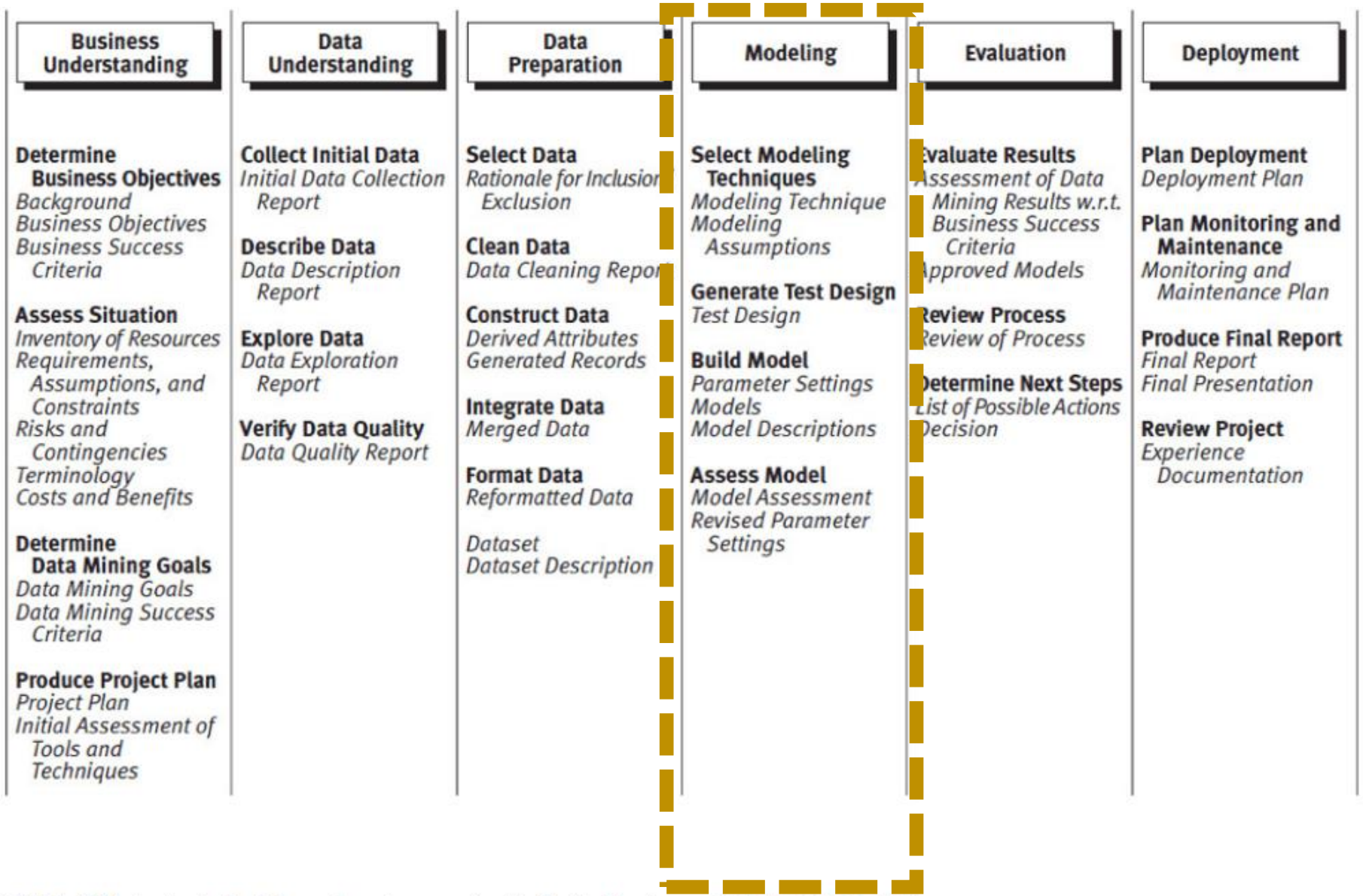
마이닝 알고리즘 2

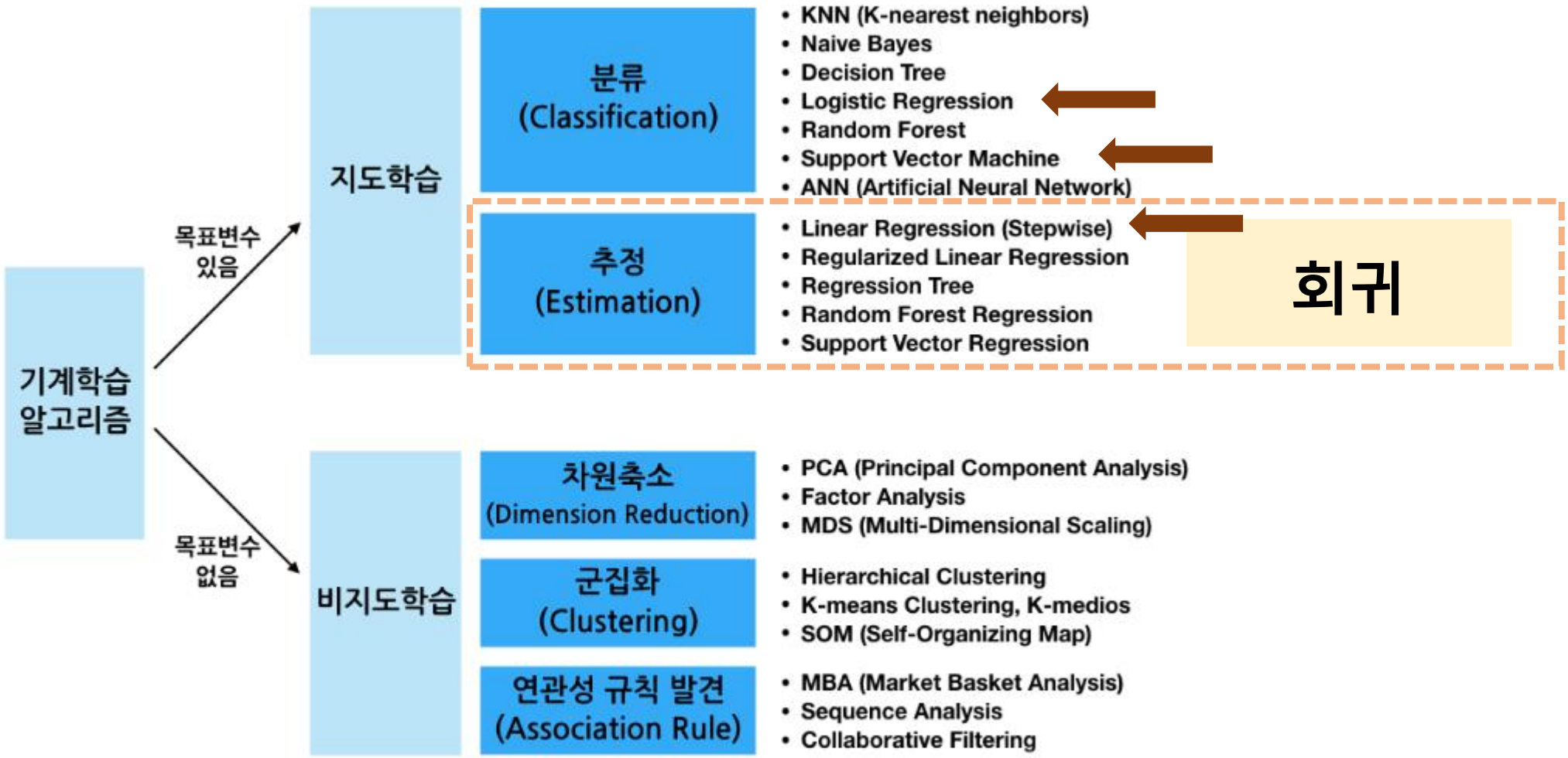
Regression, SVM

마이닝 알고리즘 (머신러닝 모델) -2



CRISP-DM tasks in **bold**, and outcomes in *italic* (table from CRISP-DM Guide)

