학습(Learning)과 모델(model)

개요

- 머신러닝에서 학습이란?

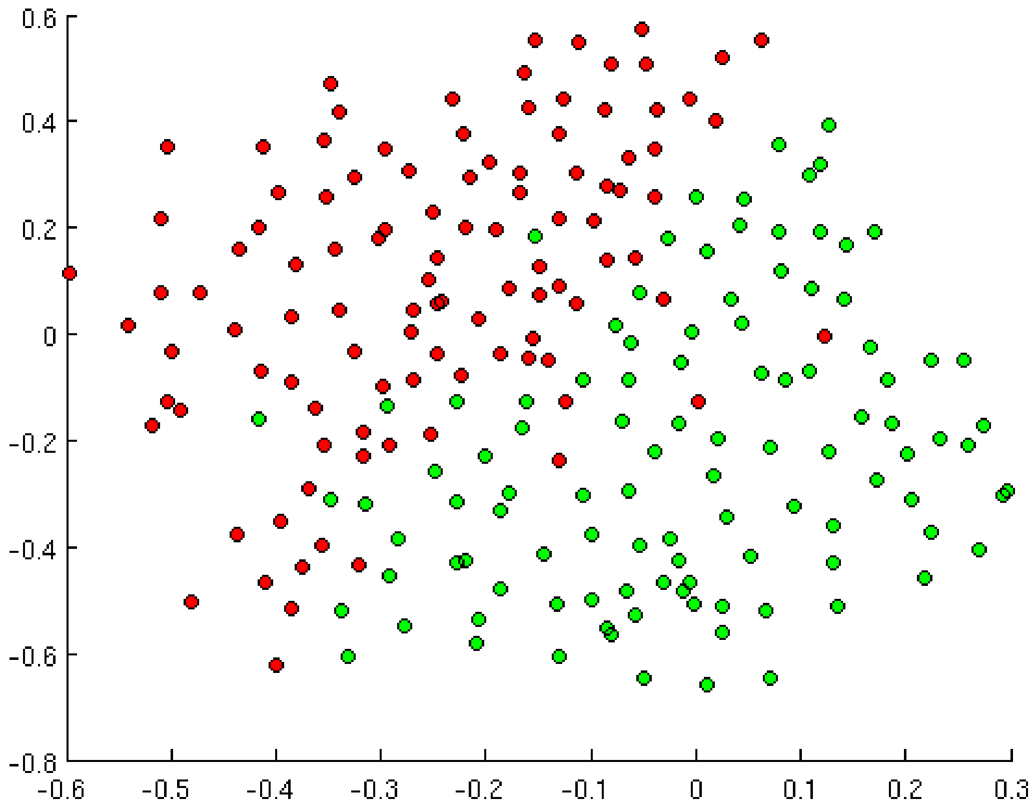
- 머신러닝에서 모델이란?

- 학습 후에 모델이 얼마나 성능이 좋은지 판단

기계학습(머신러닝, Machine Learning)이란?

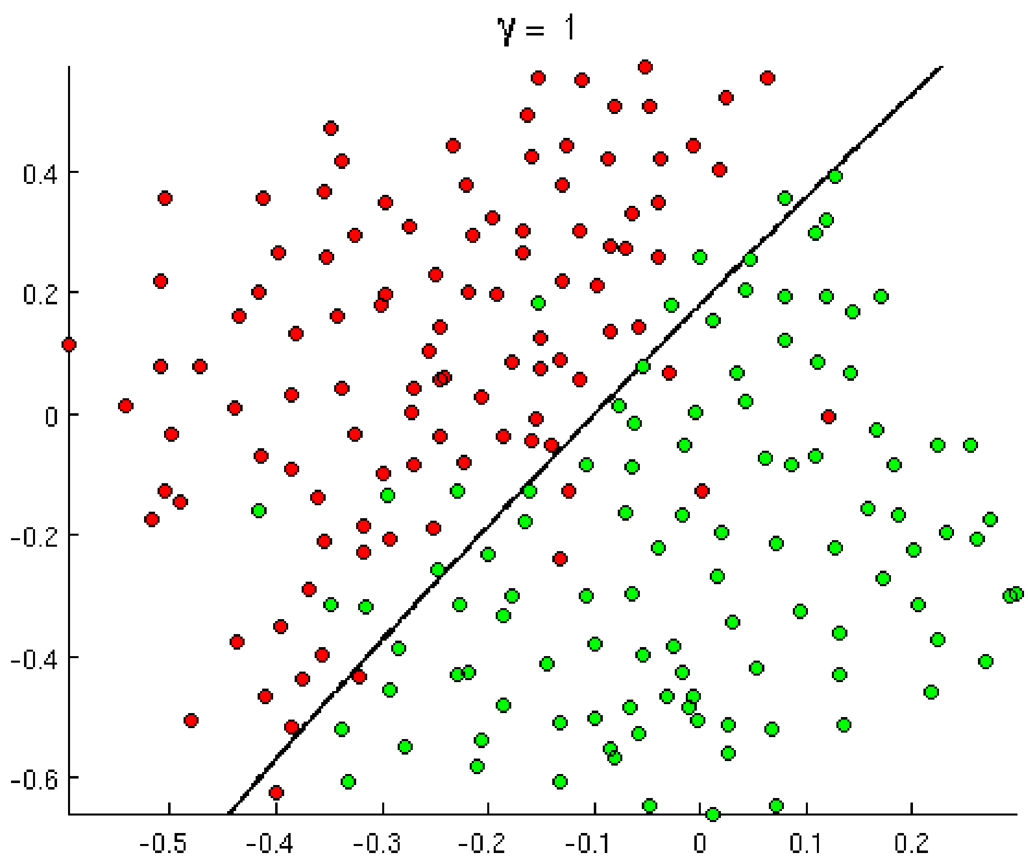
예제 : 이진 분류(Binary Classification)

- 붉은 점과 녹색 점을 어떻게 사람은 분류할까?



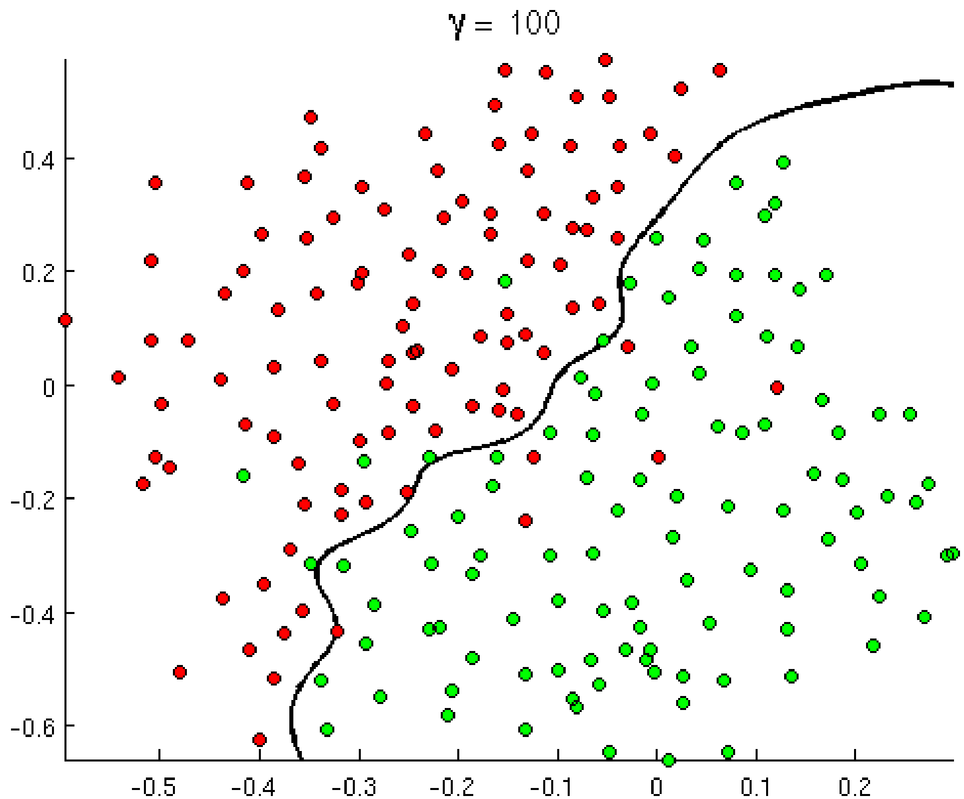
예제 : 이진 분류(Binary Classification)

- 붉은 점과 녹색 점을 어떻게 사람은 분류할까?



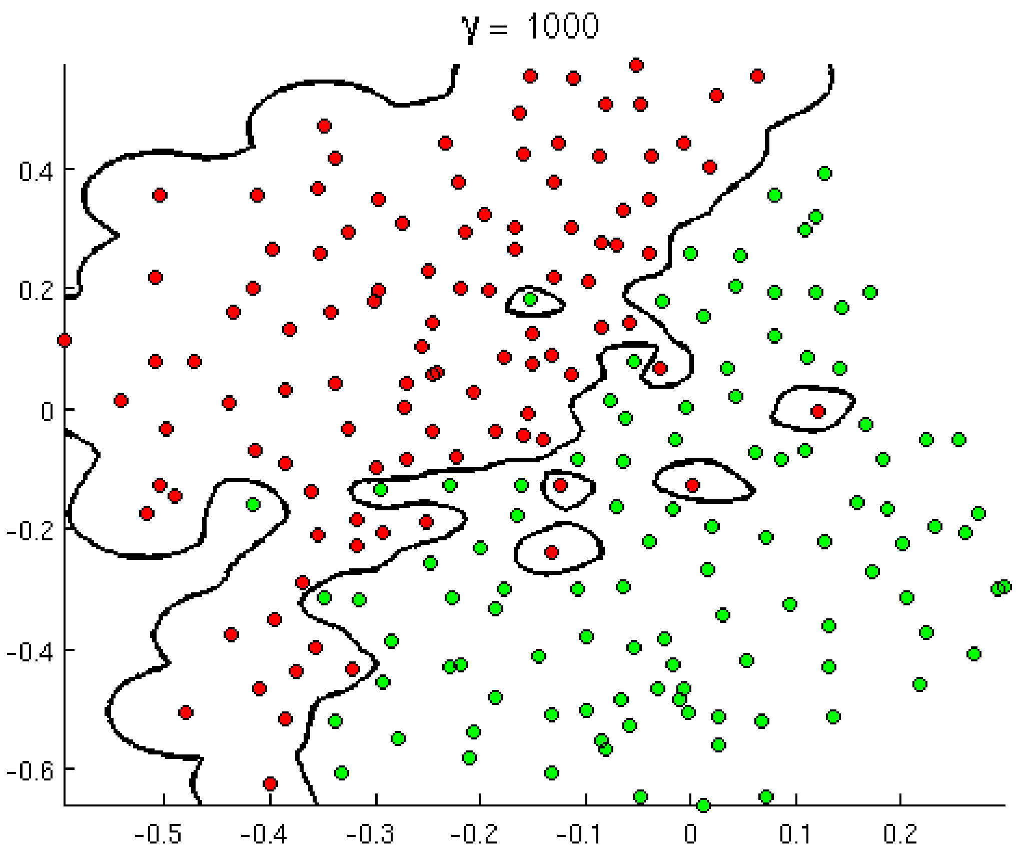
예제 : 이진 분류(Binary Classification)

- 붉은 점과 녹색 점을 어떻게 사람은 분류할까?



