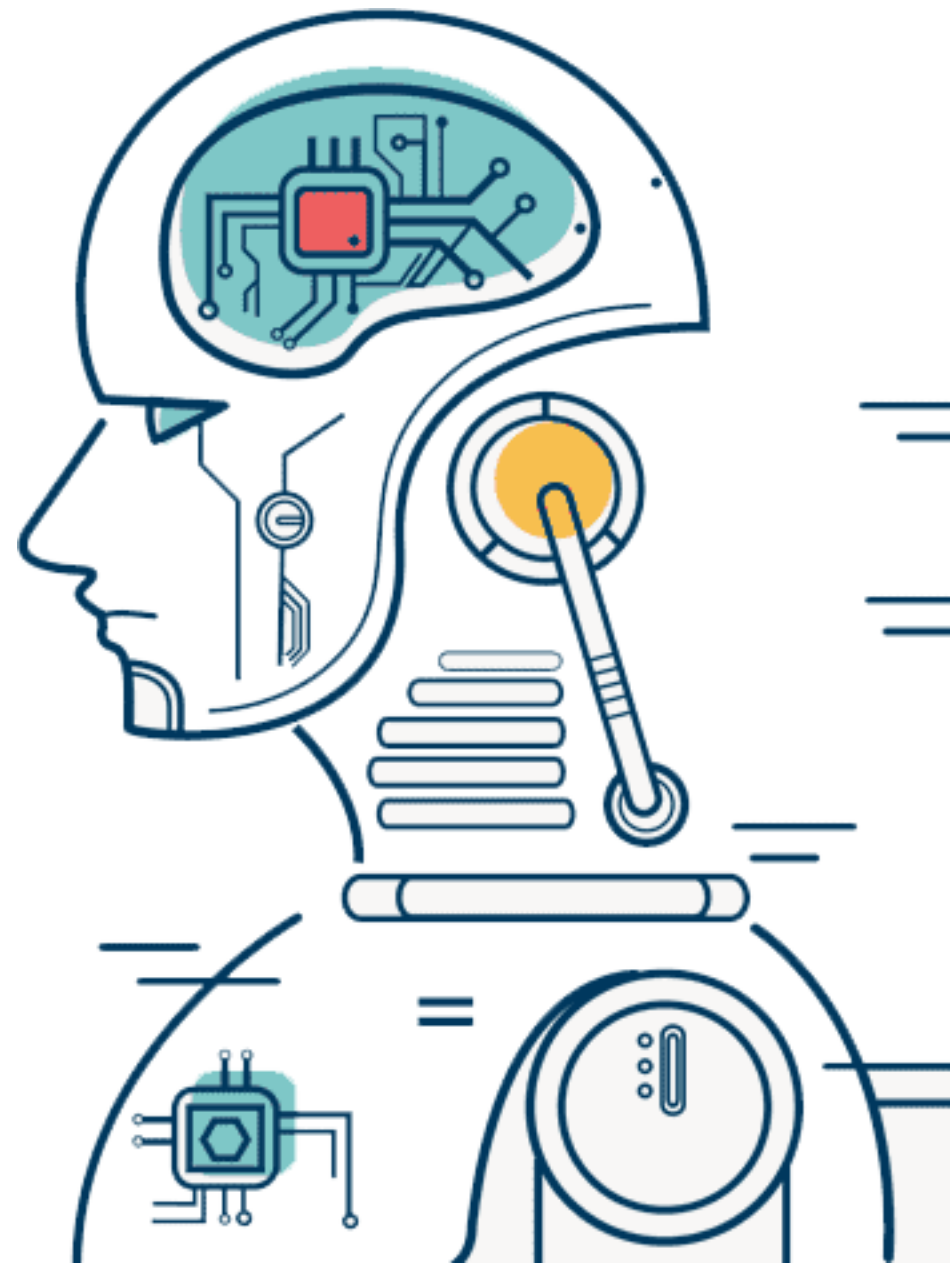


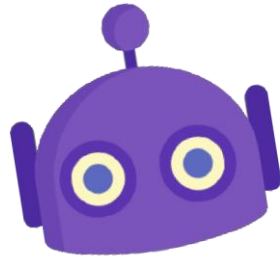
# Machine Learning

## Chapter 6 지도 학습

(Logistic Regression, SVM, 분류평가지표,  
GridSearch)

