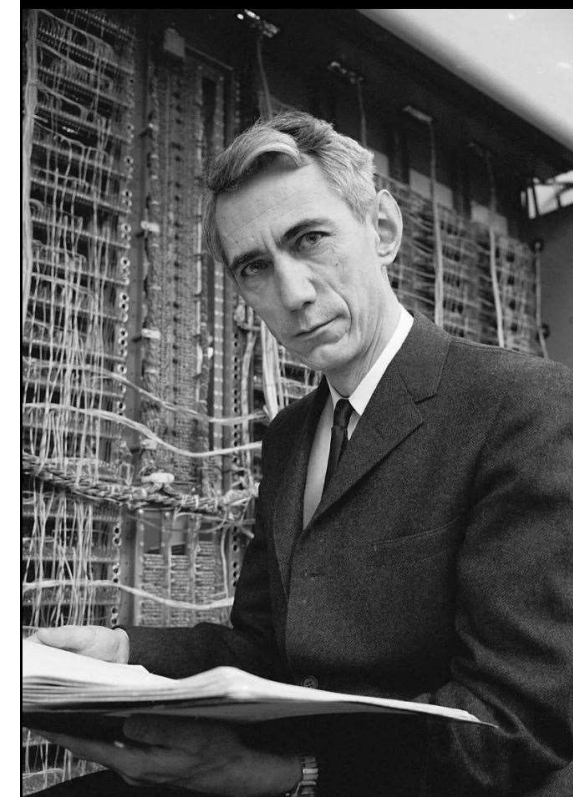
SHANNON ENTROPY

$$H(X) = - \sum_{i=1}^n p_i \log_2 p_i$$

		Calculation	Entropy
50%	50%	$-(0.5 * \log_2 0.5 + 0.5 * \log_2 0.5) = 1$	1
100%	0%	$-(1.0 * \log_2 1 + 0.0 * \log_2 0) = 0$	0
90%	10%	$-(0.9 * \log_2 0.9 + 0.1 * \log_2 0.1) = 0.47$	0.47

$0 \leq \text{Entropy} \leq 1$

클로드 새넌
Claude Shannon



본명	클로드 엘우드 새넌 Claude Elwood Shannon
출생	1916년 4월 30일 미국 미시간 주 페토스키
사망	2001년 2월 24일 (향년 84세) 미국 매사추세츠 주 메드퍼드
국적	 미국
직업	학자, 교수
분야	수학, 컴퓨터과학, 전자공학, 암호학
【펼치기 · 접기】	

▪ 불확실성이 높을수록 엔트로피는 큰 값을 가짐

- Binary Classification: $0 \leq \text{Entropy} \leq 1$
- 8-classes Classification: $0 \leq \text{Entropy} \leq 3$
- 16 classes Classification: $0 \leq \text{Entropy} \leq 4$

▪ 머신러닝에서 엔트로피 활용 예

- [1] Deep Learning의 Loss Function
- [2] Decision Tree
- [3] Active Learning

MLP: Loss Function

- **Loss function:** 학습 모델이 얼마나 잘못 예측하고 있는지는 표현하는 지표

- 값이 낮을수록 모델이 정확하게 예측했다고 해석할 수 있음
- Ex. Cross Entropy Error (CEE) 계산 방법

$$CEE(y, y') = - \sum_{i=1}^N y_i \times \log(y_i')$$

- ❖ y: 정답 값
- ❖ y': 예측 값



$$h(x) = - \sum_{i=1}^n (p_i \log_2(p_i))$$

0	1	2	3	4	5	6	7	8	9
0	0	1	0	0	0	0	0	0	0

정답 값 (y, one-hot)

Model A의 예측 결과

0	1	2	3	4	5	6	7	8	9
0	0	0.8	0	0	0	0.1	0	0.1	0

예측 확률 (y') **CEE = 0.2231**

$$CEE(y, y') = -(1 \times \log(0.8)) = 0.2231$$

■ 머신러닝에서 엔트로피 사용 예: [Decision Tree \(DT\)](#)