마이닝 알고리즘 (머신러닝 모델) -2



https://dbrang.tistory.com/1520

상관관계 분석, 상관계수

Correlation

 # 상관관계.4th.pptx

회귀

마이닝 알고리즘 (머신러닝 모델) -2

회귀 (Regression)

마이닝 알고리즘 (머신러닝 모델) -2

회귀 (Regression)

공분산

$$\mathrm{Cov}(X,Y) = \frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})$$

상관

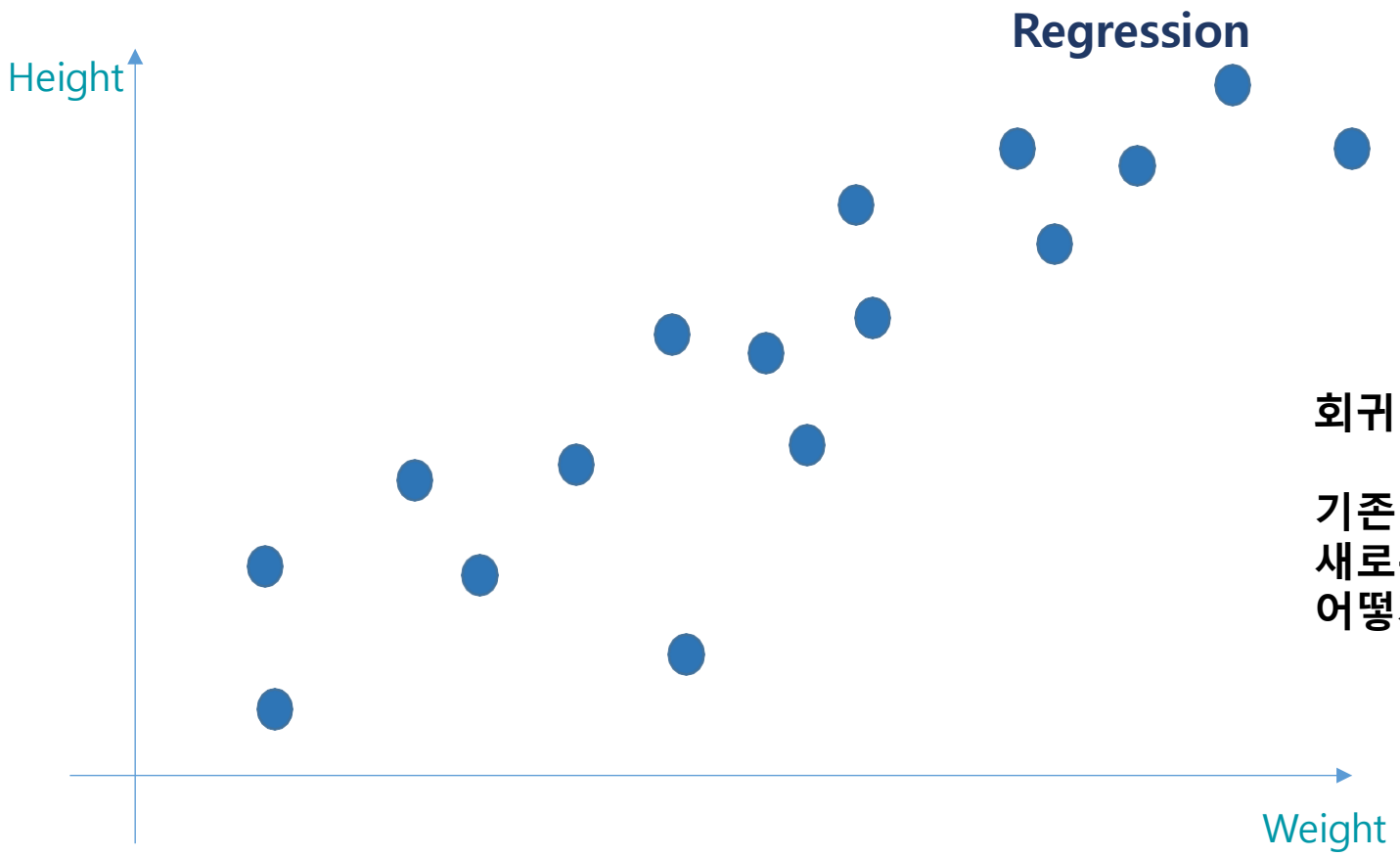
$$r = \frac{\mathrm{Cov}(X,Y)}{\sigma_X \sigma_Y}$$

회귀

$$y = mx + b$$

마이닝 알고리즘 (머신러닝 모델) -2

회귀 (Regression)