예제 : 이진 분류(Binary Classification)

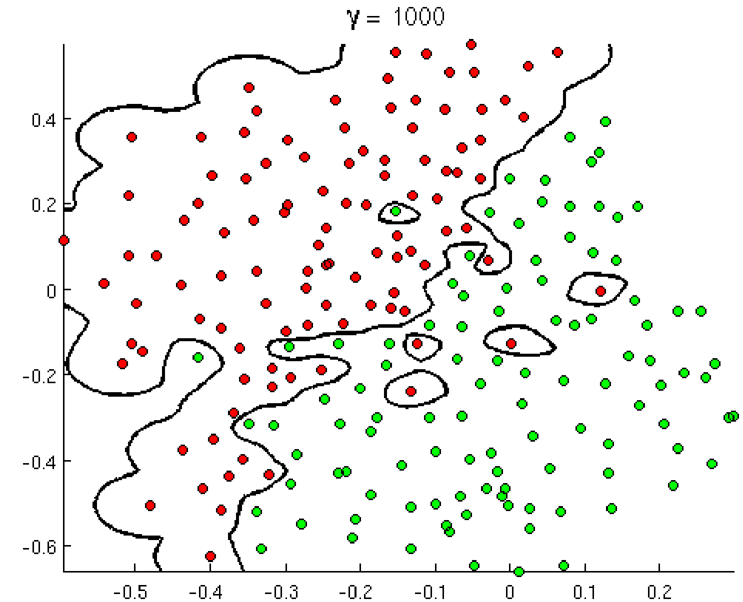
- 붉은 점과 녹색 점을 어떻게 사람은 분류할까?

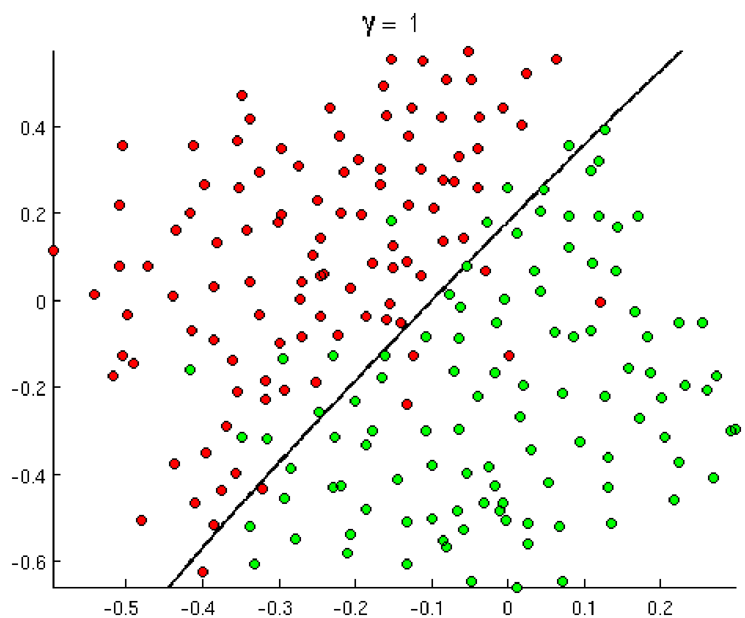


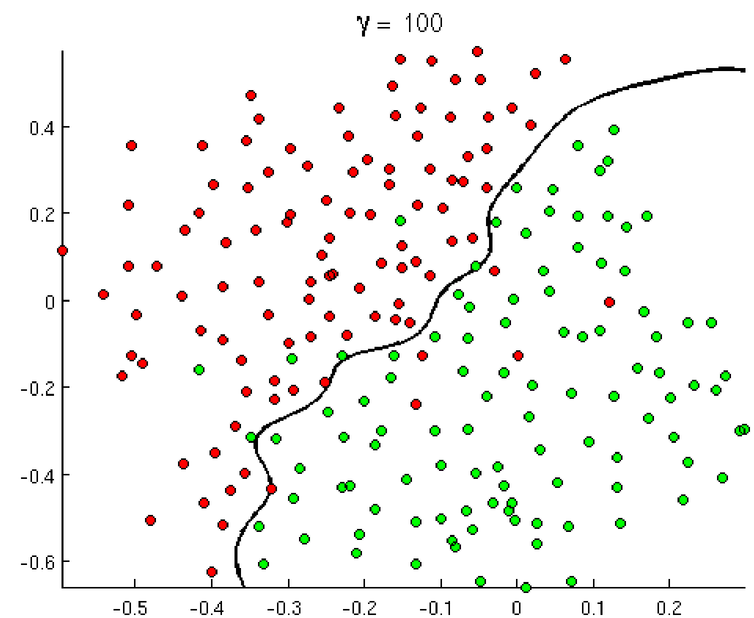
예제 : 이진 분류(Binary Classification)

어떤 분류가 정답인가?

성능에 따라 정답이 다르다.

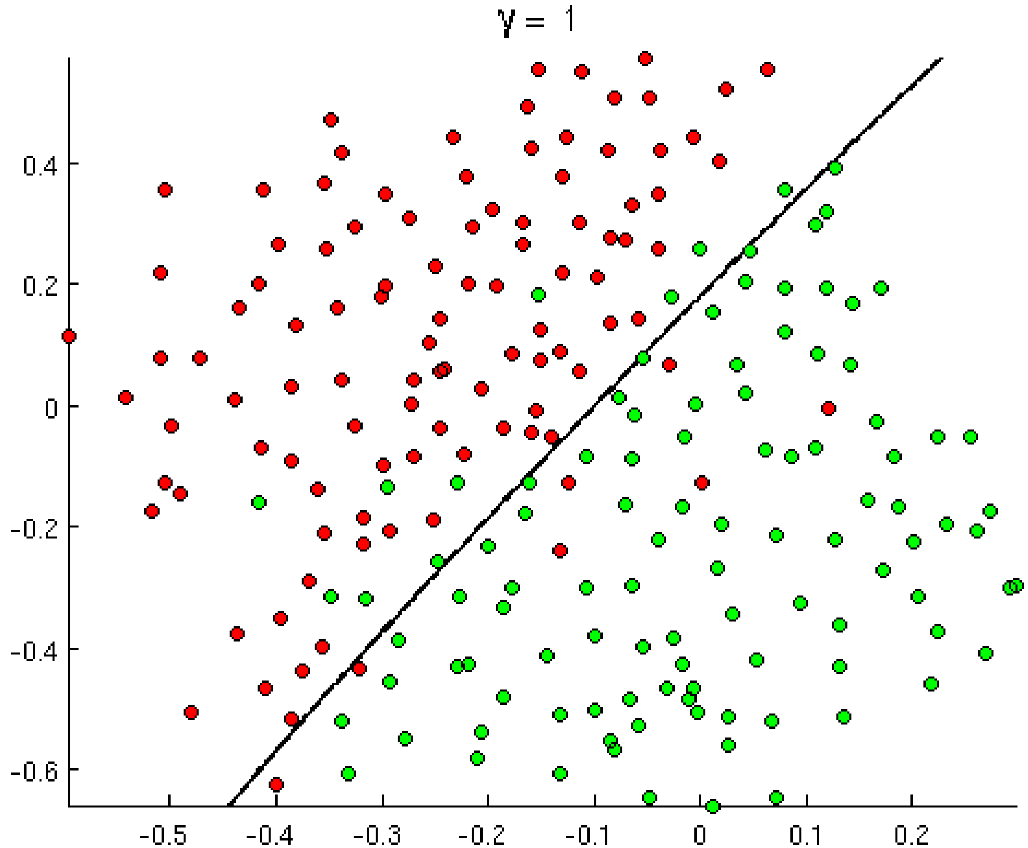






모델(Model)

선형(직선) = aX + b

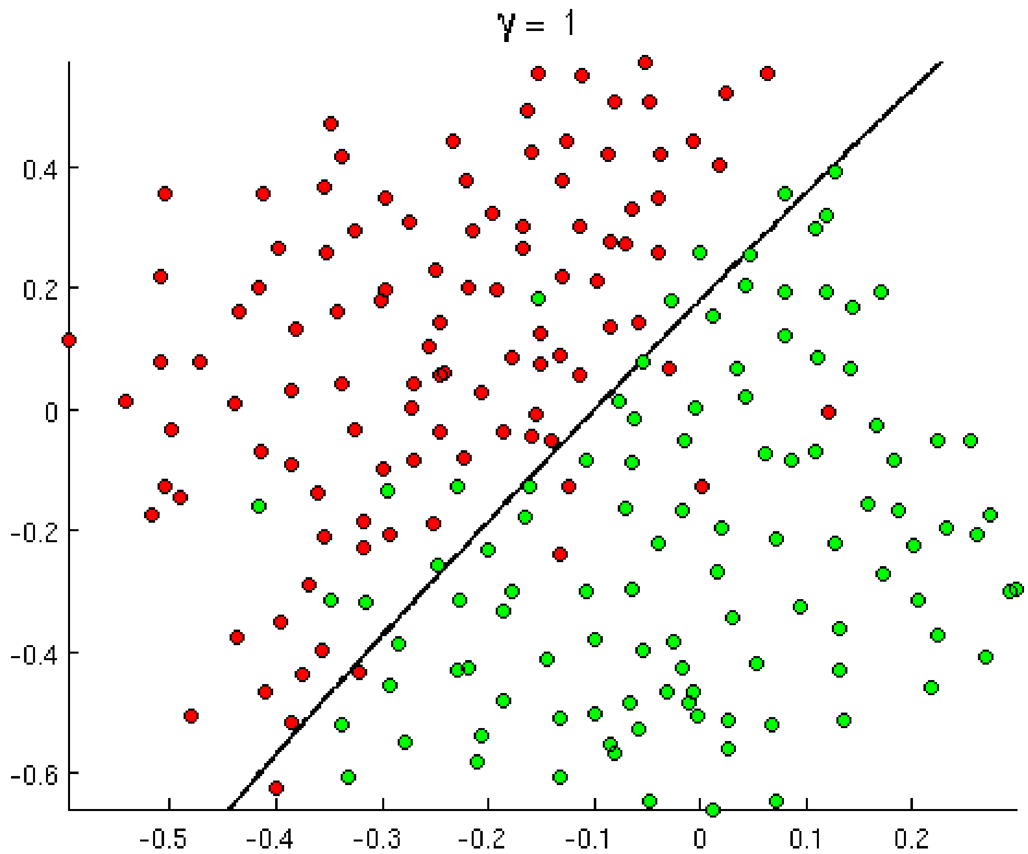


학습(Learning)

즉 직선의기울기와 y절편을 구하는 것
이것이 딥러닝의 정수.