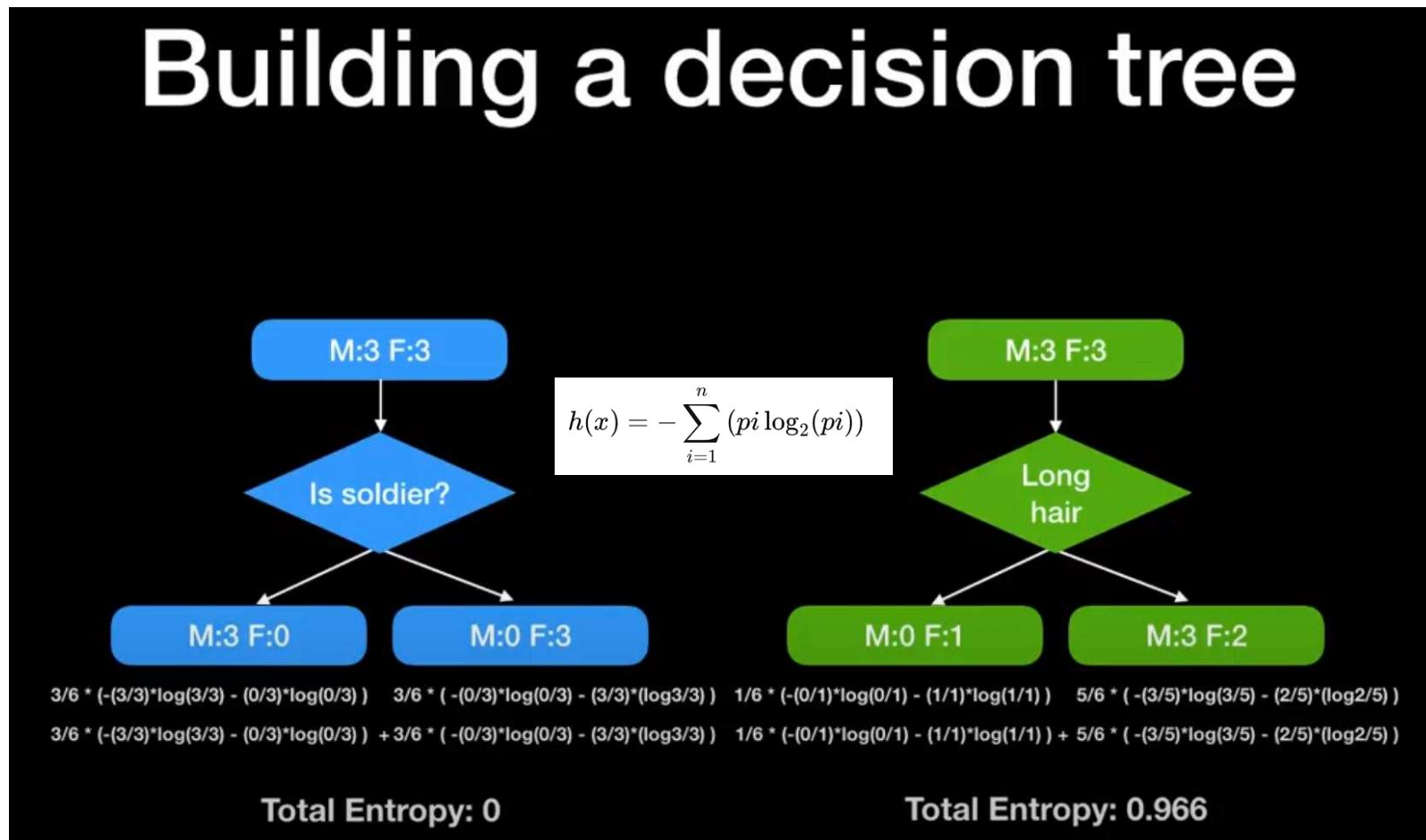
- DT에서 **확실히** 구분이 되는 특징을 먼저 구분해 주는 것이 중요
- 확실히 구분이 되는 특징은 **불확실성(엔트로피)**가 작다는 것을 의미

person	Is soldier?	Long hair?	gender
1	yes	no	Male
2	no	no	Female
3	yes	no	Male
4	no	yes	Female
5	yes	no	Male
6	no	no	Female