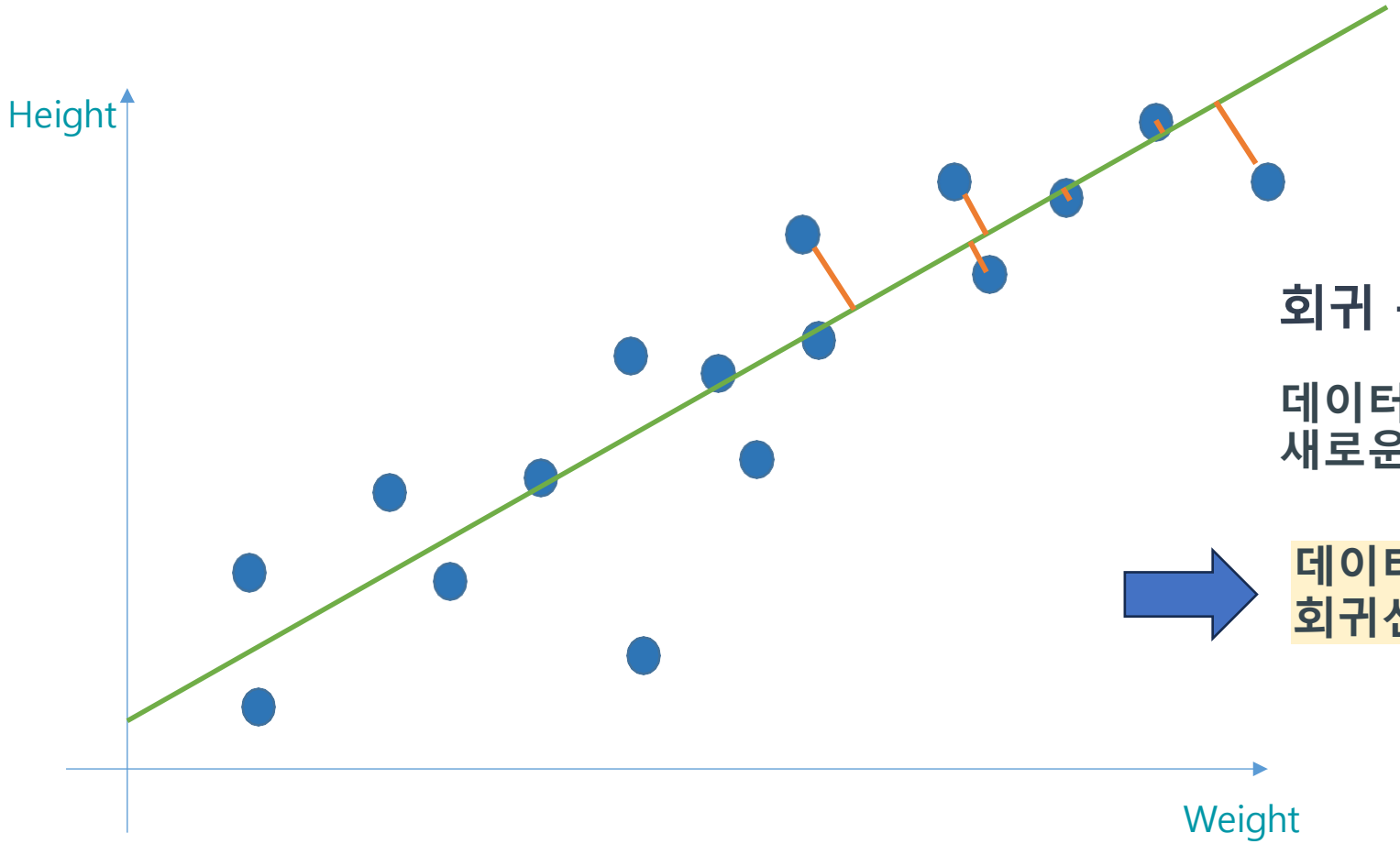


회귀 문제:

기존 데이터를 통한 학습 후,
새로운 데이터의 결과값을 예측
어떻게?

마이닝 알고리즘 (머신러닝 모델) -2

회귀 (Regression)



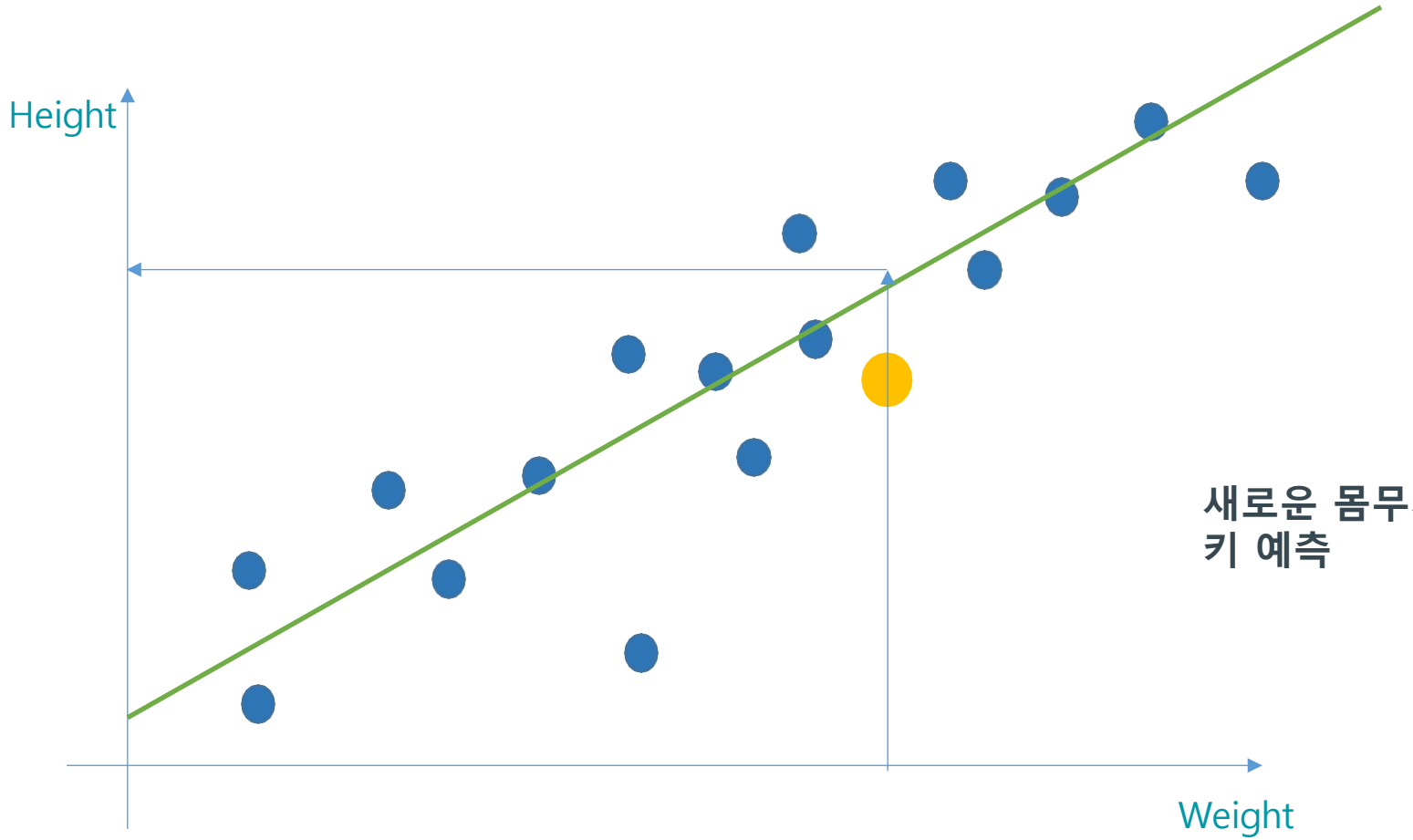
회귀 문제:

데이터가 이렇게 주어져 있을 때
새로운 데이터의 결과값을 어떻게 예측?

데이터들로부터 가장 오차가 적은
회귀선(Regression Line) 계산!