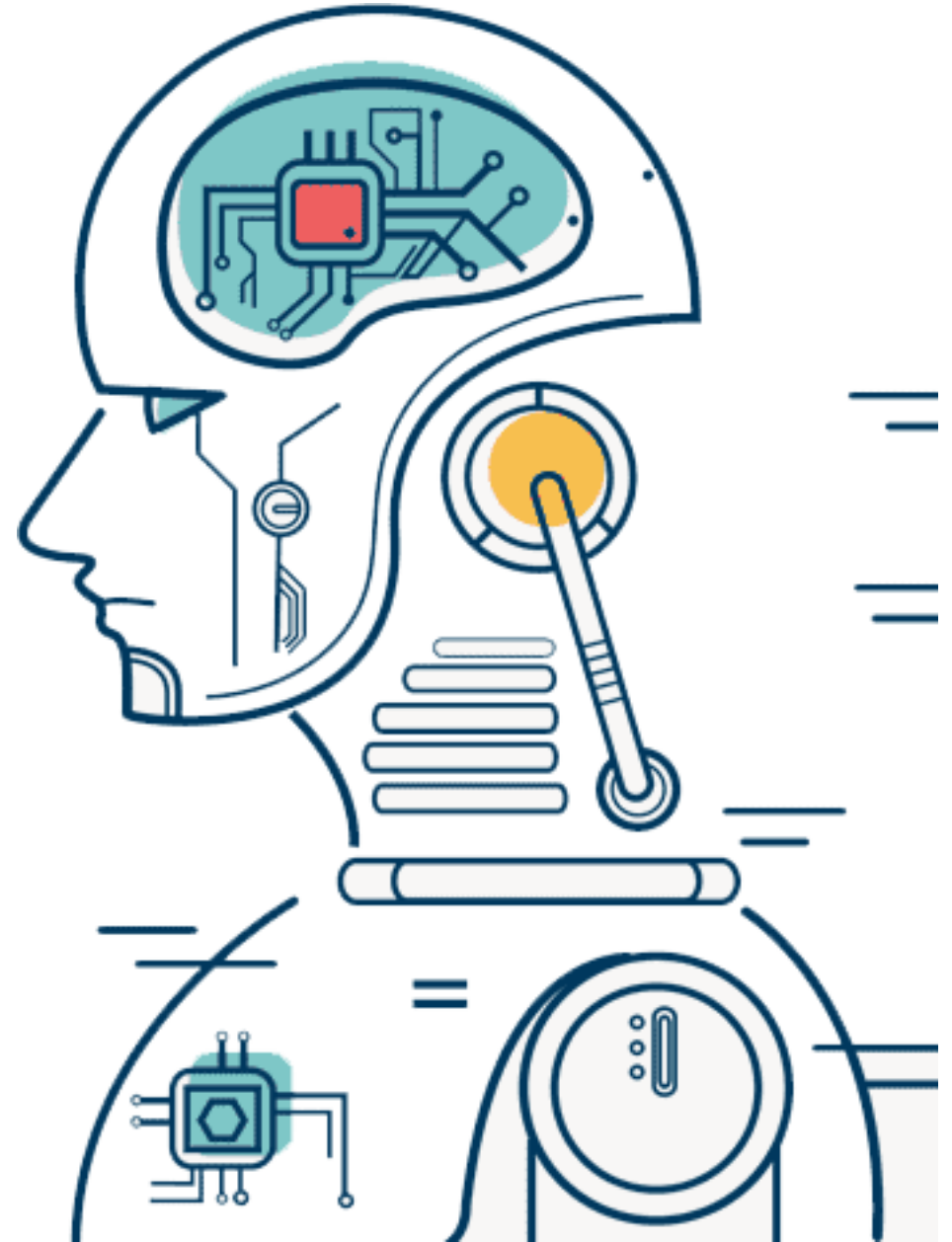


- 선형 분류모델을 이해하고 사용 할 수 있다.
- 다양한 분류평가 지표를 이해 할 수 있다.
- GridSearch를 이용한 파라미터 튜닝을 할 수 있다.



# Linear Model

(Classification)



## 분류용 선형 모델

$$y = w_1x_1 + w_2x_2 + w_3x_3 + \cdots + w_px_p + b > 0$$

- 특성들의 가중치 합이 0보다 크면 class를 +1 (양성클래스)로 0보다 작다면 클래스를 -1 (음성클래스)로 분류한다.
- 분류용 선형모델은 결정 경계가 입력의 선형함수

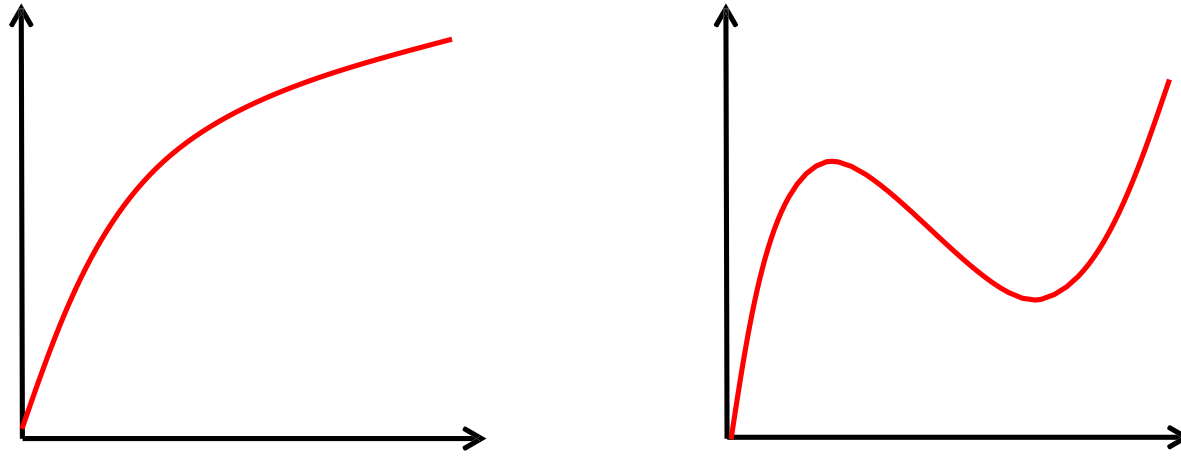
## 분류용 선형 모델

- Logistic Regression  
(Regression 단어가 붙지만 분류용 모델)
- Linear Support Vector Machine

## 장단점

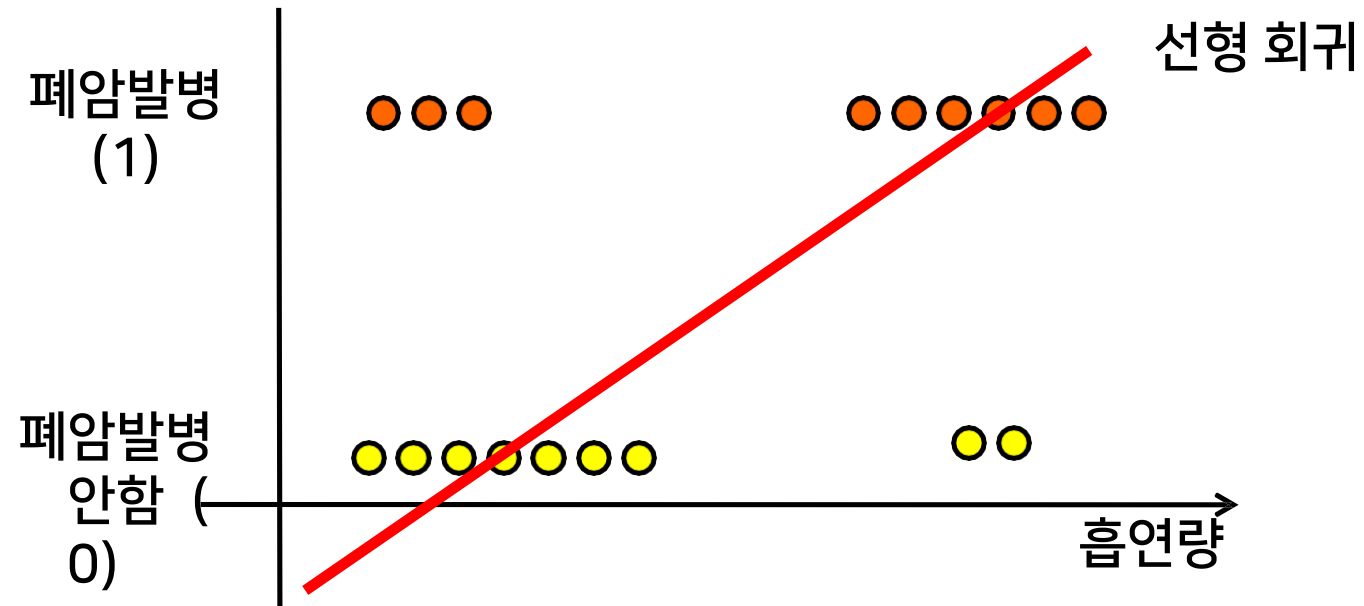
- 선형 모델은 학습 속도가 빠르고 예측도 빠르다.
- 매우 큰 데이터 세트와 희소 (sparse)한 데이터 세트에서도 잘 동작한다.
- 특성이 많을 수록 더욱 잘 동작한다.
- 저차원(특성이 적은)데이터에서는 다른 모델이 더 좋은 경우가 많다.