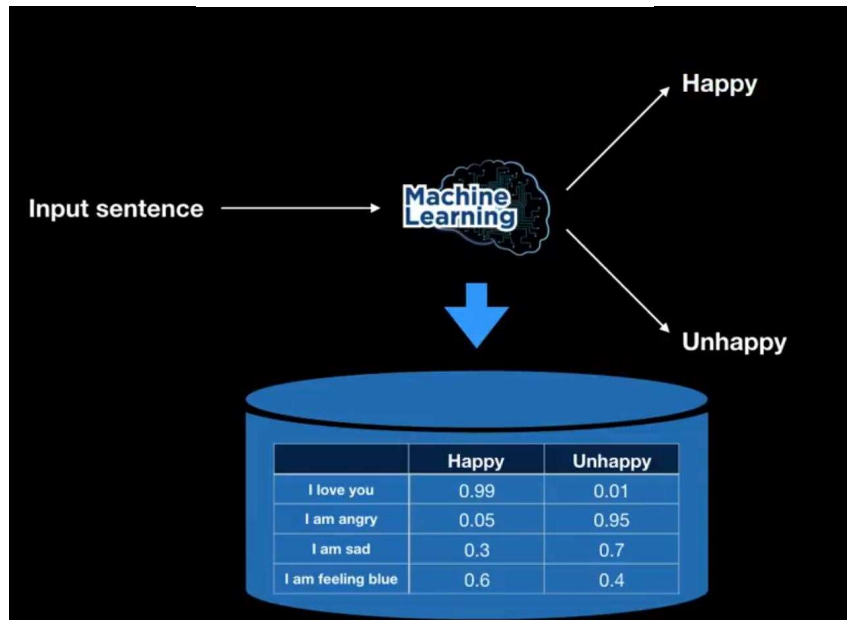
■ 머신러닝에서 엔트로피 사용 예: [Decision Tree \(DT\)](#)

- DT에서 **확실히** 구분이 되는 특징을 먼저 구분해 주는 것이 중요
- 확실히 구분이 되는 특징은 **불확실성(엔트로피)**가 작다는 것을 의미