마이닝 알고리즘 (머신러닝 모델) -2

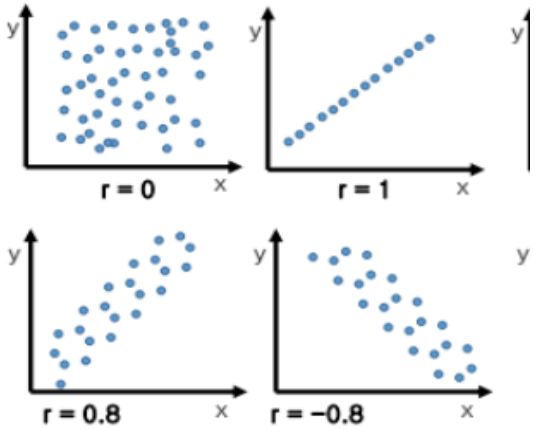
회귀 (Regression)



새로운 몸무게 데이터가 들어왔을 때,
키 예측

마이닝 알고리즘 (머신러닝 모델) -2

상관계수 *vs* 회귀계수



- **상관분석(correlation analysis)**
  - ✓ 두 변수 간의 **상관관계**를 탐색하기 위해 사용
  - ✓ 연속적인 **두 변수 간**에 **선형 관계**가 있는지 탐색 및 확인
  - ✓ 상관 분석에서 구해지는 상관 계수 r은 두 변수의 직선적인 연관성의 정도를 나타냄.
  - ✓ x가 증가하면 y도 증가하는가? 혹은 감소하는 가?를 나타냄
  - ✓ 한 변수의 변화가 다른 변수의 변화에 영향을 미치는 가에 대한 관계를 분석
  - ✓ but 상관관계는 두 변수 간의 관련성만을 의미할 뿐 **인과관계를 밝히지 못함**

    (상관 계수 r은 선형적인 관계의 정도를 의미)

마이닝 알고리즘 (머신러닝 모델) -2

상관계수 *vs* **회귀계수**

- **회귀분석(regression anaylsis)**
  - ✓ 상관분석의 확장
  - ✓ 입력 변수와 목표 변수 간의 **인과관계**를 파악
  - ✓ 회귀분석은 **한 개 이상의** 독립변수들(x)이 종속 변수(y)에 미치는 영향 분석
  - ✓ 회귀분석을 통해 측정한 x 값으로 목표인 y 값을 **추정**

마이닝 알고리즘 (머신러닝 모델) -2

회귀계수

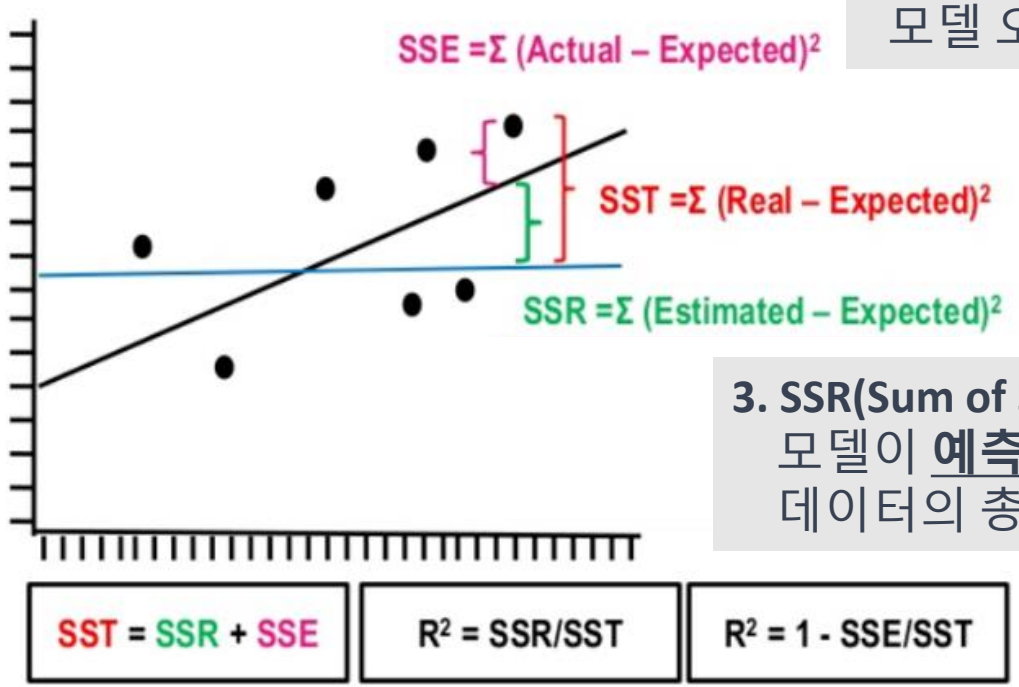
회귀 계수(Regression Coefficient)

- 독립 변수가 종속 변수에 미치는 **영향의 크기와 방향**을 나타내는 값
- **독립 변수의 한 단위 변화가 종속 변수를 얼마나 변화시키는지**를 나타냄

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n + \epsilon$$

마이닝 알고리즘 (머신러닝 모델) -2

회귀계수



1.SSE(Sum of Squares Due to Error):
실제 값과 회귀 모델에서 **예측한 추정값 간의 차이** 제곱의 합
모델 오류의 척도

2. SST(Total Sum of Squares):
실제 값과 데이터의 **전체 평균 간의 차이**를 제곱한 값의 합
이는 데이터의 총 분산을 나타내며 결정 계수 R^2를 계산할 때 분모로 사용.

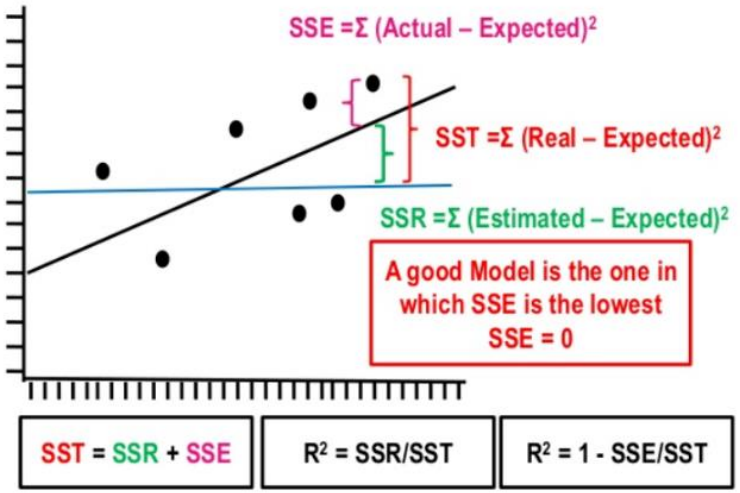
3. SSR(Sum of Squares Due to Regression):
모델이 **예측한 추정값과 데이터의 전체 평균 간의 차이**를 제곱한 값의 합
데이터의 총 분산 중 모델이 얼마나 많은 부분을 설명하는지 나타냄

R^2 : 회귀 모델의 독립 변수에 의해 설명되는 종속 변수의 분산 비율

ST = SSR + SSE'
전체 분산을 모델에 의해 설명되는 분산과 오류로 인한 분산으로 분할

마이닝 알고리즘 (머신러닝 모델) -2

회귀계수



1. SST (Total Sum of Squares, 총제곱합)
전체 변동성 = 데이터의 전체적인 퍼짐 정도

2. SSR (Regression Sum of Squares, 회귀제곱합)
모델이 설명하는 변동성 = 예측값이 평균값과 얼마나 차이가 있는가?

3. SSE (Error Sum of Squares, 오차제곱합)
모델이 설명하지 못하는 변동성 = 실제값과 예측값 사이의 차이
모델이 예측하지 못한 오차(Actual - Predicted)의 제곱합

ex, 시험 점수 60, 80, 50, 90, 75 평균 70점
SST : 각 학생 점수와 평균 사이의 차이를 제곱, 모두 더한 값 <- Data 가 얼마나 퍼져 있는가?
SSR : 모델이 예측한 점수와 평균 사이의 차이를 제곱, 모두 더한 값
<- 모델이 설명할 수 있는 데이터의 변화량 **(공부 시간)**
SSE : 실제 학생 성적과 모델 예측 값의 차이를 제곱, 모든 더한 값