선형(직선) = aX + b

a, b 등 모델을 결정하는 변수 값을 찾는 과정



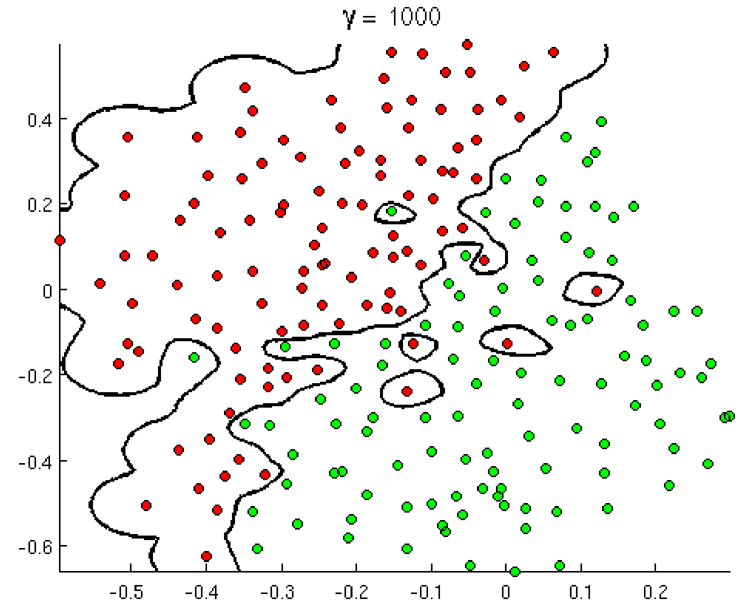
모델(Model)

선형(직선) = aX + b

다차원 곡선 = a X^n + … + b X + c

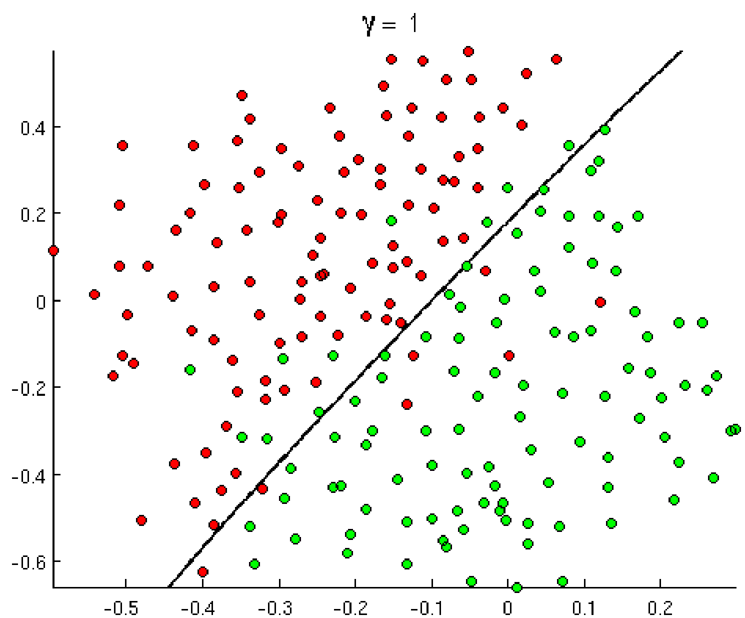
비선형적 곡면 = ….

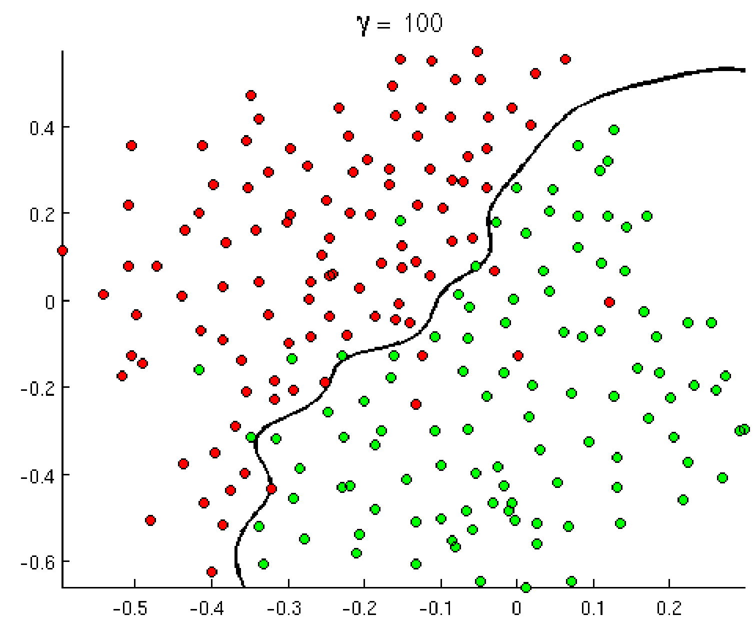
…



사용자가 모델을 선택해 주어야 함

선택한 모델의 파라메타 값을 찾아내는 과정이 "학습"





학습(Learning)

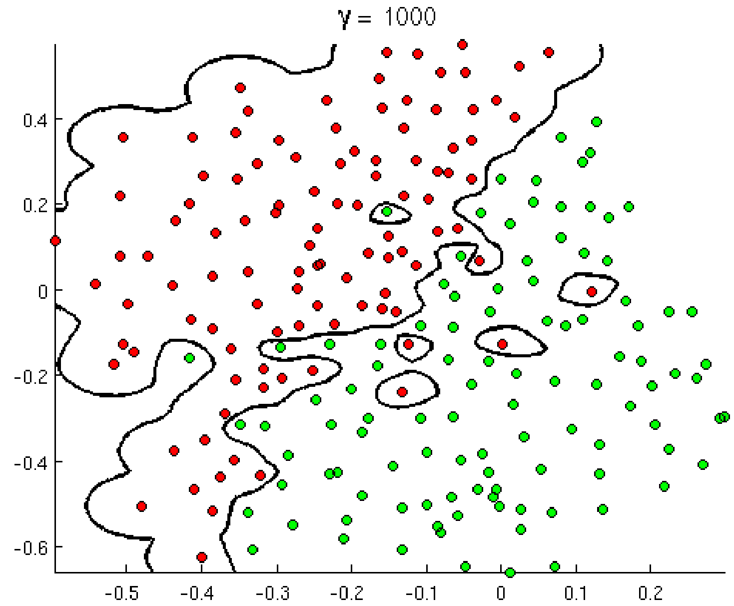
선형(직선) = aX + b

다차원 곡선 = a X^n + … + b X + c

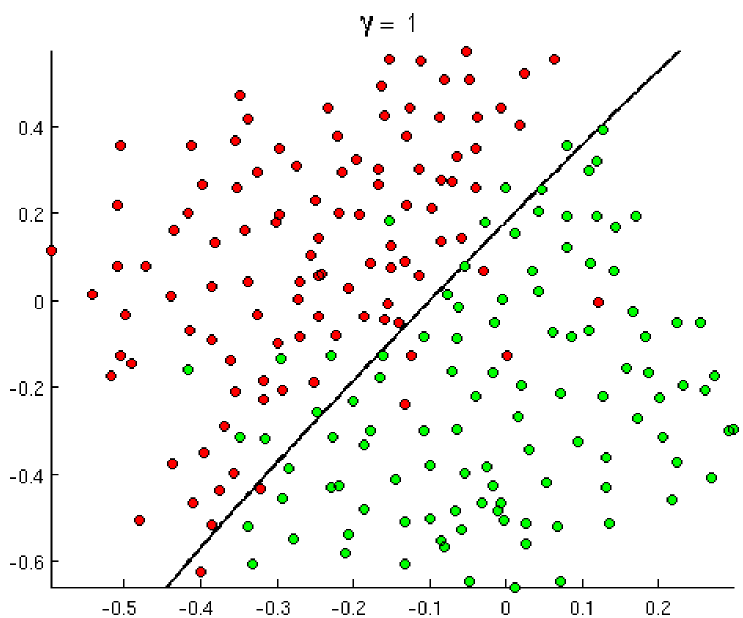
비선형적 곡면 = ….

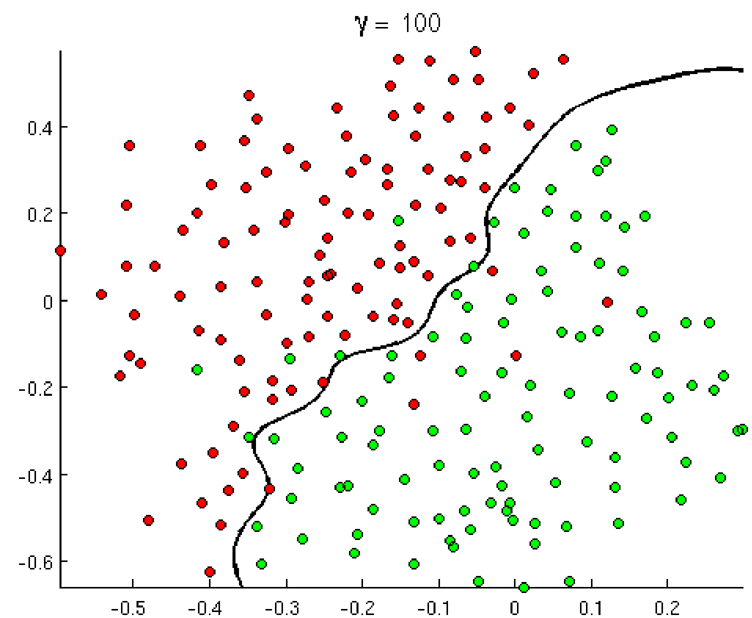
…

a, b, c 등 모델을 결정하는 변수 값을 찾는 과정



파라메타들이 학습을 함.