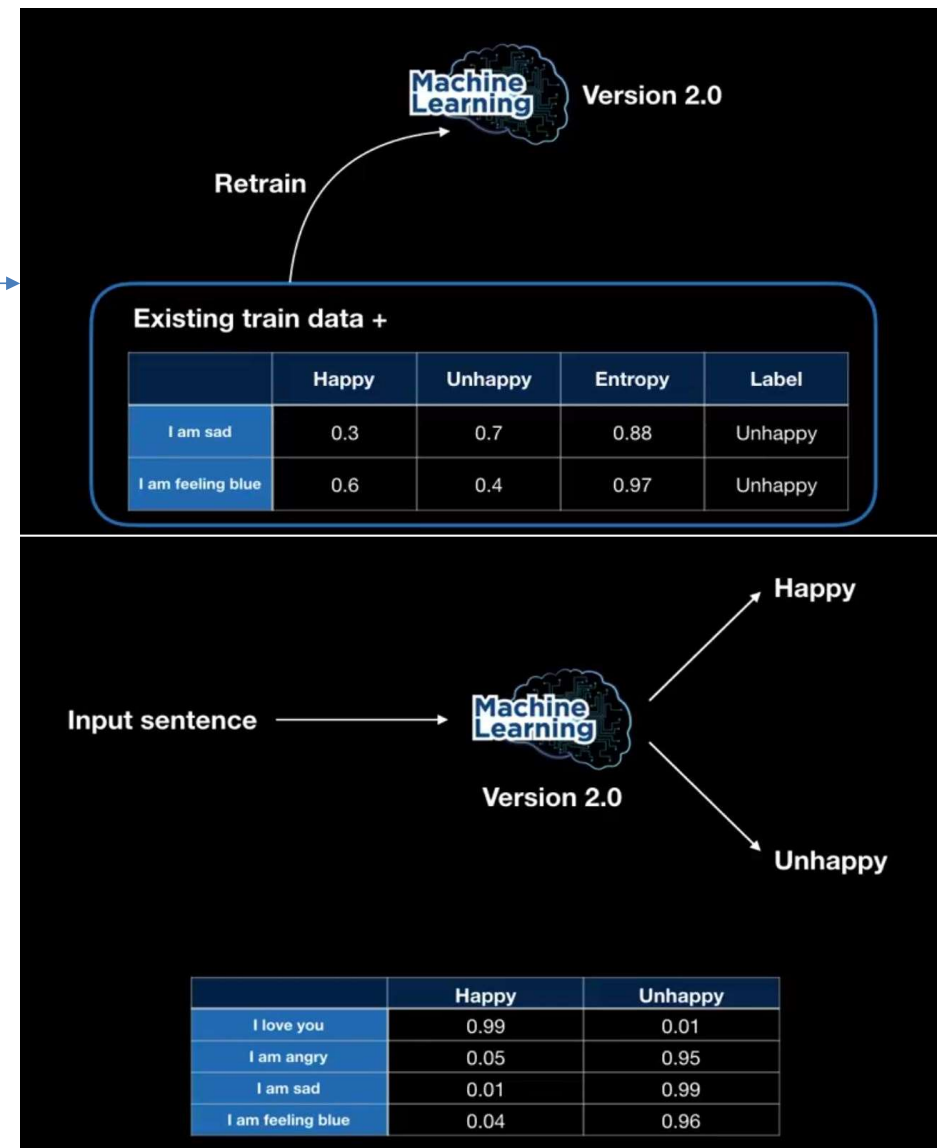


■ 머신러닝에서 엔트로피 사용 예: [Active Learning](#)

$$h(x) = - \sum_{i=1}^n (p_i \log_2(p_i))$$



	Happy	Unhappy	Entropy
I love you	0.99	0.01	0.08
I am angry	0.05	0.95	0.29
I am sad	0.3	0.7	0.88
I am feeling blue	0.6	0.4	0.97



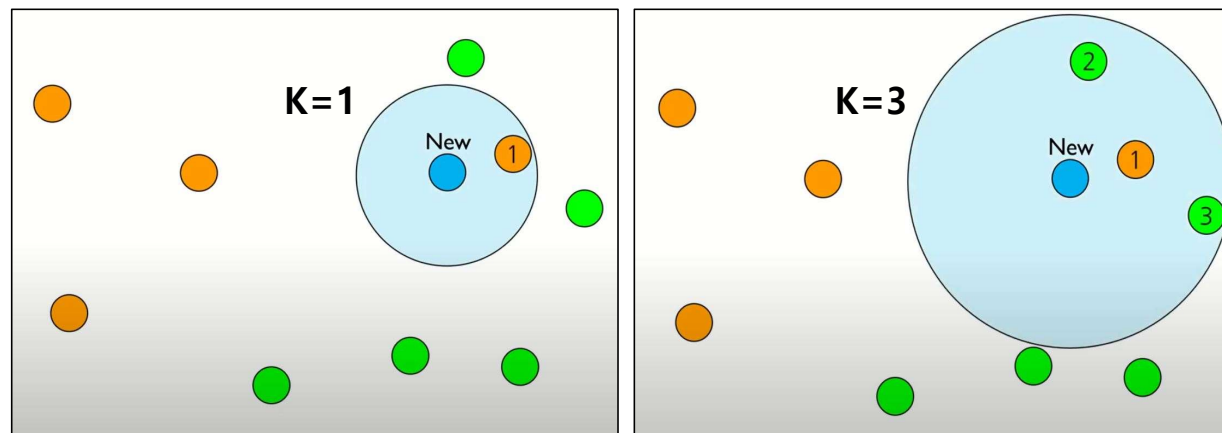
K-NEAREST NEIGHBOR (KNN)