마이닝 알고리즘 (머신러닝 모델) -2

회귀계수

SST (Total Sum of Squares)

$$SST = \sum (y_i - \bar{y})^2$$

SSR (Regression Sum of Squares)

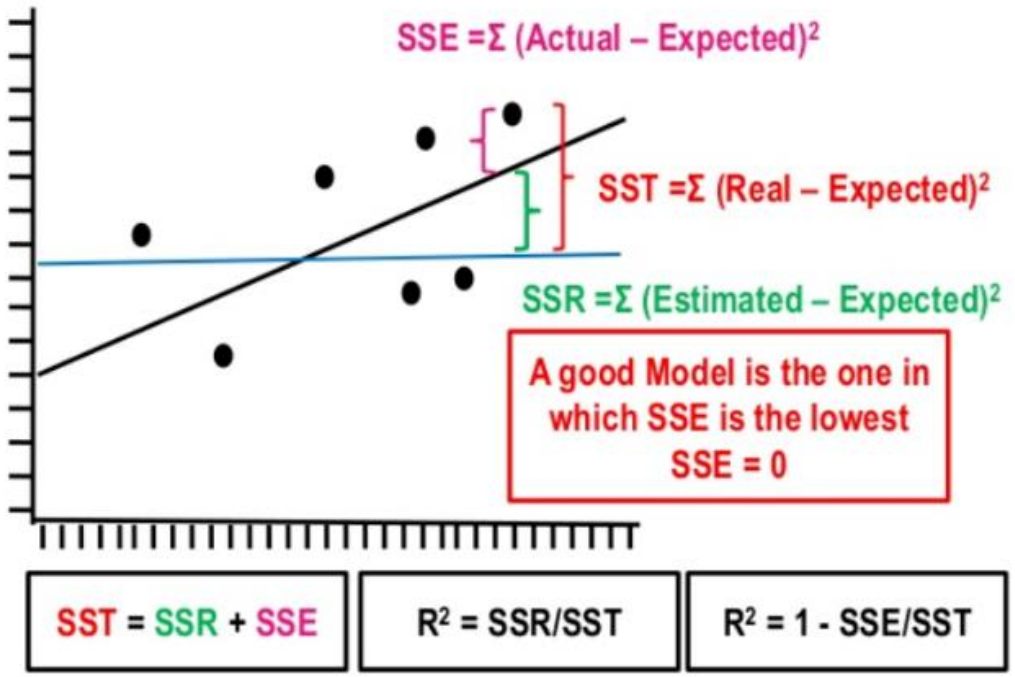
$$SSR = \sum (\hat{y}_i - \bar{y})^2$$

SSE (Sum of Squared Errors)

$$SSE = \sum (y_i - \hat{y}_i)^2$$

결정계수 R2

$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

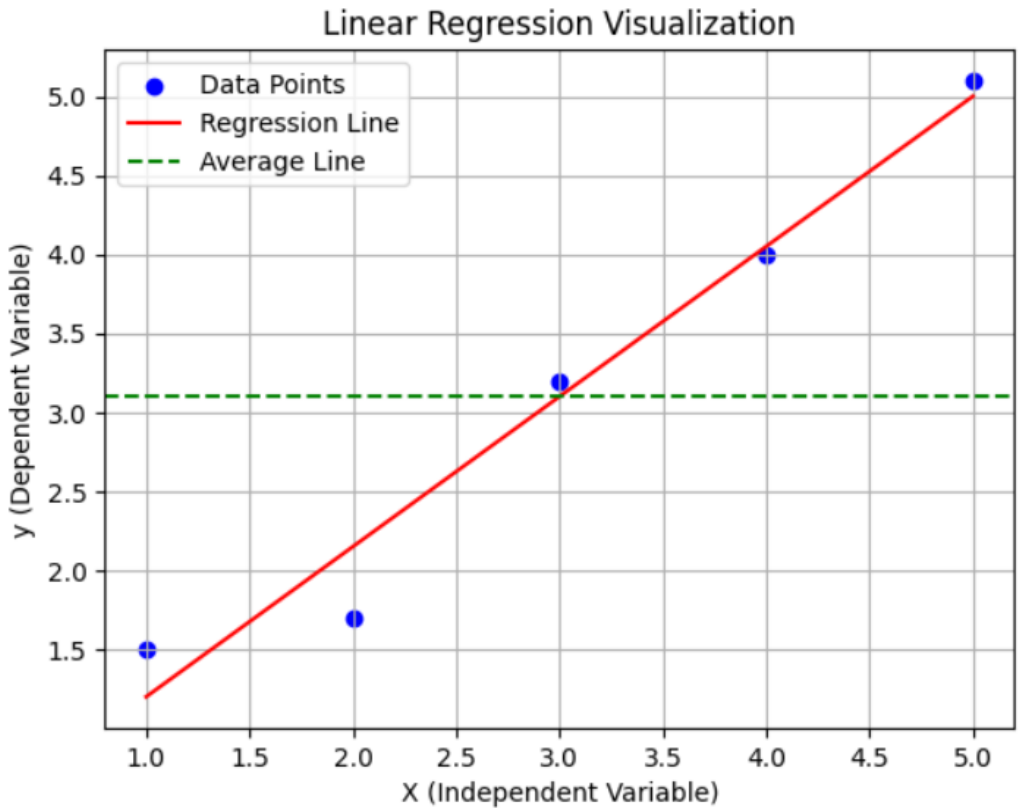


마이닝 알고리즘 (머신러닝 모델) -2

회귀계수

- 실습

  2.05.SST.SSR.SSE.R2.ipynb



마이닝 알고리즘 (머신러닝 모델) -2

회귀계수

```python
import numpy as np
from sklearn.linear_model import LinearRegression

# 예제 데이터
X = np.array([2, 3, 4, 5, 6])  # 공부 시간
Y = np.array([5, 7, 9, 11, 13])  # 시험 점수

# 상관계수 계산
correlation_matrix = np.corrcoef(X, Y)
correlation_coefficient = correlation_matrix[0, 1]
print("상관계수:", correlation_coefficient)

# 선형 회귀 모델 생성 및 훈련
model = LinearRegression()
model.fit(X.reshape(-1, 1), Y)

# 회귀계수
regression_coefficient = model.coef_[0]
print("회귀계수:", regression_coefficient)

# 결정계수 (R² ) 계산
r_squared = model.score(X.reshape(-1, 1), Y)
print("결정계수 (R² ):", r_squared)
```

```
상관계수: 0.9999999999999999
회귀계수: 2.0
결정계수 (R² ): 1.0
```

 # Regression.correlation.ipynb

- **상관계수는 두 변수 간의 선형 관계의 강도를 나타냄**
  단순히 두 변수 사이에 강력한 선형 연관성이 있음을 나타냄
- **회귀계수는 독립 변수 X가 종속 변수 Y에 미치는 영향의 크기와 방향을 나타냄**

마이닝 알고리즘 (머신러닝 모델) -2

회귀계수

```
상관계수: 0.9999999999999999
회귀계수: 2.0
결정계수 (R² ): 1.0
```

- **상관계수**: 거의 1에 가까운 이 상관계수는 두 변수 사이에 매우 강한 양의 선형 관계가 있음을 나타냄
  이는 한 변수가 증가할 때 다른 변수도 거의 동일한 비율로 증가한다는 것을 의미

- **회귀계수**: 회귀계수가 2.0이라는 것은 독립 변수(여기서는 공부 시간)가 한 단위 증가할 때마다 종속 변수(시험 점수)가 평균적으로 2단위 증가한다는 것을 나타냄
  예를 들어, 공부 시간이 1시간 증가하면 시험 점수는 평균적으로 2점 증가.