- 선형 회귀로 풀리지 않는 문제 → 독립변수와 종속변수가 비선형 관계인 경우



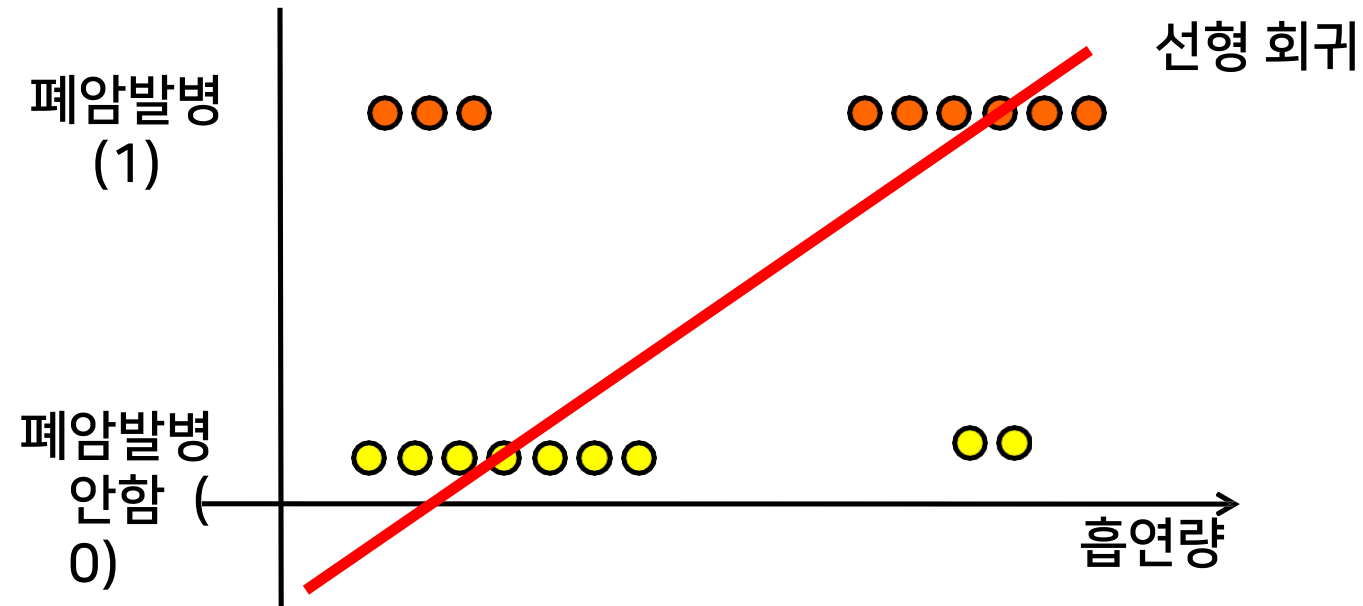
- 회귀를 사용하여 데이터가 어떤 범주에 속할 확률을 0에서 1 사이의 값으로 예측하고 그 확률에 따라 가능성이 더 높은 범주에 속하는 것으로 분류해주는 지도 학습 알고리즘

흡연량과 폐암 발병의 관계 → 연속적으로 변하는 흡연량의 결과에 대해 “폐암에 걸렸다” 또는 “걸리지 않았다?”의 결과만 필요



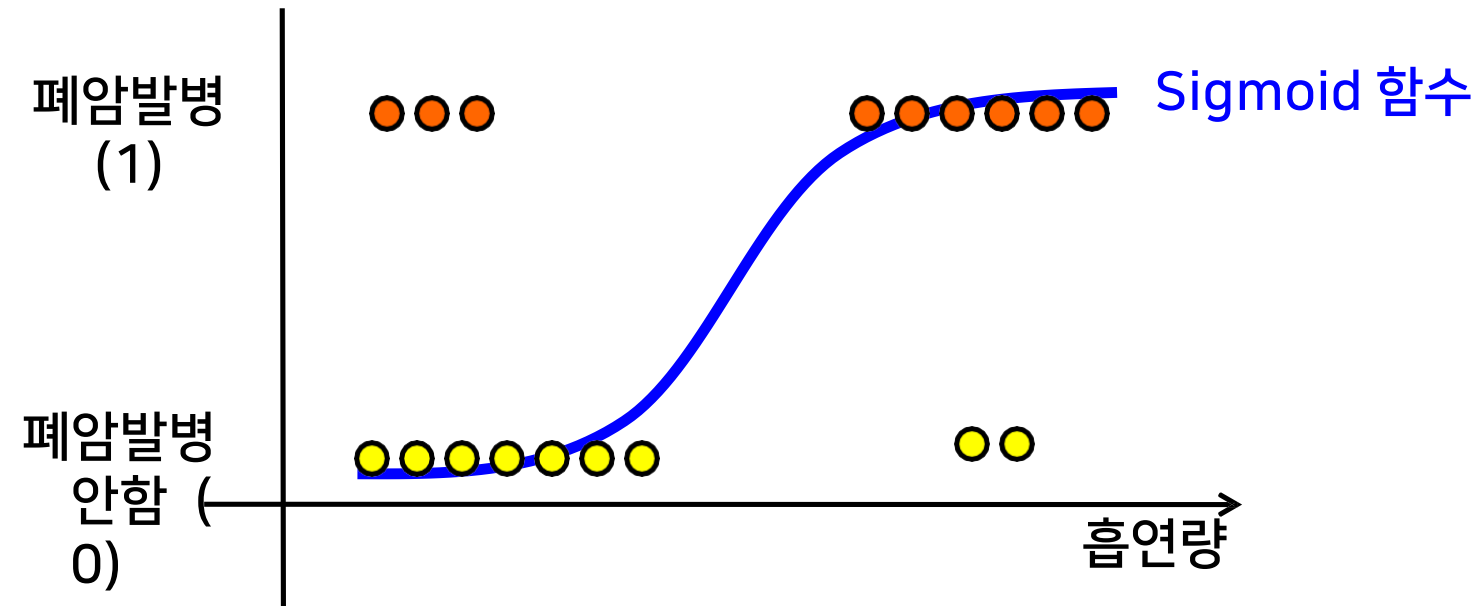


## - 선형 회귀



흡연량이 아주 작은 경우 폐암 발생확률이  
음수 값이거나 1보다 커지는 문제

- 해결 방법은 ?