▪ Supervised Learning: Model-based Learning

- Linear/Ridge/Lasso/Elastic Regression
- Deep Learning(MLP & CNN)
- Support Vector Machine
- Decision Tree

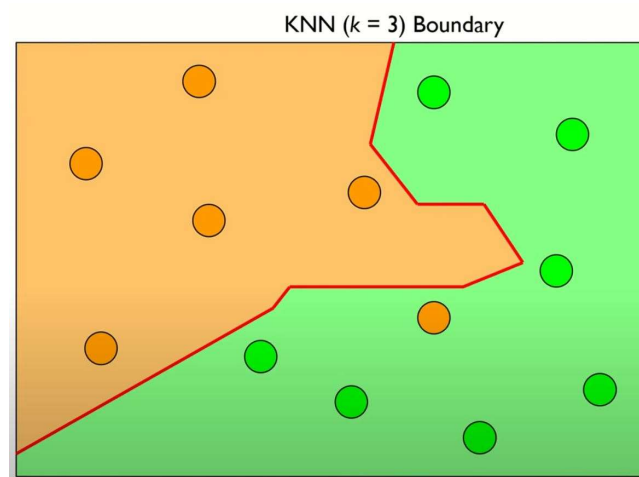
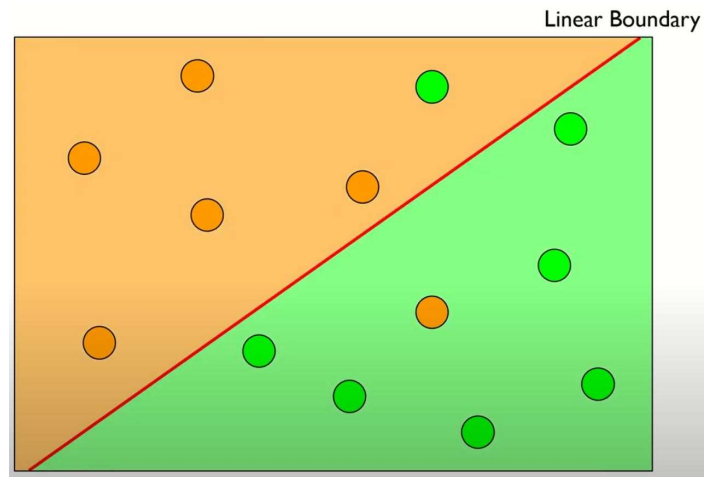
▪ Unsupervised Learning

- KNN Method(or Algorithm): [Memory-based Learning] or [Lazy Learning]