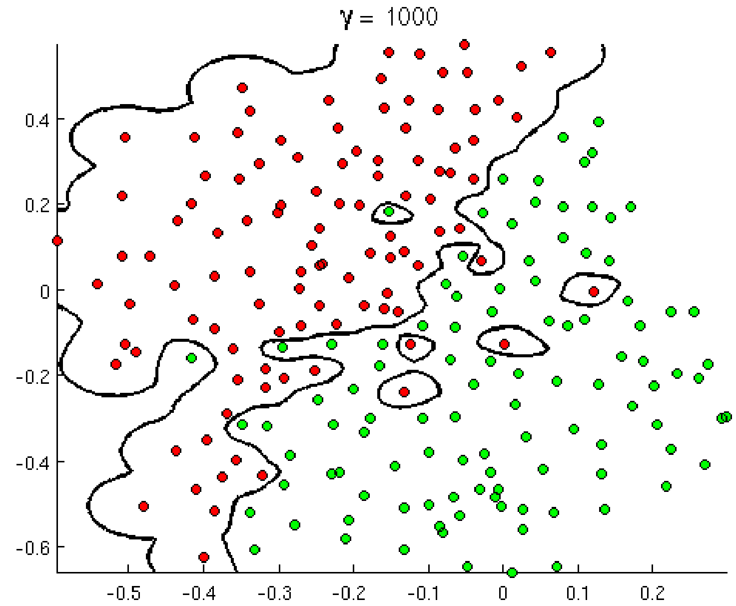


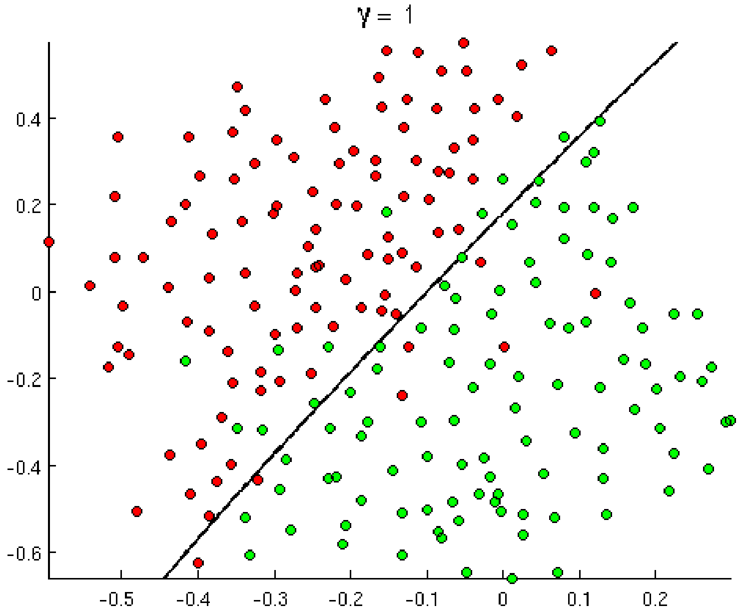


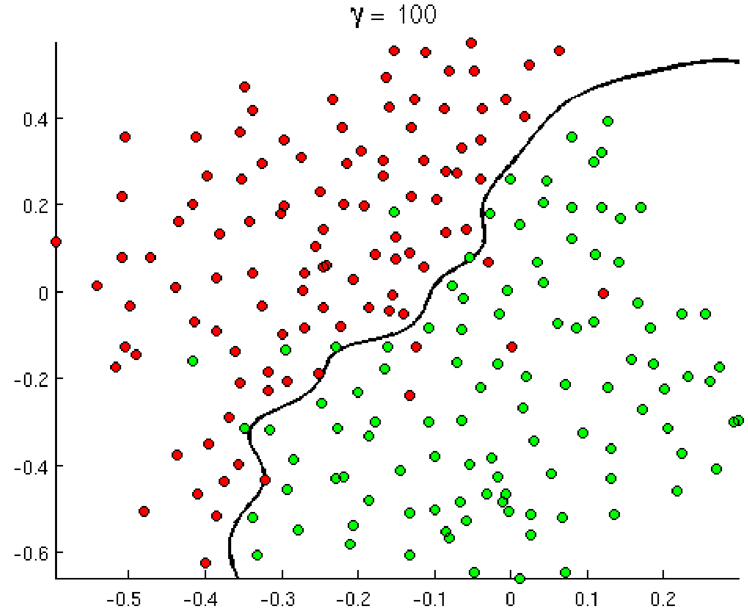
학습(Learning)

a, b, c 등 모델을 결정하는 변수 값을 찾는 과정

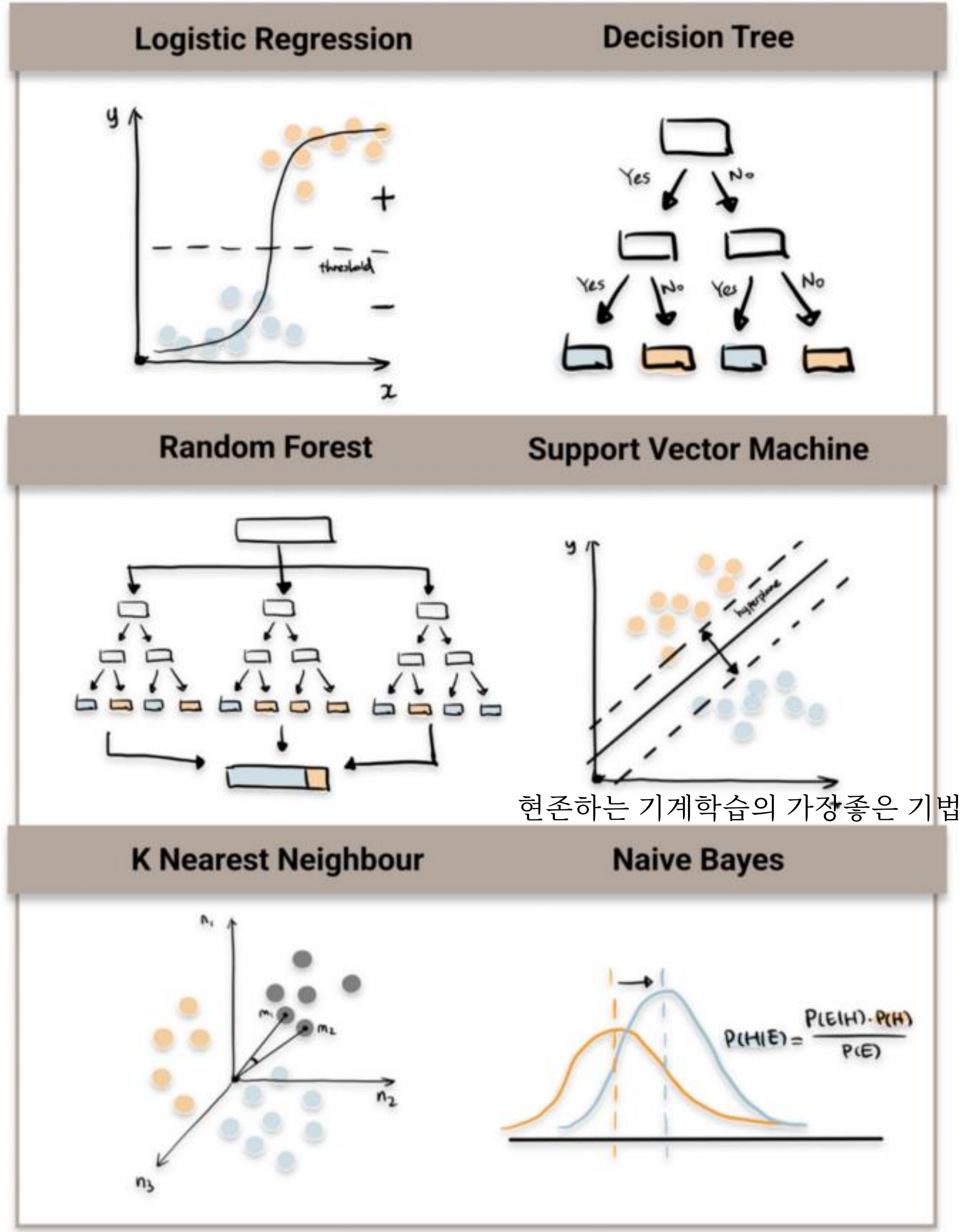
최적의 변수? 최적에 대한 objective function 정의 필요하고, 해당 문제를 풀 수 있어야 함







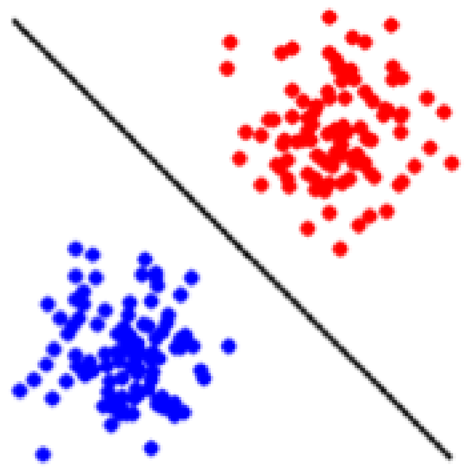
Classification 기법들



Linear Regression (선형 회귀)

가장 공통점이 많게 선을 긋는 행위?

- **선형 분류기**

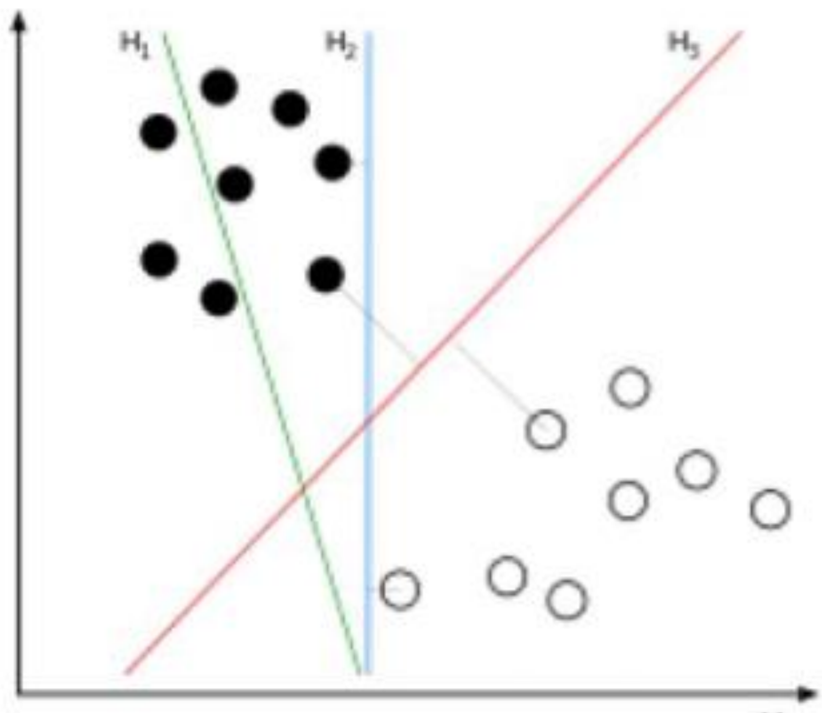


- **예를 들면, x 가 집합1(빨간색)이면 wx > 0**
- **x 가 집합2(파란색)이면 wx < 0**

Support Vector Machine (SVM)

- SVM은 Logistic Regression(LR)과 클래스를 분류한다는데 유사하지만, LR과 다르게 확률 값을 제공하지 않음

- 최적화를 통해 최대 간극(margin)을 보장

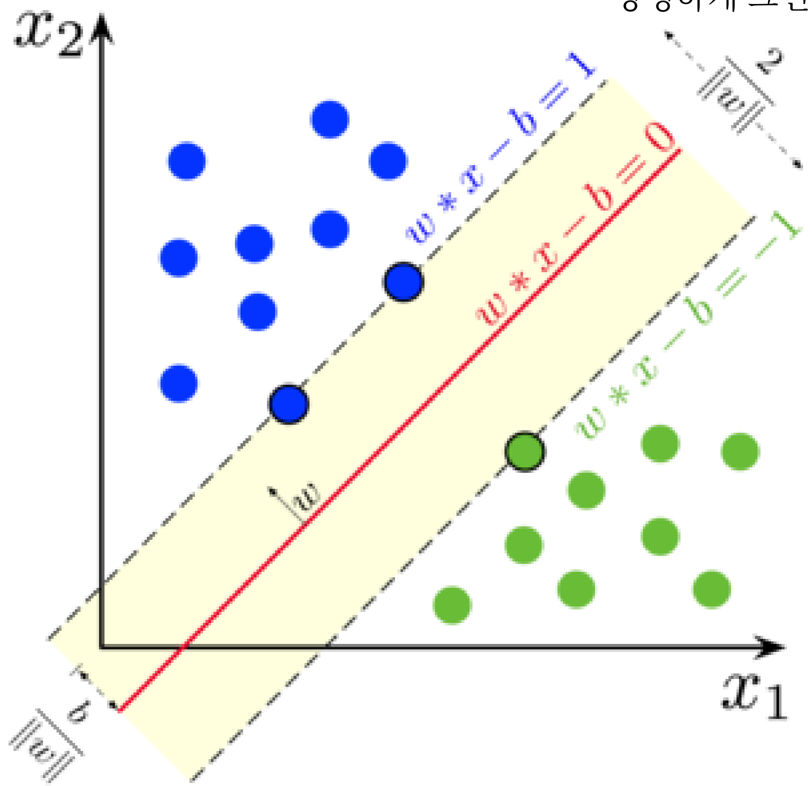


Support Vector Machine (SVM)