흡연량에 따라 폐암 발생확률이 0~1 범위의 값이 됨

$$y = wx$$

폐암 유무  
(0, 1)

흡연량  
( $-\infty \sim \infty$ )



종속변수  $y$ 와 독립변수  $x$ 가 동일한 범위가 되도록 조정  
→  $y$ 를 확률 모형으로 변환  
→ Log를 취해서 범위를  $-\infty$  에서  $\infty$  로 변경

$$\frac{p(y)}{1-p(y)} \rightarrow \ln \frac{p(y)}{1-p(y)} = wx$$

$x$ 가  $-\infty \sim \infty$  범위의 값이어도  
 $p(y)$ 는 0~1 범위 값이 나옴

$$p(y) = \frac{e^{wx}}{1 + e^{wx}}$$

Sigmoid 함수

## 주요 매개변수(Hyperparameter)

scikit-learn의 경우

LogisticRegression(C, max\_iter)

- 규제 강도의 역수 : C  
(값이 작을수록 규제가 강해짐)
- 최대 반복횟수 : max\_iter  
(값을 크게 잡아 주어야 학습이 제대로 됨)
- 기본적으로 L2규제 사용, 중요한 특성이 몇 개 없다면 L1규제를 사용해도 무방  
(주요 특성을 알고 싶을 때 L1 규제를 사용하기도 한다.)

## wine 데이터셋

- 포르투갈의 비뉴 베르드 지방에서 만들어진 와인을 측정한 데이터
- 1,599개의 레드와인 데이터, 4,898개의 화이트와인 데이터 (총 6,497개 데이터)
- 12개의 정보와 1개의 클래스로 구성

	0	1	2	3	4	5	6	7	8	9	10	11	12
	주석산 농도	아세트 산농도	구연산 농도	진류당 분농도	염화나트 륨농도	유리아 화산 농도	총 아 황산 농도	밀도	pH	황산 칼륨 농도	알코올 도수	와인맛 (5-9등 급)	레드 1/화 이트0
964	8.5	0.47	0.27	1.9	0.058	18	36	0.99518	3.16	0.85	11.1	6	1
664	12.1	0.4	0.52	2	0.092	18	54	1	3.03	0.66	10.2	5	1
1692	6.9	0.21	0.33	1.8	0.034	18	136	0.9899	3.25	0.41	12.6	7	0
5801	6.7	0.24	0.31	2.3	0.044	18	113	0.99013	3.29	0.46	12.9	6	0
6497	6	0.21	0.38	0.8	0.02	22	98	0.98941	3.26	0.32	11.8	6	0