▪ KNN Algorithm

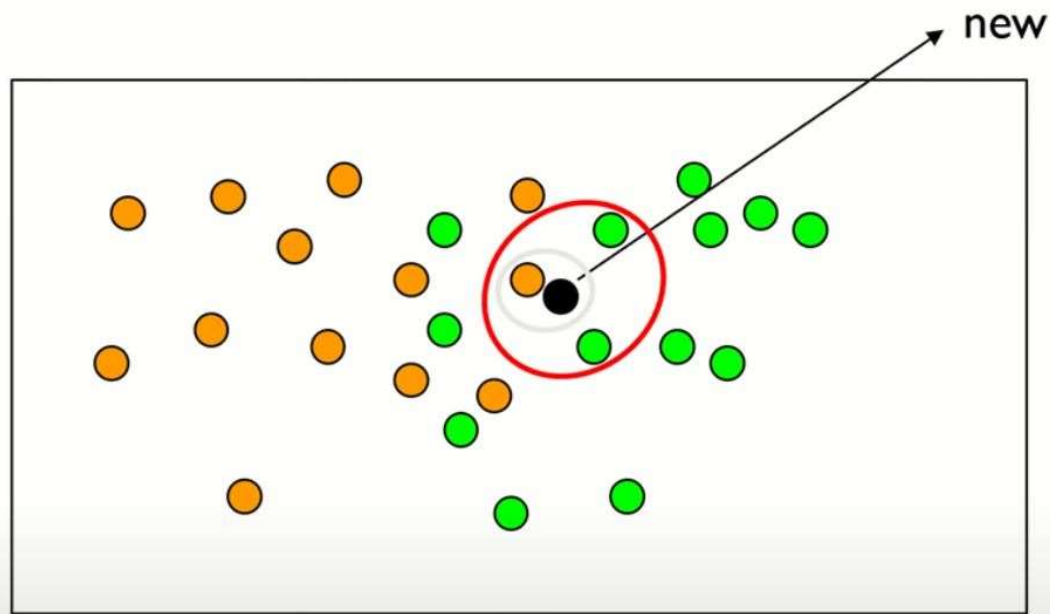
- 선형 vs. 비선형



- KNN 응용: (1) KNN 분류, (2) KNN 추정

▪ KNN 분류

- 인접한 K개의 데이터로부터 **Majority Voting**



$k = \#$ of nearest neighbors

$k = 1$: Orange

$k = 3$: Green

■ KNN 분류

- 인접한 K개의 데이터로부터 **Majority Voting**

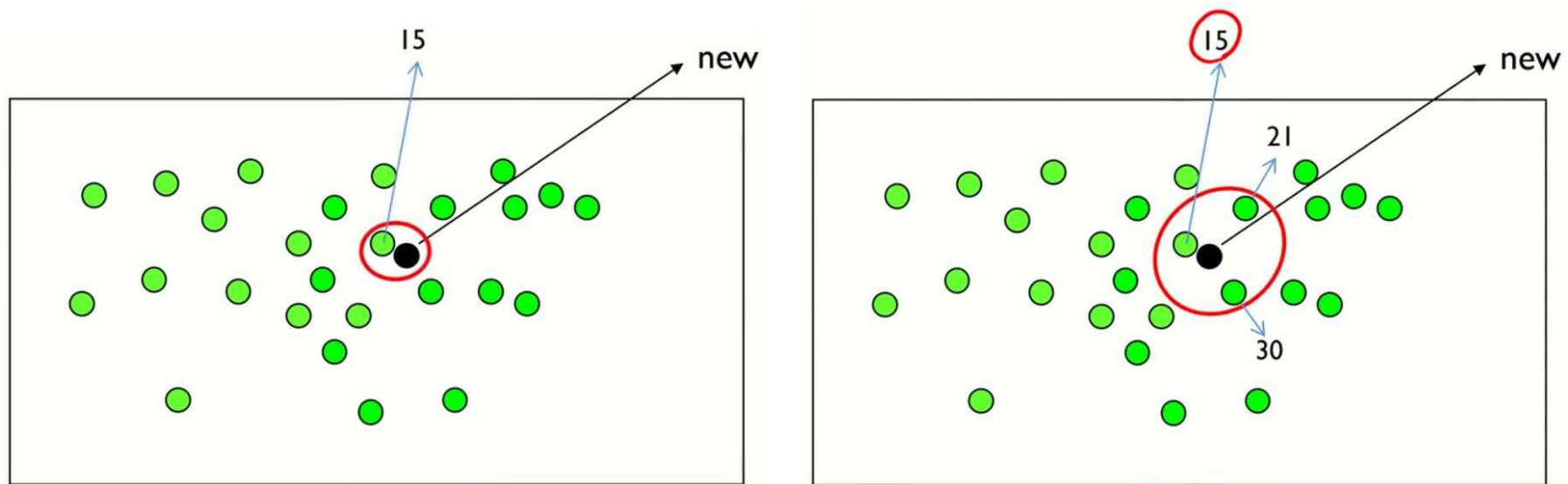
유전자 정보					환자 상태
사람	유전자A	유전자 B	유전자 C	유전자 D	질병유무
A	2.54	4.33	3.99	2.57	정상
B	3.12	3.87	3.84	3.04	정상
C	2.76	4.17	5.63	3.28	정상
D	3.87	3.56	4.25	3.65	질병
E	3.55	3.91	2.68	4.22	질병
F	4.12	2.86	3.30	3.71	질병
G	3.24	3.68	3.82	3.77	?