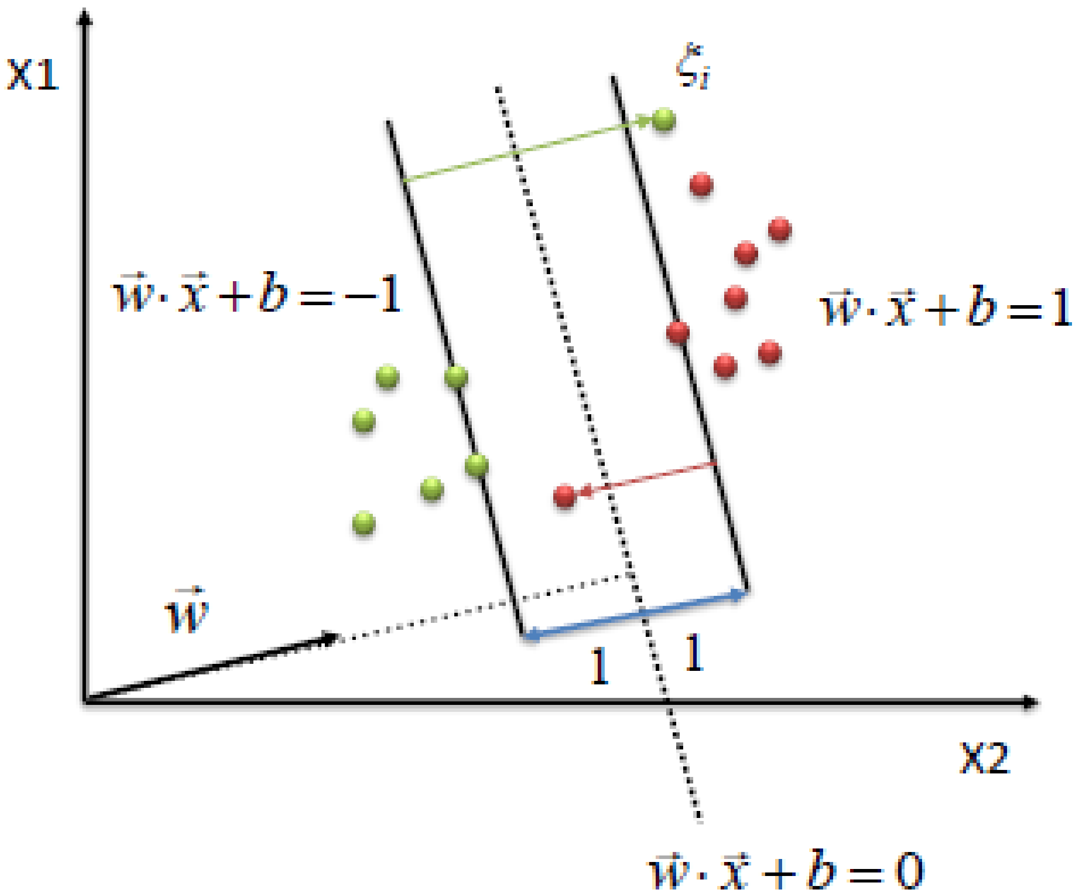
- SVM은 Logistic Regression(LR)과 클래스를 분류한다는데 유사하지만, LR과 다르게 확률 값을 제공하지 않음

- 최적화를 통해 최대 간극(margin)을 보장

평행하게 그린 양 경계선 간 간격 폭이 가장 넓은 직선



Support Vector Machine (SVM)



약간의 오류데이터는 허용

Constraint becomes :

$$y_i(w\cdot x_i+b)\geq 1-\xi_i,\ \forall x_i$$
$$\xi_i\geq 0$$

에러

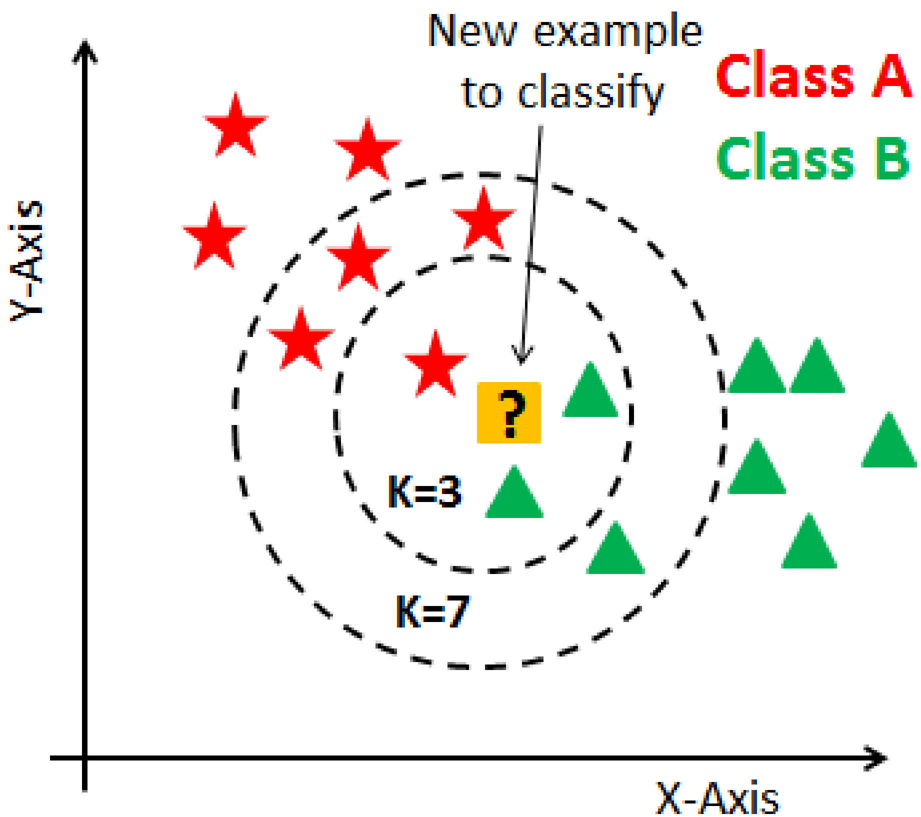
Objective function penalizes for misclassified instances and those within the margin

$$\min \frac{1}{2}\|w\|^2+C\sum_i \xi_i$$

C trades-off margin width and misclassifications

k Nearest Neighbor (k-NN)

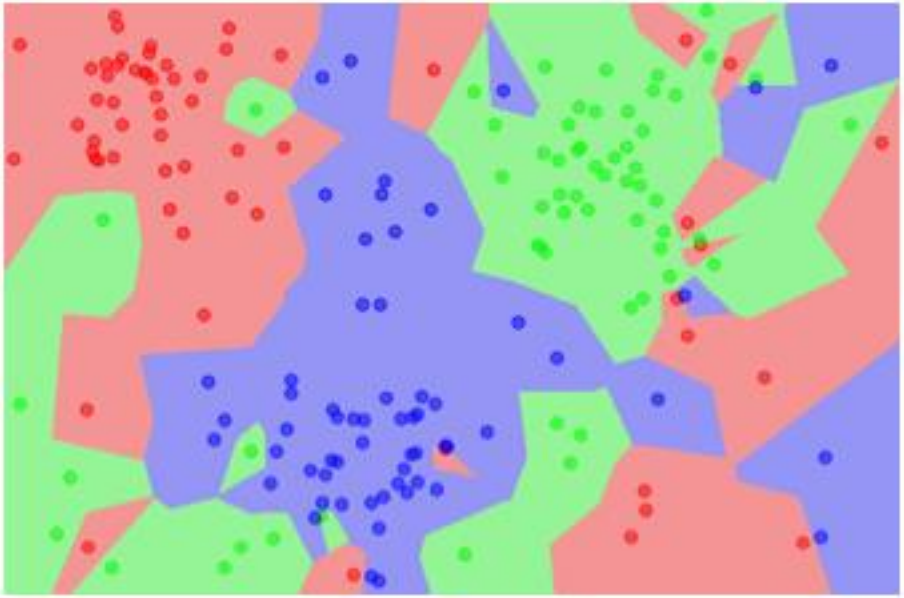
- k개의 가장 가까운 점들을 찾아서, 해당 점들의 분류결과 평균으로 분류



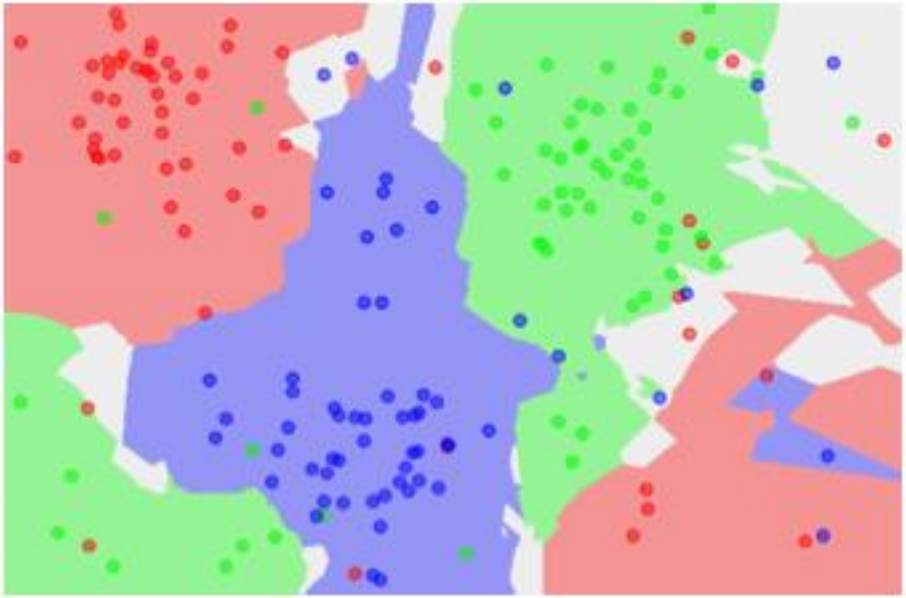
k Nearest Neighbor (k-NN)