wine 데이터를 이용한  
Logistic Regression 모델 학습

최적의 규제값  $C$ 를 찾아보자

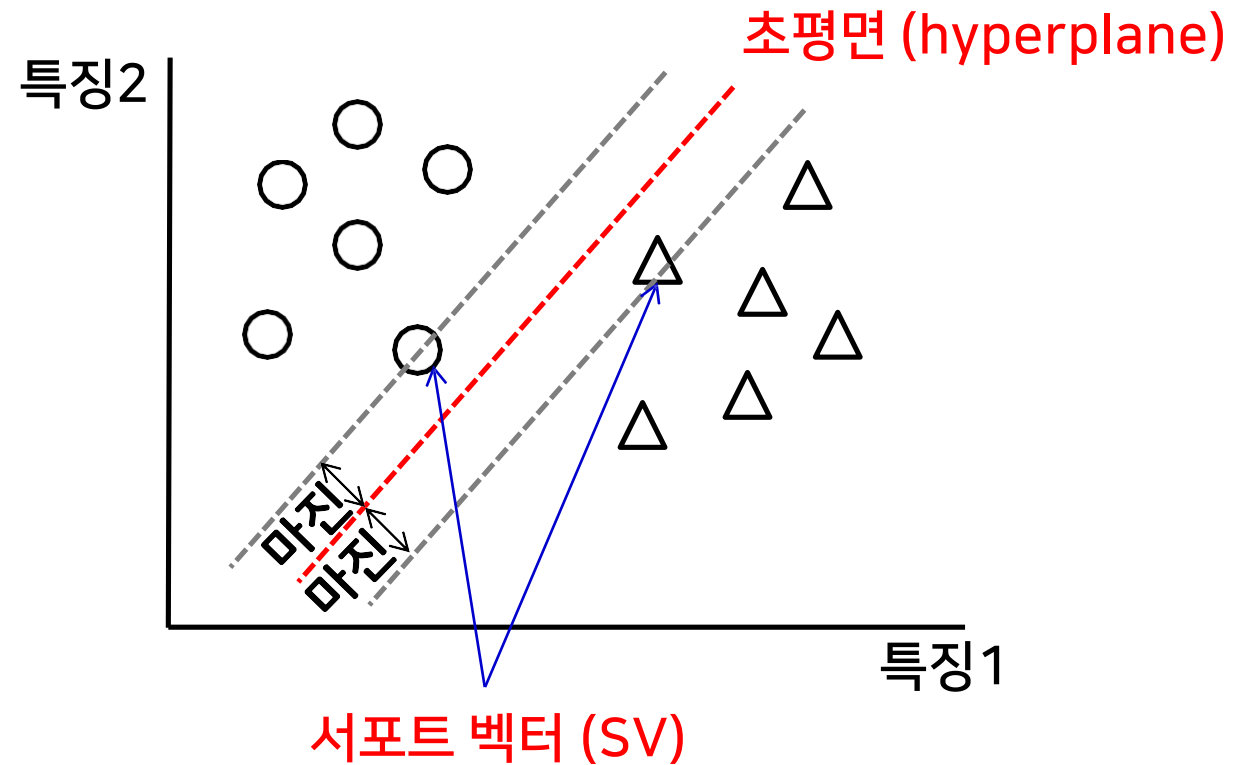
교차 검증을 적용해 보자



## SVM (Support Vector Machines)

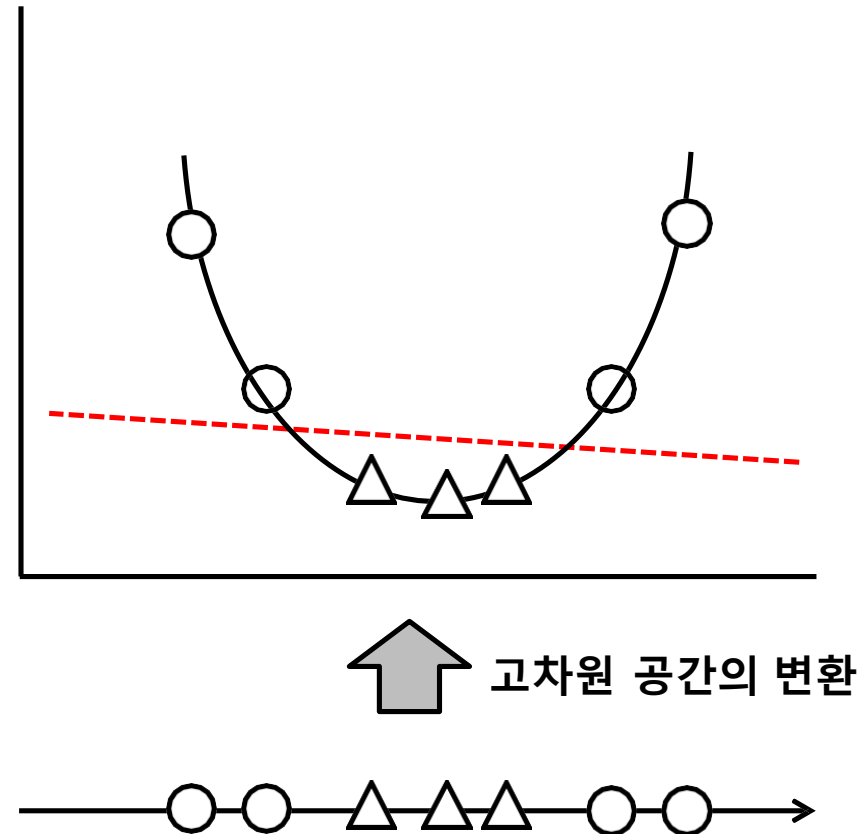
- 종이에 선형적으로 분리 가능한 2가지 유형의 포인트가 있다고 가정하면 SVM은 이 점들을 2가지 유형으로 분리하고 모든 점들로부터 가능한 멀리 위치하는 직선을 발견
- N차원 장소에서 2가지 유형의 점 집합이 주어지면 SVM은 (N-1) 차원의 초평면(hyperplane)을 생성하여, 이 점들을 두 그룹으로 분리
- 커널이라는 방법을 사용하여 비선형 데이터 분리 → 선형 커널, 다항식 커널, RBF (Radial Basis Function) 커널

## SVM (Support Vector Machines)



## SVM (Support Vector Machines)

- 커널을 적용한 결정 경계의 변화



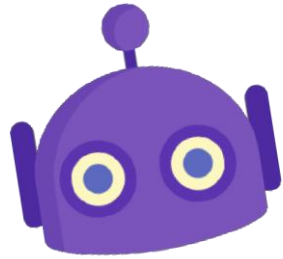
## 주요 매개변수(Hyperparameter)

scikit-learn의 경우

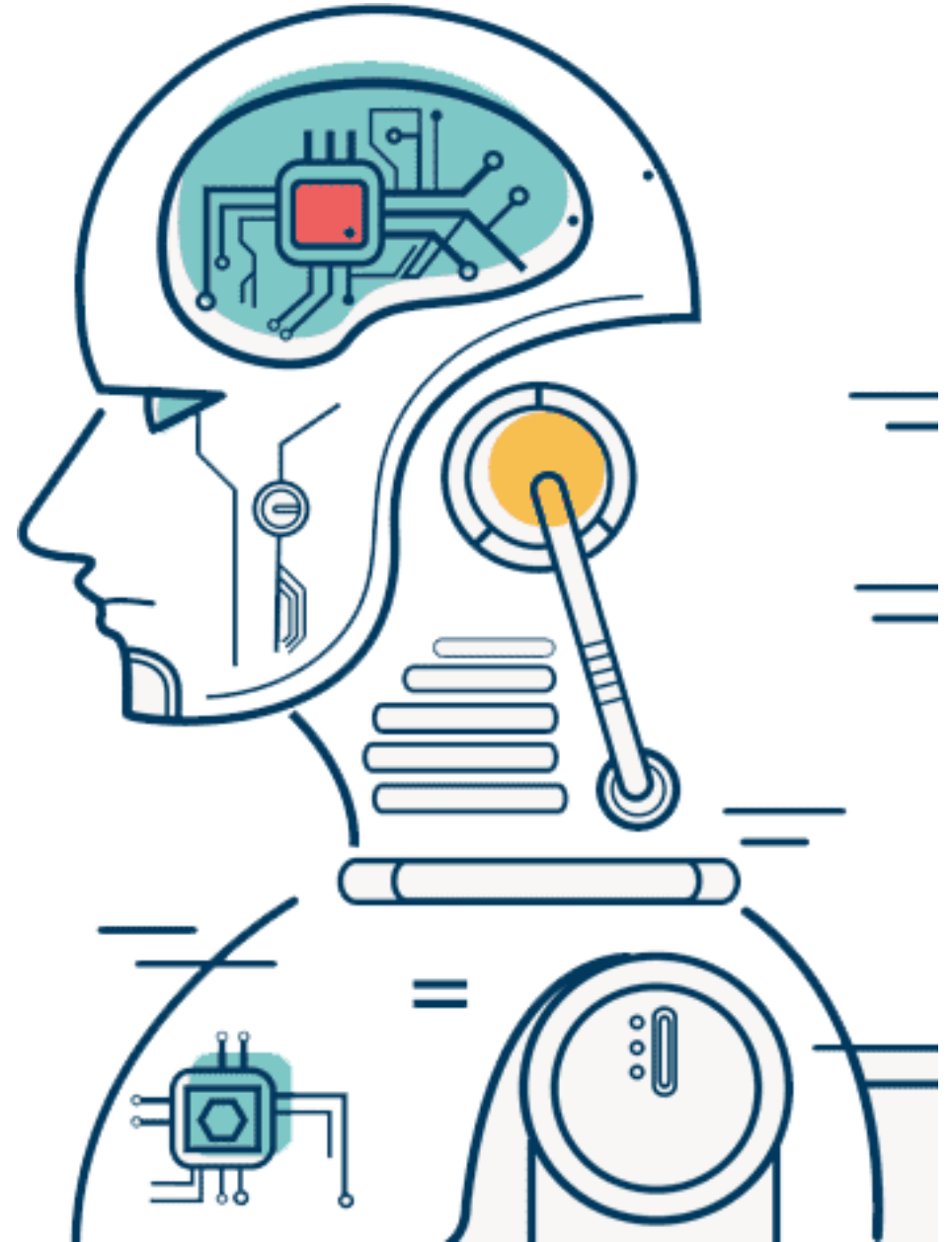
LinearSVC (C)