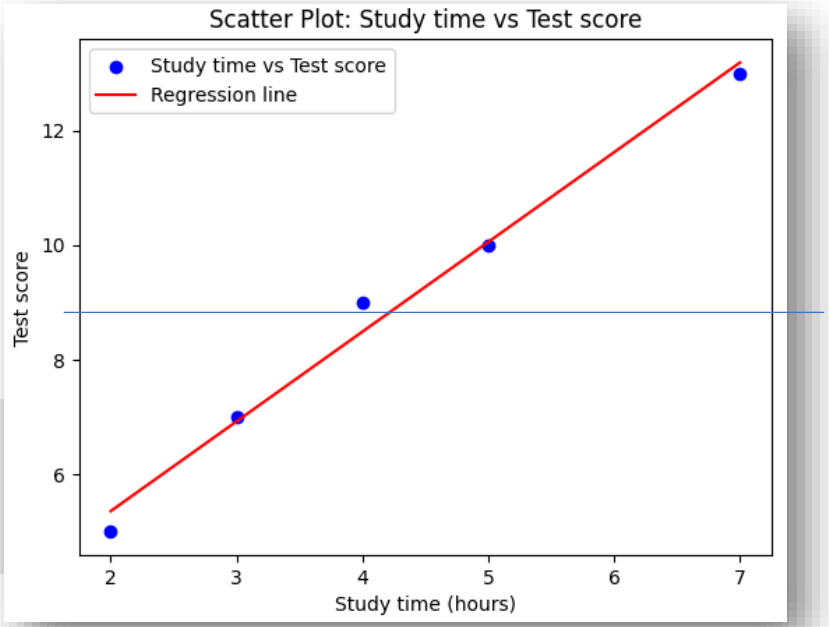
*** 결정 계수 R^2 :** 모델의 전반적인 적합도나 모델이 종속 변수의 변동성을 설명
R^2 값이 높을수록 모델이 종속 변수의 분산 중 더 큰 부분을 설명한다는 것을 나타냄

마이닝 알고리즘 (머신러닝 모델) -2

회귀계수

상관계수: 0.994

회귀계수: 1.567
결정계수 (R^2): 0.988



TASK
결과 해석

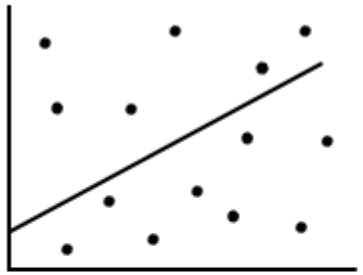
- **상관계수**:
- **회귀계수**:

  *** 결정 계수 R^2 :**

- **SST**:
- **SSE:**
- **SSR:**

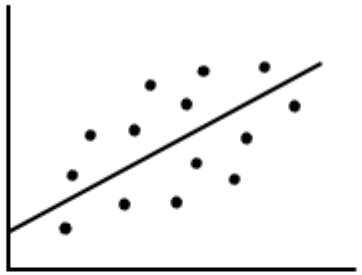
마이닝 알고리즘 (머신러닝 모델) -2

회귀



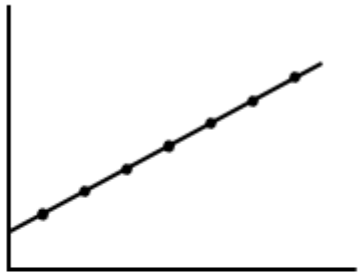
$R^2 = 0$

〈 믿을 게 못 된다 〉



$R^2 = 0.5$

〈 어느 정도 믿을 만 하다 〉



$R^2 = 1$

〈 믿을 만 하다 〉

마이닝 알고리즘 (머신러닝 모델) -2

다중 회귀 vs 선형 회귀

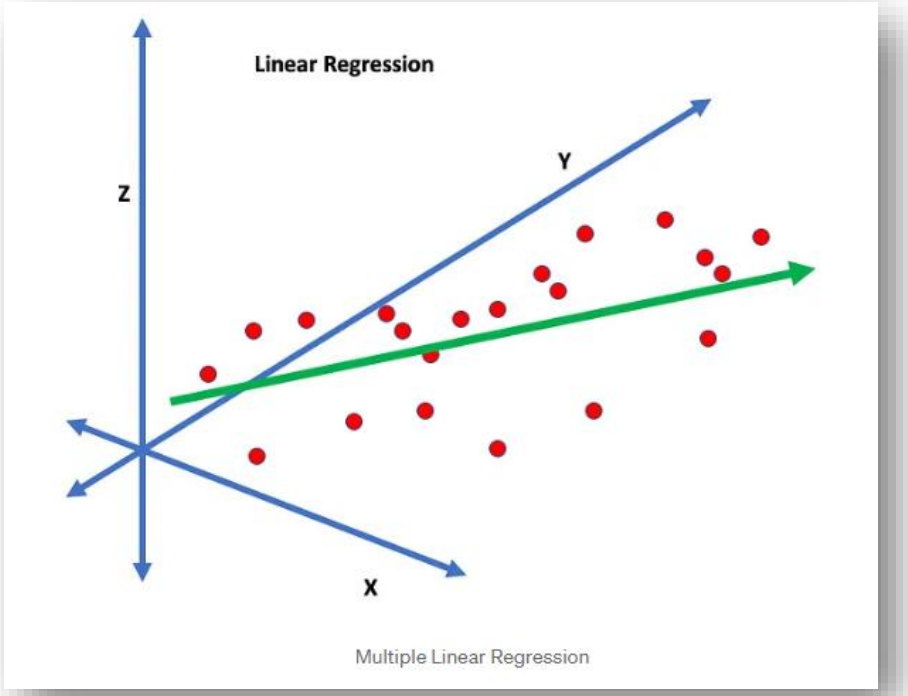
- **선형 회귀 (Simple Linear Regression)**
  - **하나의 독립 변수**와 하나의 종속 변수를 이용해 데이터 간의 선형 관계를 모델링하는 방법
    (하나의 입력 변수와 하나의 출력 변수 사이의 관계를 학습)
  - $\mathbf{y=w_1 \cdot x_1+b}$ (x_1:입력 변수, y: 출력 변수, w_1 :회귀 계수, b: 편향)