- **k개의 가장 가까운 점들을 찾아서, 해당 점들의 분류결과 평균으로 분류**

미리 계산을 해놓아 전처리과정으로 바로바로 나올 수 있게



k = 1인 경우



k = 5인 경우

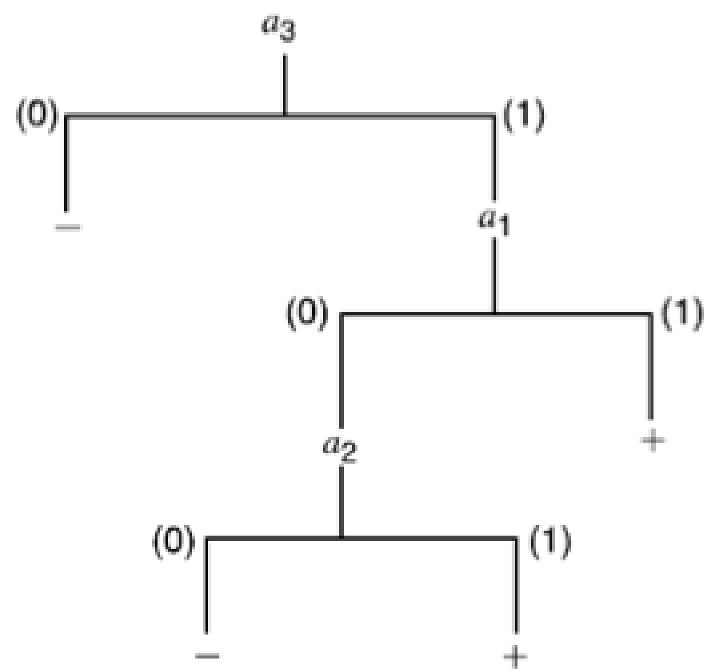
Decision Tree (결정 트리)

- "스무고개"놀이와 결정 트리

- 3가지 속성을 가진 데이터로 각 속성은 0, 1로 표현되고, 목적 클래스가 +, - 로 분류되는 데이터가 있을 경우

a3부터 확인해 -인지 +인지 판별
즉 a3가 0이면 항상 -이므로 이거로 분류

No.	a_1	a_2	a_3	C
1	0	0	0	−
2	0	0	1	−
3	0	1	0	−
4	0	1	1	+
5	1	0	0	−
6	1	0	1	+
7	1	1	0	−
8	1	1	1	+



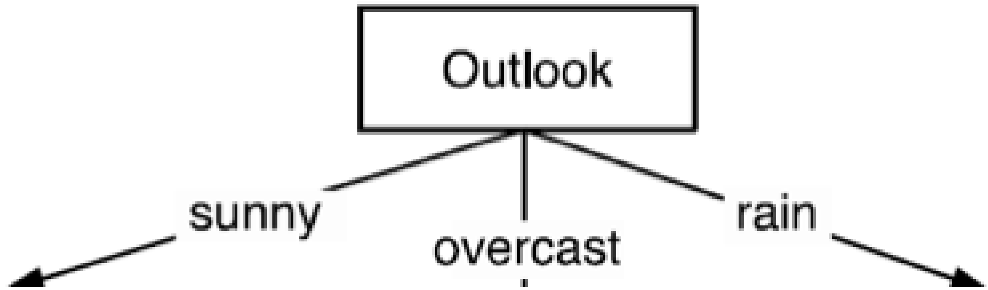
Decision Tree (결정 트리)

- **야구 경기가 열린 날의 날씨 데이터**

Day	Outlook	Temperature	Humidity	Wind	Playball
D1	sunny	hot	high	weak	no
D2	sunny	hot	high	strong	no
D3	overcast	hot	high	weak	yes
D4	rain	mild	high	weak	yes
D5	rain	cool	normal	weak	yes
D6	rain	cool	normal	strong	no
D7	overcast	cool	normal	weak	yes
D8	sunny	mild	high	weak	no
D9	sunny	cool	normal	weak	yes
D10	rain	mild	normal	strong	yes
D11	sunny	mild	normal	strong	yes
D12	overcast	mild	high	strong	yes
D13	overcast	hot	normal	weak	yes
D14	rain	mild	high	strong	no

Decision Tree (결정 트리)

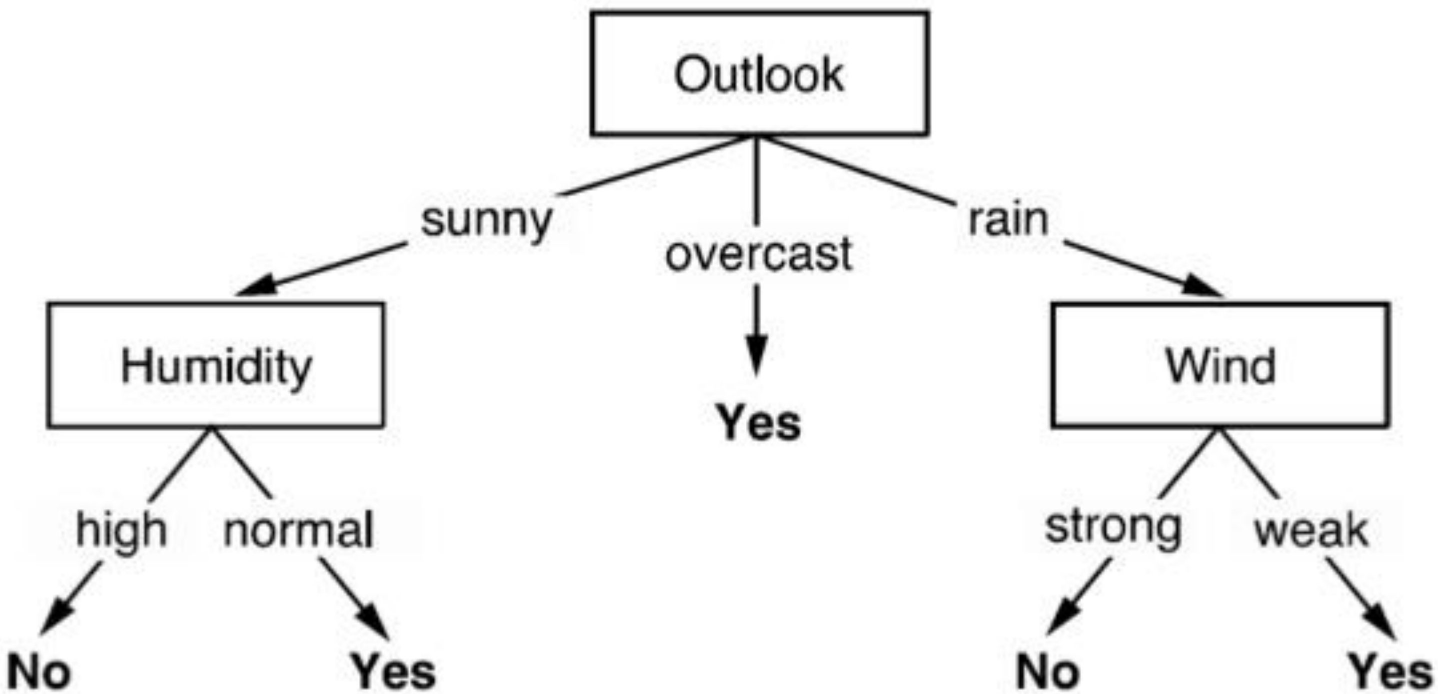
Outlook을 맨 처음 노드로 선택한 후 다음 노드의 선택은?



Decision Tree (결정 트리)

Outlook을 맨 처음 노드로 선택한 후 다음 노드의 선택은?

일반적으론 모든 질문에 다 확인해야되나 최적화를 하면 몇개의 질문으로도 그냥 나눌 수 있다.

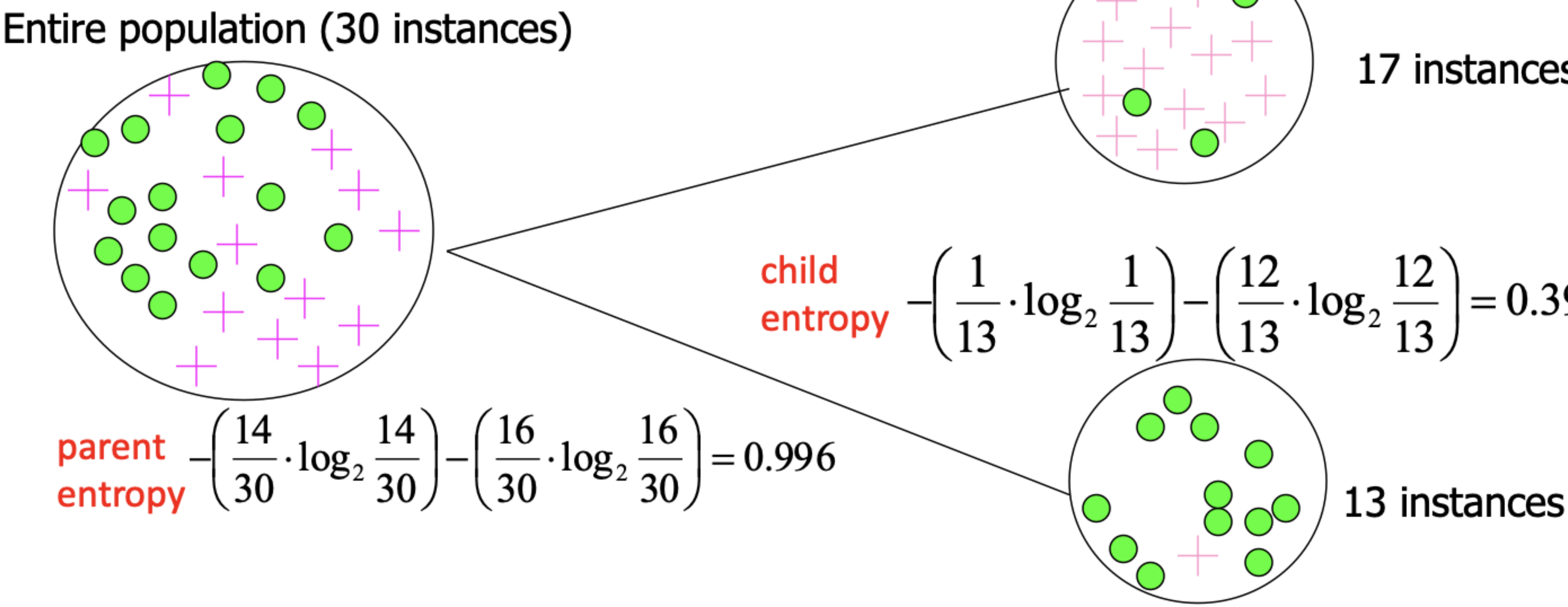


$$H(X) = -\sum_{i=1}^{n} P(X=i)\log_2 P(X=i)$$

Calculating Information Gain

Information Gain = entropy(parent) – [average entropy(children)]

child entropy $-\left(\frac{13}{17}\cdot\log_2\frac{13}{17}\right)-\left(\frac{4}{17}\cdot\log_2\frac{4}{17}\right)=0.787$



child entropy $-\left(\frac{1}{13}\cdot\log_2\frac{1}{13}\right)-\left(\frac{12}{13}\cdot\log_2\frac{12}{13}\right)=0.391$

parent entropy $-\left(\frac{14}{30}\cdot\log_2\frac{14}{30}\right)-\left(\frac{16}{30}\cdot\log_2\frac{16}{30}\right)=0.996$