- 규제 강도 : C  
(값이 작을수록 규제가 강해짐)
- 기본적으로 L2규제를 사용, 하지만 중요한 특성이 몇 개 없다면 L1규제를 사용해도 무방  
(주요 특성을 알고 싶을 때 L1 규제를 사용하기도 한다.)

wine 데이터를 LinearSVC 모델로  
학습해보자.



# 분류 평가 지표



## Confusion\_matrix

negative class	TN	FP
positive class	FN	TP
	predicted negative	predicted positive

**정확도  
(Accuracy)**  
전체 중에 정확히  
맞춘 비율

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

## Confusion\_matrix

100명 중 암 환자는 5명

실제 암 X	95	0
	TN	FP
실제 암 O	0	5
	FN	TP
예측 암 X		
예측 암 O		

$$\frac{100}{100}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$



## Confusion\_matrix

100명 중 암 환자는 5명

실제 암 X	95	0
	TN	FP
실제 암 O	5	0
	FN	TP
예측 암 X		
예측 암 O		

$$\frac{95}{100}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

## Confusion\_matrix

negative class	TN	FP
positive class	FN	TP
	predicted negative	predicted positive

**재현율  
(Recall)**  
실제 양성 중에  
예측 양성 비율

$$\text{Recall} = \frac{TP}{TP + FN}$$

## Confusion\_matrix

100명 중 암 환자는 5명

실제 암 X	95	0
	TN	FP
실제 암 O	5	0
	FN	TP
	예측 암 X	예측 암 O

$$\frac{0}{5}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

## Confusion\_matrix

100명 중 암 환자는 5명

실제 암 X	95	0
	TN	FP
실제 암 0	0	5
	FN	TP
예측 암 X		예측 암 0

$$\frac{5}{5}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

## Confusion\_matrix

negative class	TN	FP
positive class	FN	TP
	predicted negative	predicted positive

**정밀도  
(Precision)**

예측 양성 중에  
실제 양성 비율

$$\text{Precision} = \frac{TP}{TP + FP}$$

## Confusion\_matrix

100명 중 암 환자는 5명

실제 암 X	95	0
	TN	FP
실제 암 0	0	5
	FN	TP
예측 암 X		예측 암 0

$$\frac{5}{5}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

## Confusion\_matrix

100명 중 암 환자는 5명

실제 암 X	0	95
	TN	FP
실제 암 O	0	5
	FN	TP
예측 암 X		
예측 암 O		

$$\frac{5}{100}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

## Confusion\_matrix

negative class	TN	FP
positive class	FN	TP
	predicted negative	predicted positive

**F1 - score**  
정밀도와 재현율의  
조화평균

$$F = 2 \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$



## 주요 매개변수(Hyperparameter)

scikit-learn의 경우

`confusion_matrix(실제값, 예측값)`

유방암 데이터를 LogisticRegression으로  
학습한 confusion matrix()를  
출력해 보세요.

- 낮은 재현율보다 높은 정밀도를 선호하는 경우

어린이에게 안전한 동영상(양성)을 걸러내는 분류기를 훈련시킬 경우 좋은 동영상이 많이 제외되더라도(낮은 재현율) 안전한 것들만 노출시키는(높은 정밀도) 분류기가 더 좋다.

- 낮은 정밀도보다 높은 재현율을 선호하는 경우

감시 카메라로 좀도둑(양성)을 잡아내는 분류기를 훈련시킬 경우 경비원이 잘못된 호출을 종종 받지만(낮은 정밀도) 거의 모든 좀도둑을 잡는(높은 재현율) 분류기가 더 좋다.

## 주요 매개변수(Hyperparameter)

scikit-learn의 경우

```
classification_report(실제값, 예측값)
```

유방암 데이터를 LogisticRegression으로  
학습한 `classification_report()`를  
출력해 보세요.

- **macro avg** : recall, precision, f1을 구해서 각각 평균을 낸 것 → 분류자가 각 클래스에 대해 얼마나 평균적으로 잘 동작하는지 알고 싶을 때 사용
- **weight avg** (가중평균) : 개별치에 각각의 중요도, 영향도(빈도) 등에 따라 가중치를 곱하여 구해지는 평균

	precision	recall	f1-score	support
악성	0.91	0.94	0.93	53
양성	0.97	0.94	0.96	90
accuracy			0.94	143
macro avg	0.94	0.94	0.94	143
weighted avg	0.94	0.94	0.94	143

## ROC(Receiver Operating Characteristic) curve

- 여러 임계값에서 분류기의 특성을 분석하는데 널리 사용되는 도구
- 클래스의 분포가 다르고 겹치는 부분이 존재한 경우에 Accuracy의 단점을 보완하기 위한 것
- 진짜 양성 비율 (TPR)에 대한 거짓 양성 비율 (FPR)을 나타냄