- **다중 선형 회귀 (Multiple Linear Regression)**
  - **여러 독립 변수**를 사용하여 종속 변수를 예측하는 회귀 방법
    (여러 개의 입력 변수와 하나의 출력 변수 사이의 관계를 학습)
  - $\mathbf{y=w_1 \cdot x_1+w_2 \cdot x_2+w_3 \cdot x_3+b}$ (x_1, x_2, x :입력 변수, y: 출력 변수, w_1... :회귀 계수, b: 편향)

마이닝 알고리즘 (머신러닝 모델) -2

다중 회귀 vs 선형 회귀

- **다중 선형 회귀 (Multiple Linear Regression)**
  - **여러 독립 변수**를 사용하여 종속 변수를 예측하는 회귀 방법
    (여러 개의 입력 변수와 하나의 출력 변수 사이의 관계를 학습)
  - $\mathbf{y=w_1 \cdot x_1+w_2 \cdot x_2+w_3 \cdot x_3+b}$ (x_1, x_2, x :입력 변수, y: 출력 변수, w_1... :회귀 계수, b: 편향)



마이닝 알고리즘 (머신러닝 모델) -2

- **실습**

2.05.MultipleRegression.iris.ipynb Multiple Linear Regression : 다중회귀 모델

아이리스(Iris) 데이터셋의 첫 세 가지 변수,

꽃받침의 길이(sepal length), 꽃받침의 너비(sepal width), 그리고 꽃잎의 길이(petal length)를 사용

꽃잎의 너비(petal width)를 예측하는 다중 선형 회귀 모델 개발

회귀 계수(Coefficients): [-0.19918374, 0.20693216, 0.52260085]
평균 제곱 오차(Mean squared error): 0.05
결정 계수(Coefficient of determination, R^2): 0.89

마이닝 알고리즘 (머신러닝 모델) -2

- **실습**

  2.05.regression.diabetes.ipynb

당뇨병 데이터셋(Diabetes dataset)을 사용
선형회귀모델 학습
타겟 변수(당뇨병 진행도) 예측

변수명	설명
Age	나이
Sex	성별
BMI	체질량 지수 (Body mass index)
BP	평균 혈압 (Average blood pressure)
S1	T-세포 (총 콜레스테롤 수치와 관련있음)
S2	저밀도 지단백질 (Low-density lipoproteins, LDL)
S3	고밀도 지단백질 (High-density lipoproteins, HDL)
S4	총 콜레스테롤 / HDL 비율
S5	혈중 트리글리세라이드 수치와 관련 있는 로그 변환된 라미티르
S6	혈당 수치

```
        age       sex       bmi        bp        s1        s2        s3  ₩
0  0.038076  0.050680  0.061696  0.021872 -0.044223 -0.034821 -0.043401
1 -0.001882 -0.044642 -0.051474 -0.026328 -0.008449 -0.019163  0.074412
2  0.085299  0.050680  0.044451 -0.005670 -0.045599 -0.034194 -0.032356
3 -0.089063 -0.044642 -0.011595 -0.036656  0.012191  0.024991 -0.036038
4  0.005383 -0.044642 -0.036385  0.021872  0.003935  0.015596  0.008142

         s4        s5        s6  target
0 -0.002592  0.019907 -0.017646   151.0
1 -0.039493 -0.068332 -0.092204    75.0
2 -0.002592  0.002861 -0.025930   141.0
3  0.034309  0.022688 -0.009362   206.0
4 -0.002592 -0.031988 -0.046641   135.0
```

마이닝 알고리즘 (머신러닝 모델) -2

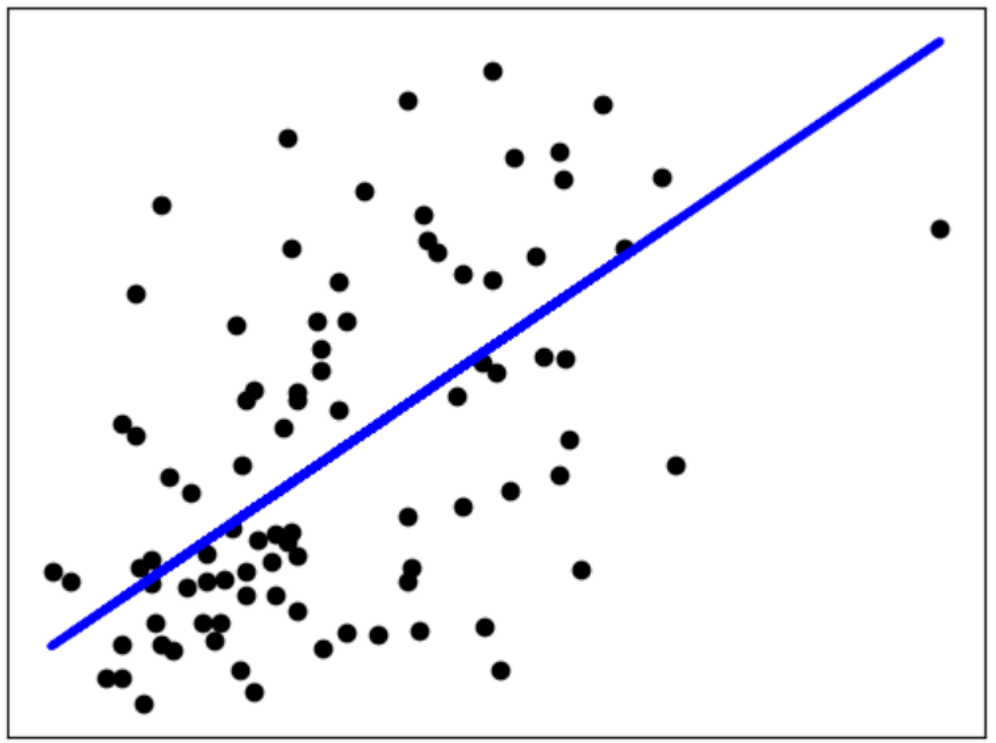
- **실습**

  2.05.regression.diabetes.ipynb

변수명	설명
Age	나이
Sex	성별
BMI	체질량 지수 (Body mass index)
BP	평균 혈압 (Average blood pressure)
S1	T-세포 (총 콜레스테롤 수치와 관련있음)
S2	저밀도 지단백질 (Low-density lipoproteins, LDL)
S3	고밀도 지단백질 (High-density lipoproteins, HDL)
S4	총 콜레스테롤 / HDL 비율
S5	혈중 트리글리세라이드 수치와 관련 있는 로그 변환된 라미티르
S6	혈당 수치

```
        age       sex       bmi        bp        s1        s2        s3  #
0  0.038076  0.050680  0.061696  0.021872 -0.044223 -0.034821 -0.043401
1 -0.001882 -0.044642 -0.051474 -0.026328 -0.008449 -0.019163  0.074412
2  0.085299  0.050680  0.044451 -0.005670 -0.045599 -0.034194 -0.032356
3 -0.089063 -0.044642 -0.011595 -0.036656  0.012191  0.024991 -0.036038
4  0.005383 -0.044642 -0.036385  0.021872  0.003935  0.015596  0.008142

         s4        s5        s6  target
0 -0.002592  0.019907 -0.017646   151.0
1 -0.039493 -0.068332 -0.092204    75.0
2 -0.002592  0.002861 -0.025930   141.0
3  0.034309  0.022688 -0.009362   206.0
4 -0.002592 -0.031988 -0.046641   135.0
```

```
Coefficients:
 [998.57768914]
Mean squared error: 4061.83
Coefficient of determination: 0.23
```