(Weighted) Average Entropy of Children = $\left(\frac{17}{30}\cdot 0.787\right)+\left(\frac{13}{30}\cdot 0.391\right)=0.615$

Information Gain= 0.996 - 0.615 = 0.38

엔트로피가 낮을수록 I.G는 증가

Decision Tree (결정 트리)

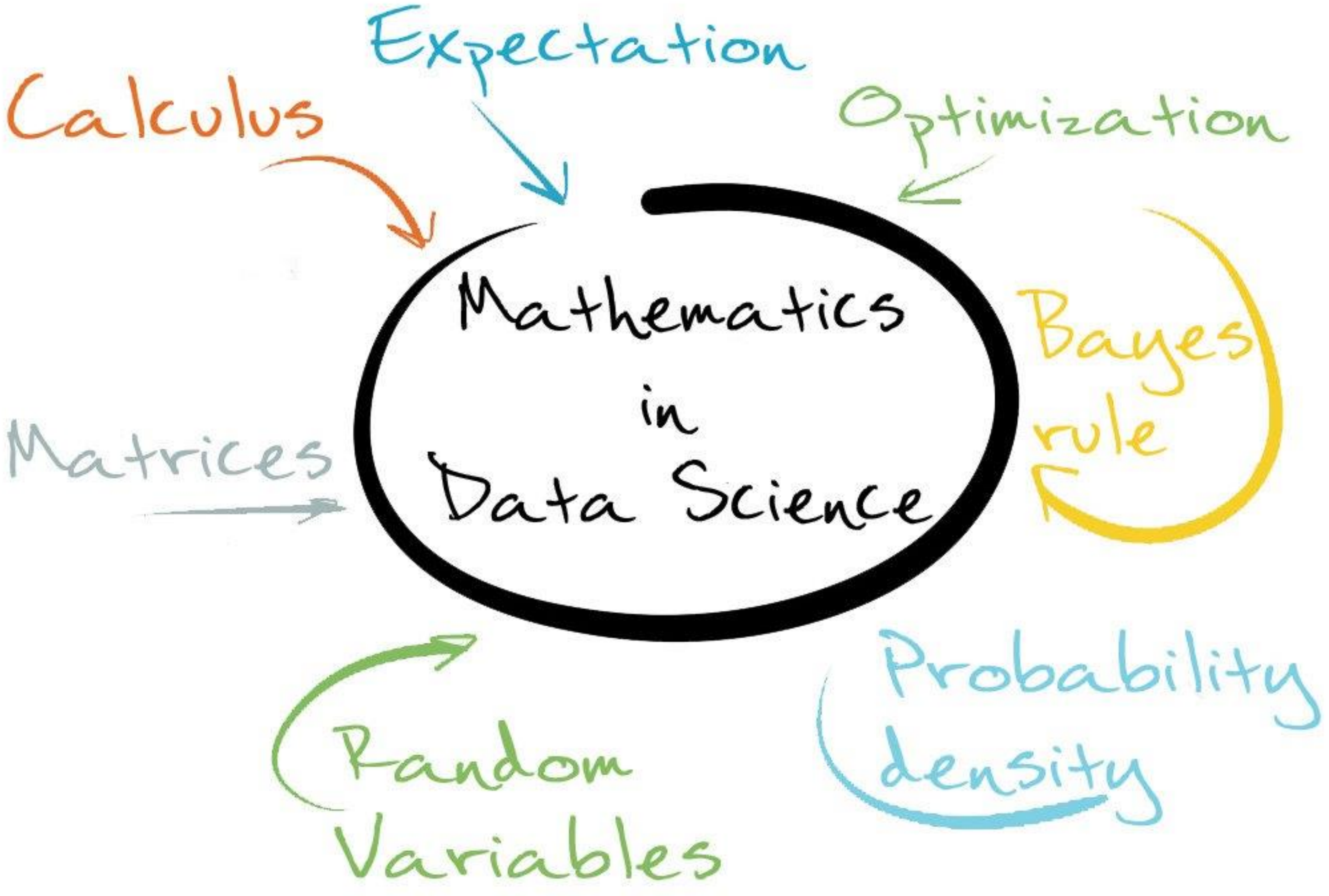
- 다른 속성도 마찬가지로 Gain을 기준으로 선택
- Outlook이 sunny인 경우에는 D_sunny = {D1, D2, D8, D9, D11}

$$Gain(D_{sunny}, Humidity) = 0.970$$

$$Gain(D_{sunny}, Temperature) = 0.570$$

$$Gain(D_{sunny}, Wind) = 0.019$$

Day	Outlook	Temperature	Humidity	Wind	Playball
D1	sunny	hot	high	weak	no
D2	sunny	hot	high	strong	no
D3	overcast	hot	high	weak	yes
D4	rain	mild	high	weak	yes
D5	rain	cool	normal	weak	yes
D6	rain	cool	normal	strong	no
D7	overcast	cool	normal	weak	yes
D8	sunny	mild	high	weak	no
D9	sunny	cool	normal	weak	yes
D10	rain	mild	normal	strong	yes
D11	sunny	mild	normal	strong	yes
D12	overcast	mild	high	strong	yes
D13	overcast	hot	normal	weak	yes
D14	rain	mild	high	strong	no

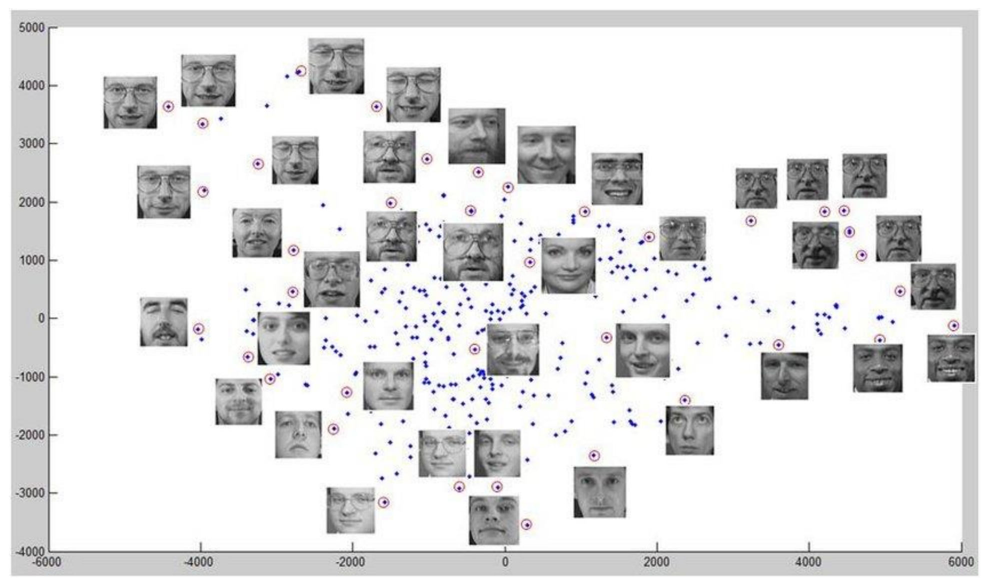


Expectation
Calculus
Optimization
Mathematics
in
Data Science
Bayes
rule
Matrices
Probability
density
Random
Variables

모든 데이터는 고차원의 점(point)

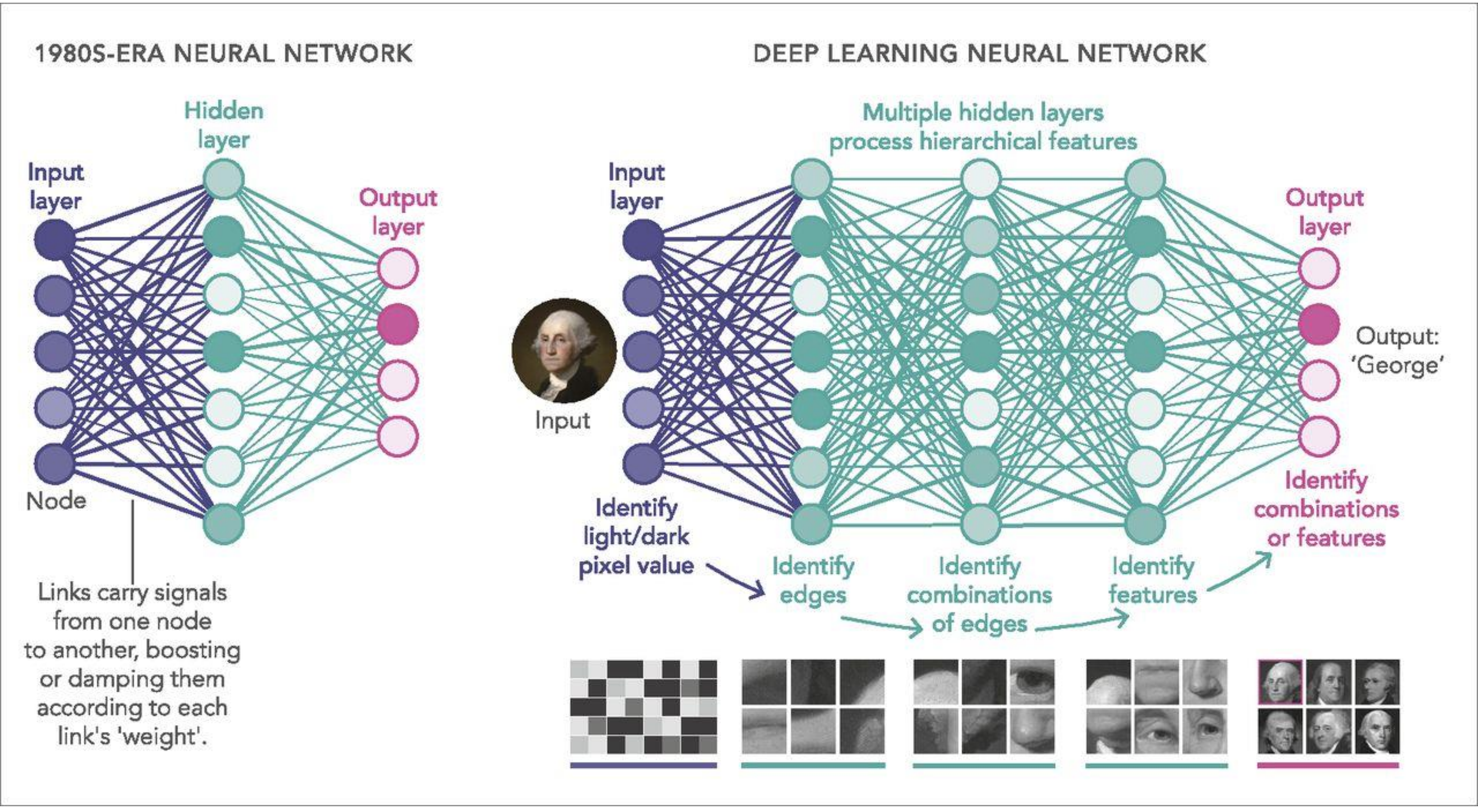
y=ax+b 에서 이제 데이터가 한없이 커지면 Y(출력 데이터셋) = WX + b (X는 입력데이터셋)

- 임의의 데이터는 고차원의 한 점으로 표현 가능
- Q. 강아지와 고양이를 구분하기 위한 모델과 학습 방법은?