환자 상태	새로운 관측치와 의 거리
정상	1.54
정상	0.76
정상	2.00
질병	0.78
질병	1.28
질병	1.31

질병	k = 1 : 정상 k = 3 : 질병
----	--------------------------

▪ KNN 예측

- 인접한 K개의 데이터로부터 평균/중간값/Min/Max/ 중에서 택일



k = number of nearest neighbors

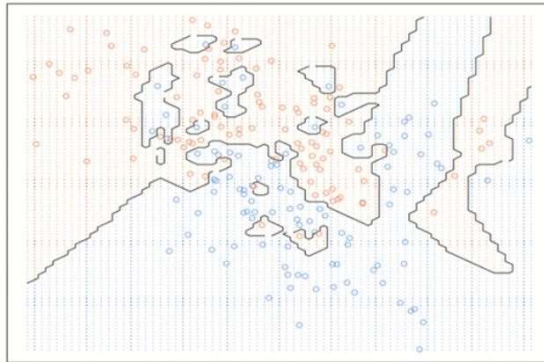
$k = 1$: new = 15

$k = 3$: new = $(15+21+30)/3 = 22$

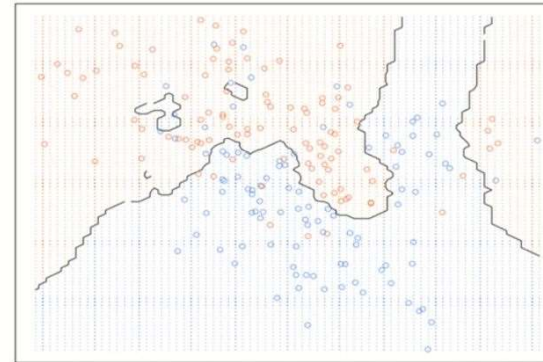
▪ KNN Algorithm 이슈

- [1] 최적의 K를 어떻게 결정할 것인가? → 인접한 학습 데이터를 몇 개까지 탐색할 것인가?

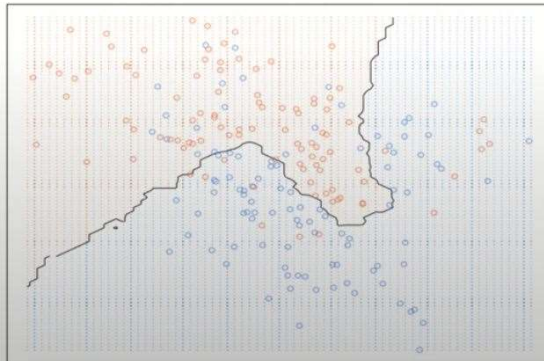
($1 \leq K \leq$ 전체 데이터 개수 → Overfitting vs Underfitting)