## ROC(Receiver Operating Characteristic) curve

- 가짜 양성비율(FPR) : 전체 음성 샘플 중에서 거짓 양성으로 잘못 분류한 비율
- 진짜 양성비율(TPR) : 재현율

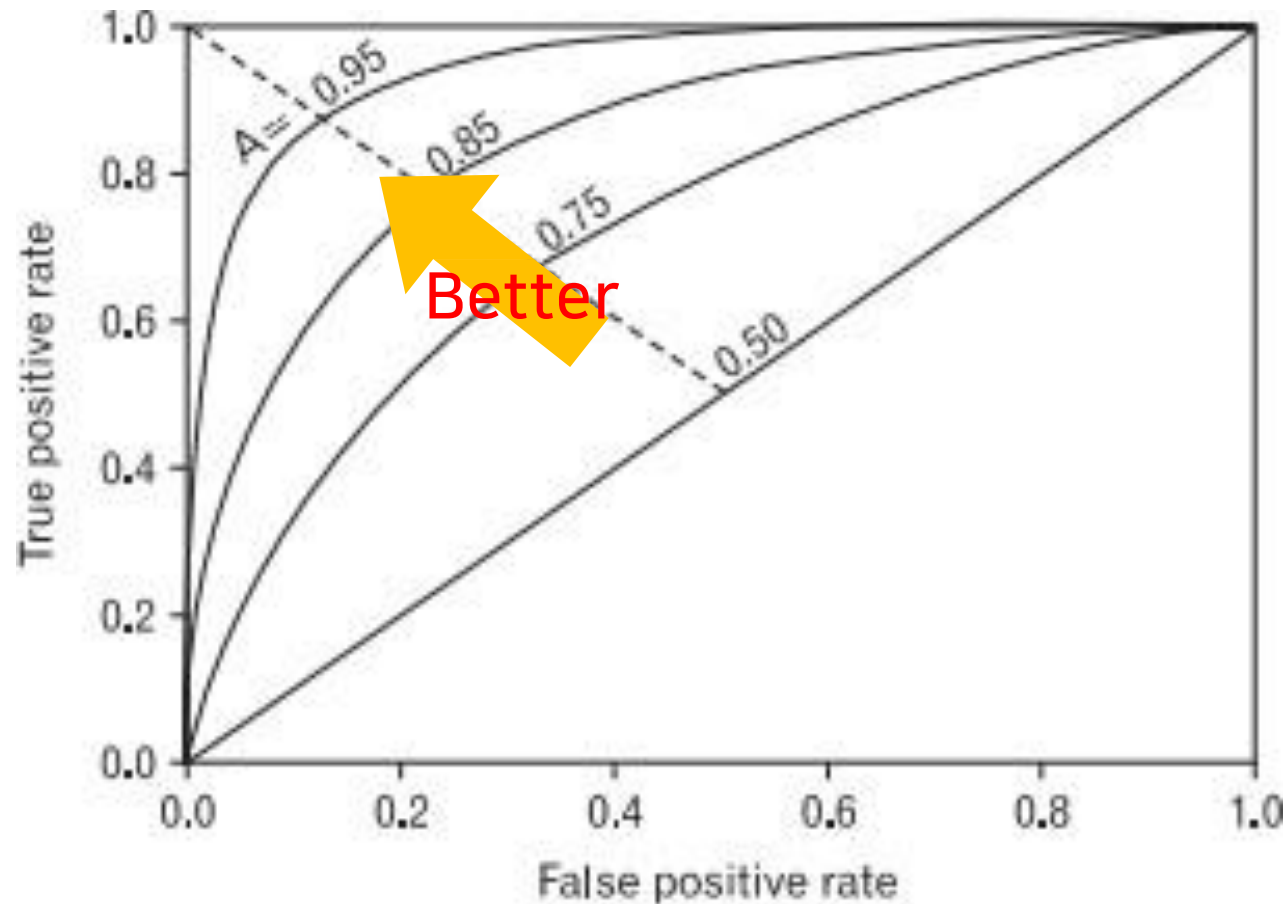
negative class	TN	FP
positive class	FN	TP
	predicted negative	predicted positive

$$FPR = \frac{FP}{FP+TN}$$

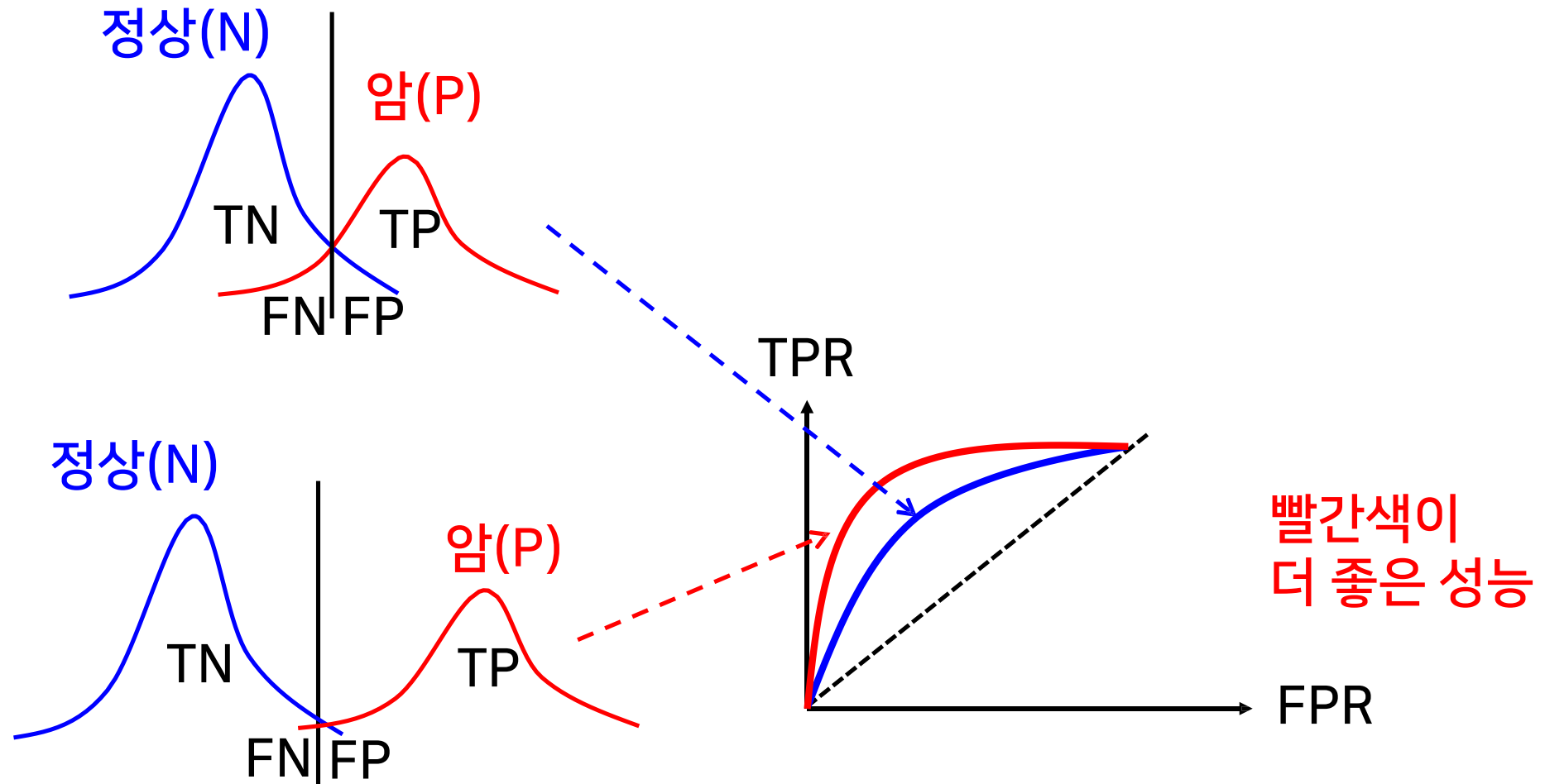
$$TPR = \frac{TP}{TP+FN} = \text{recall}$$