사진별 픽셀을 하나씩 나열해서 점으로 표현(RGB 3개의 값이 존재)

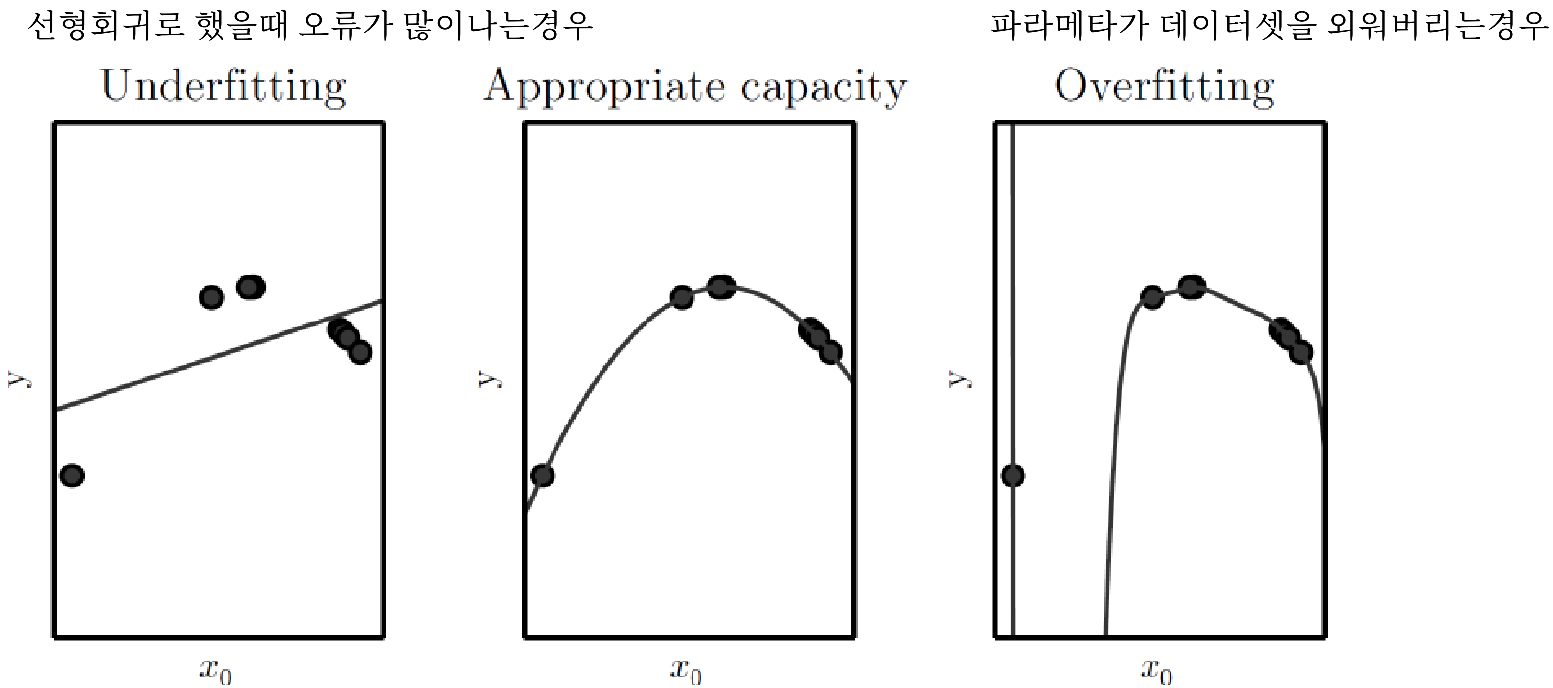
(Deep) Neural Network



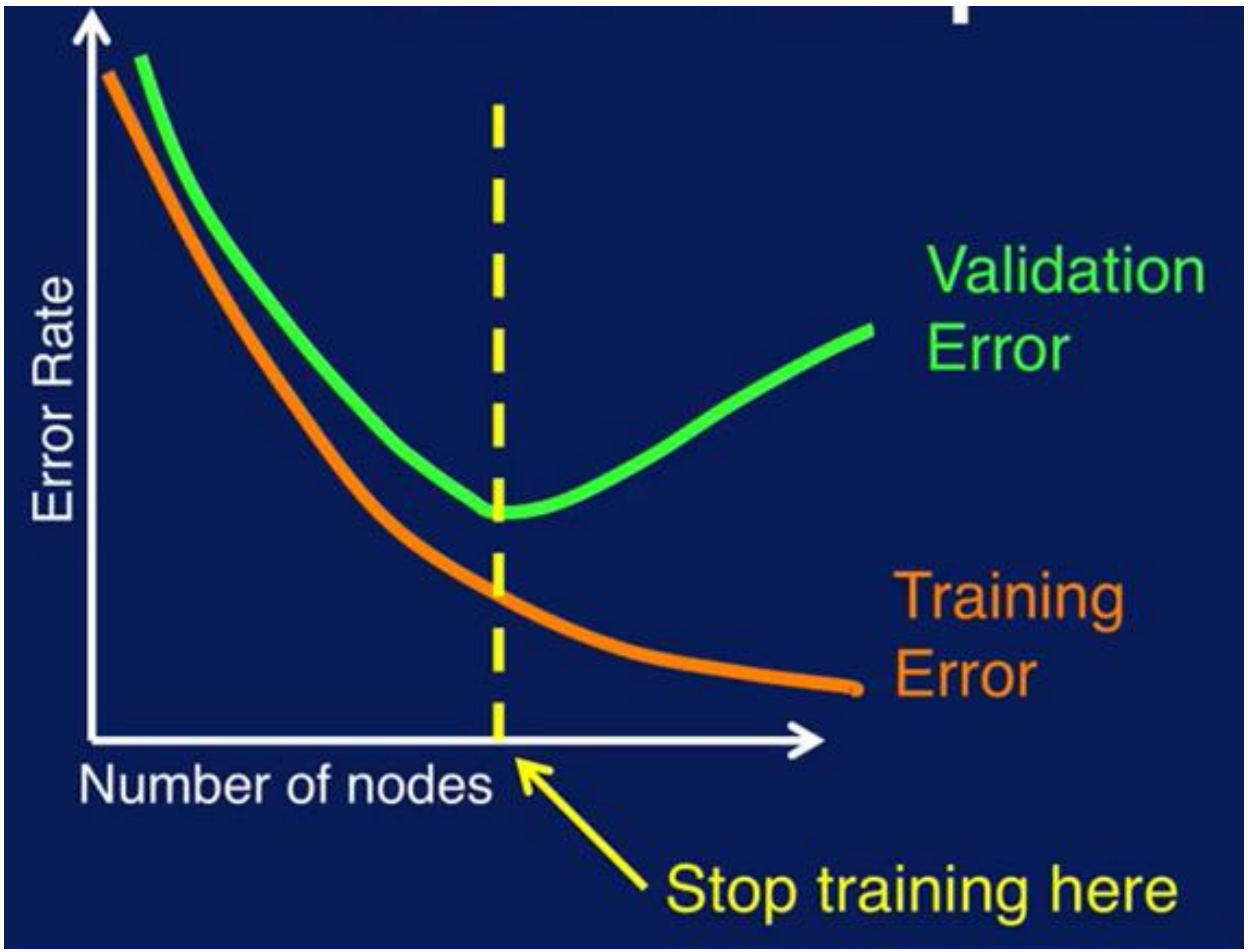
3. 모델 선택 및 성능평가

모델 선택 : Under/Over Fitting

- Underfitting : 모델이 너무 단순한 경우 발생
- Overfitting : 모델이 너무 복잡한 경우 발생



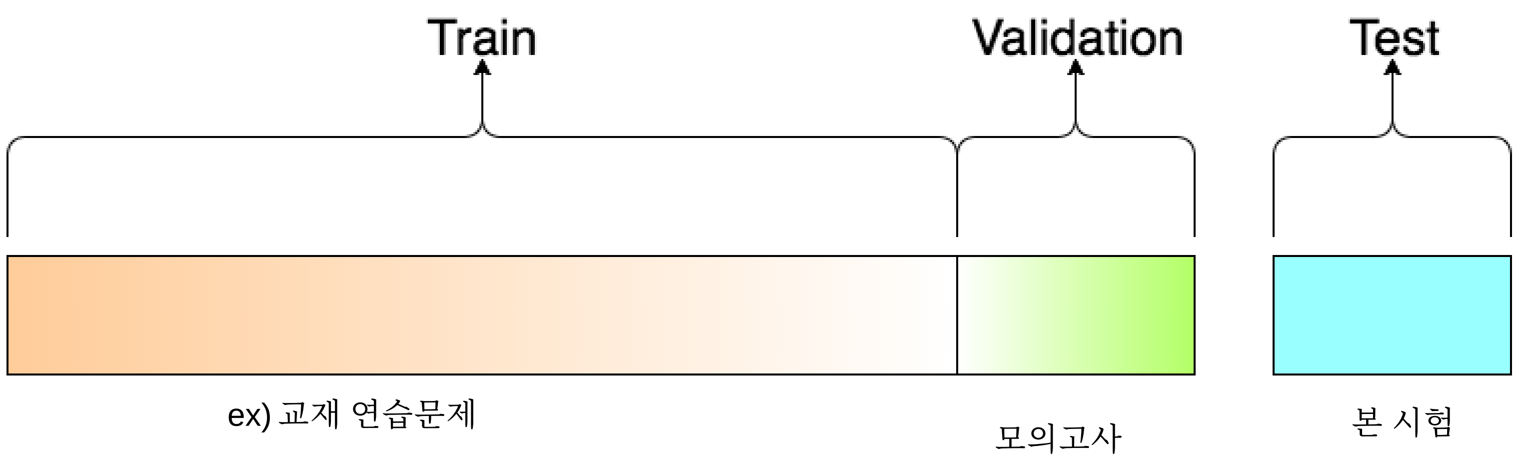
언제까지 학습(최적화)를 해야할까?



모델 성능평가 : 데이터셋의 구분

- Train Set : 모델 학습용 데이터 셋
- Validation Set : 학습시 Overfitting 등을 체크할 때 사용
  전체다 훈련시키지 않고 나눠서 검증 및 측정
- Test Set : 모델 성능 측정시 사용 (학습시 사용되지 않은 데이터 셋)

* 그럼에도 불구하고, 실제 성능을 보장해 주지는 못함



성능 지표

앞이 실제 답, 뒤가 예측한 답

	Actual Positive	Actual Negative
Predicted Positive	TP	FP
Predicted Negative	FN	TN

$$\text{취소율(recall rate)} = \frac{TP}{TP + FN}$$

$$\text{정밀도(precision)} = \frac{TP}{TP + FP}$$

$$\text{참 긍정률(TPR, True Positive Rate)} = \frac{TP}{TP + FN}$$

$$\text{거짓 긍정률(FPR, False Positive Rate)} = \frac{FP}{FP + TN}$$