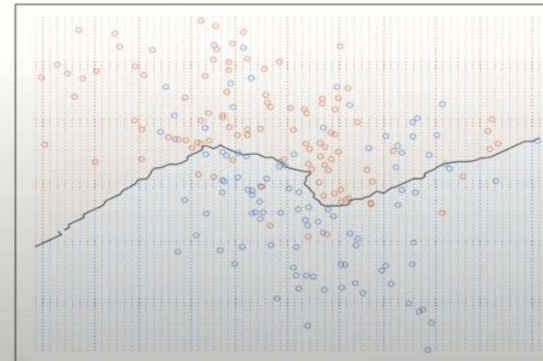
1-nearest neighbor



5-nearest neighbor



15-nearest neighbor



50-nearest neighbor

▪ KNN Algorithm 이슈

- [1] 최적의 K를 어떻게 결정할 것인가? (인접한 학습 데이터를 몇 개까지 탐색할 것인가?)

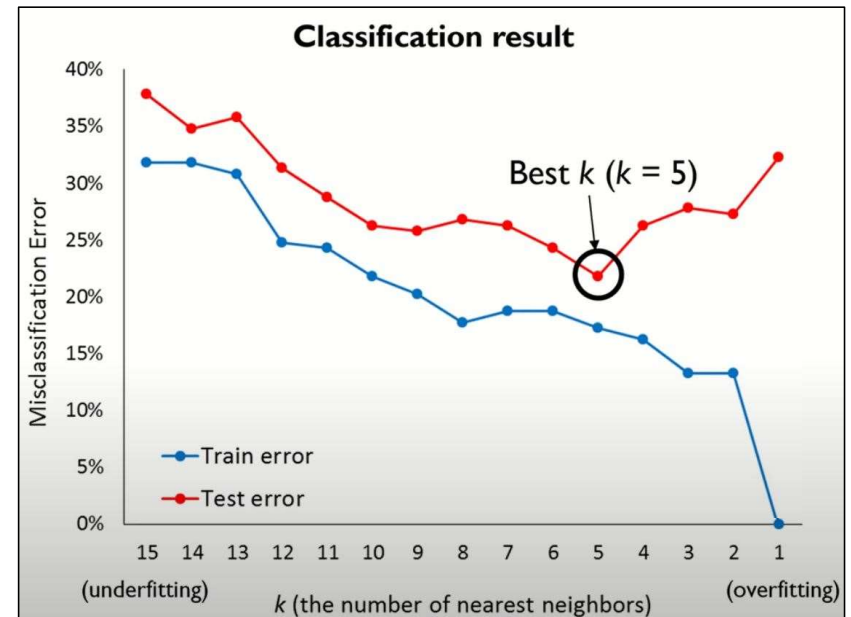
• 분류모델

$$MisclassError_k = \frac{1}{k} \sum_{i=1}^k I(c_i \neq \hat{c}_i) \text{ for } k = 1, 2, \dots, k^*$$

$I(\cdot)$: Indicator Function

• 예측모델

$$SSE_k = \sum_{i=1}^k (y_i - \hat{y}_i)^2 \text{ for } k = 1, 2, \dots, k^*$$



▪ KNN Algorithm 이슈

- [2] 데이터간 거리는 어떻게 측정할 것인가? (Distance Measurements)

- **L1 Norm (Manhattan Distance)** $d_{Manhattan}(X,Y) = \sum_{i=1}^n |x_i - y_i|$
- **L2 Norm (Euclidean Distance)** $d_{(A,B)} = \sqrt{(a_1 - b_1)^2 + \dots + (a_p - b_p)^2} = \sqrt{\sum_{i=1}^p (a_i - b_i)^2}$
- **Mahalanobis Distance** $d_{Mahalanobis}(X,Y) = \sqrt{(X - Y)^T \Sigma^{-1} (X - Y)},$
- **Correlation Distance** Σ^{-1} : inverse of covariance matrix

$$d_{Corr}(X,Y) = 1 - r$$

$$\text{where } r = \sigma_{XY}$$