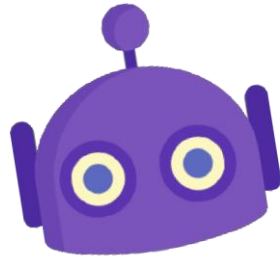


## ROC(Receiver Operating Characteristic) curve

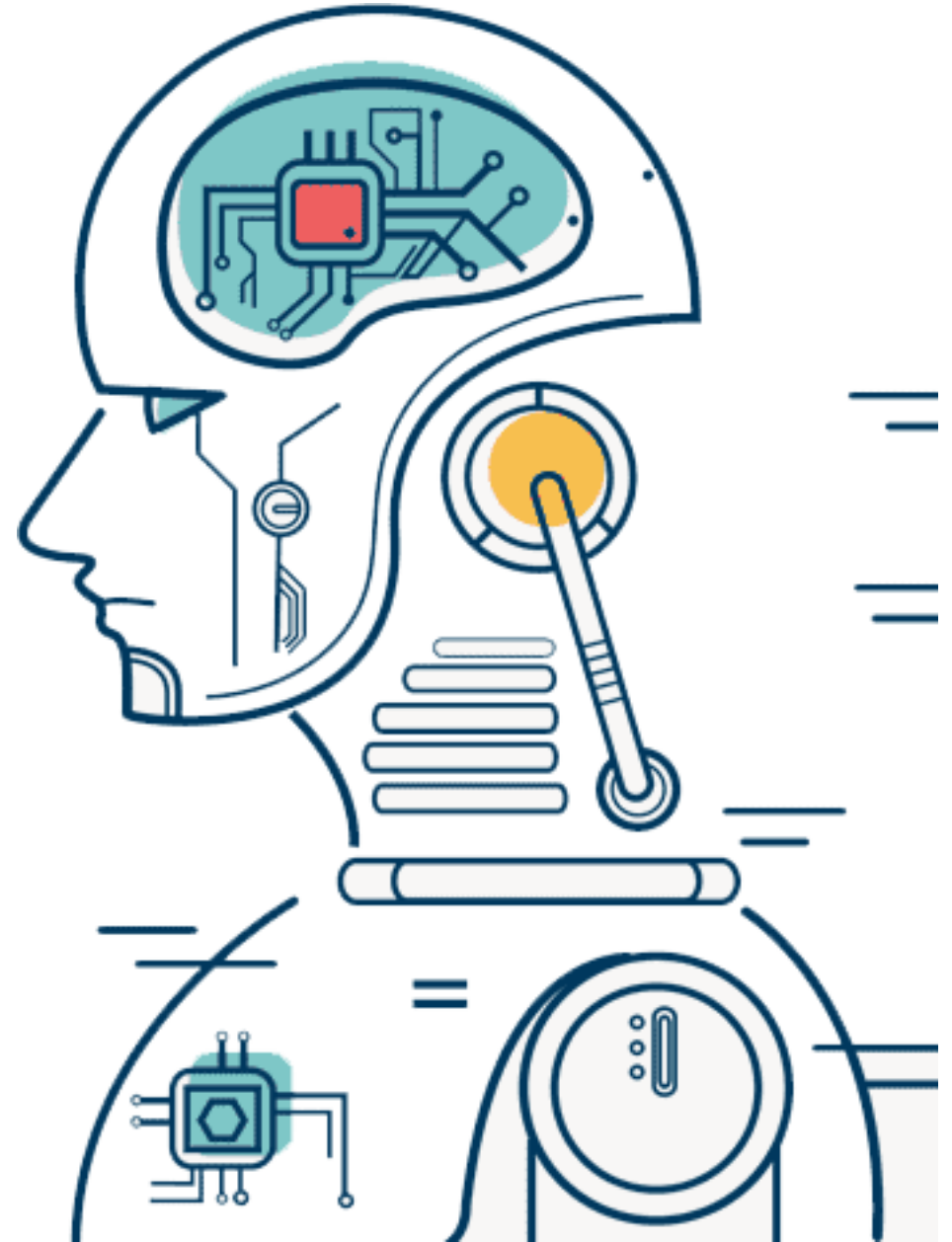


## ROC(Receiver Operating Characteristic) curve





# Gridsearch



- 매개변수를 선택하는 것은 머신러닝에서 중요한 일
- 관심 있는 매개변수들을 대상으로 가능한 모든 조합 시도하는 것

## 주요 매개변수(Hyperparameter)

scikit-learn의 경우

**GridSearchCV(모델, 모델의 파라미터목록, cv)**

- cv : 교차검증 시 나눌 fold 수

- **best\_params\_** : GridSearch 후에 찾는 최고의 파라미터 값
- **best\_score\_** : 최고의 파라미터를 사용한 교차 검증 점수
- **best\_estimator\_** : 전체 파라미터 값

iris 데이터를 사용하여  
DecisionTree 모델의 최적 파라미터 찾기  
(max\_depth, min\_leaf\_nodes, min\_samples\_leaf)