마이닝 알고리즘 (머신러닝 모델) -2

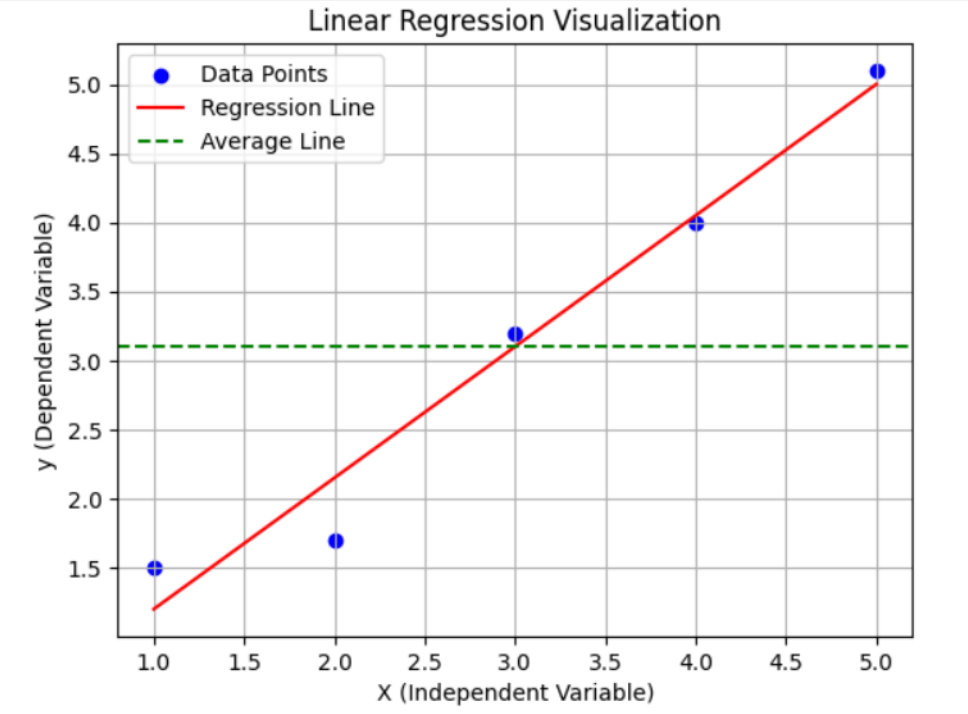
TASK 2.05.regression.diabetes.ipynb

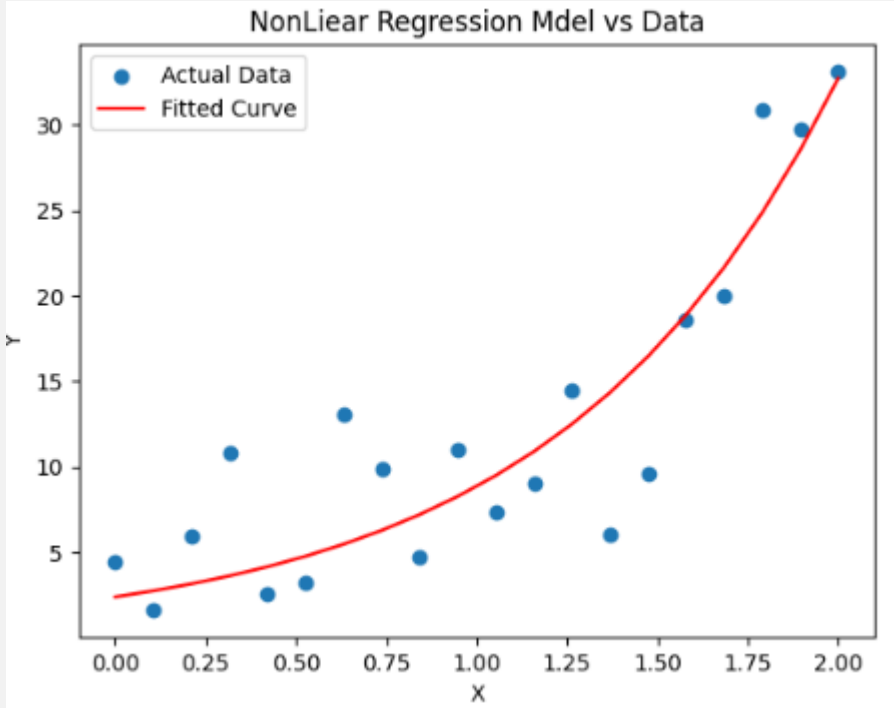
- 다른 변수 활용
- 다중회귀모델 학습

변수명	설명
Age	나이
Sex	성별
BMI	체질량 지수 (Body mass index)
BP	평균 혈압 (Average blood pressure)
S1	T-세포 (총 콜레스테롤 수치와 관련있음)
S2	저밀도 지단백질 (Low-density lipoproteins, LDL)
S3	고밀도 지단백질 (High-density lipoproteins, HDL)
S4	총 콜레스테롤 / HDL 비율
S5	혈중 트리글리세라이드 수치와 관련 있는 로그 변환된 라미티르
S6	혈당 수치

마이닝 알고리즘 (머신러닝 모델) -2

회귀 모델 - 선형 vs 비선형





마이닝 알고리즘 (머신러닝 모델) -2

회귀 모델 - 선형 vs 비선형

선형 회귀식 : $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_n X_n + \epsilon$

비선형 회귀식 : $Y = f(X, \beta) + \epsilon$

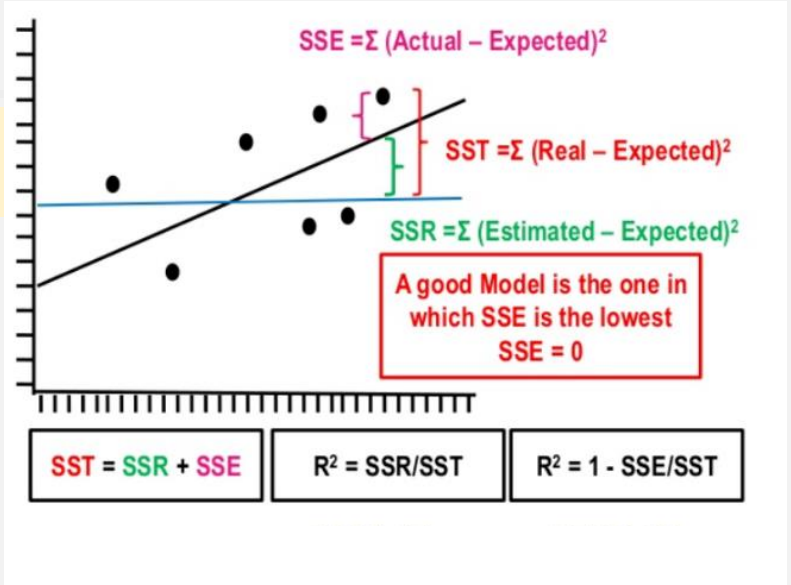
Ex,Y=β0+β1log(X)+ϵ

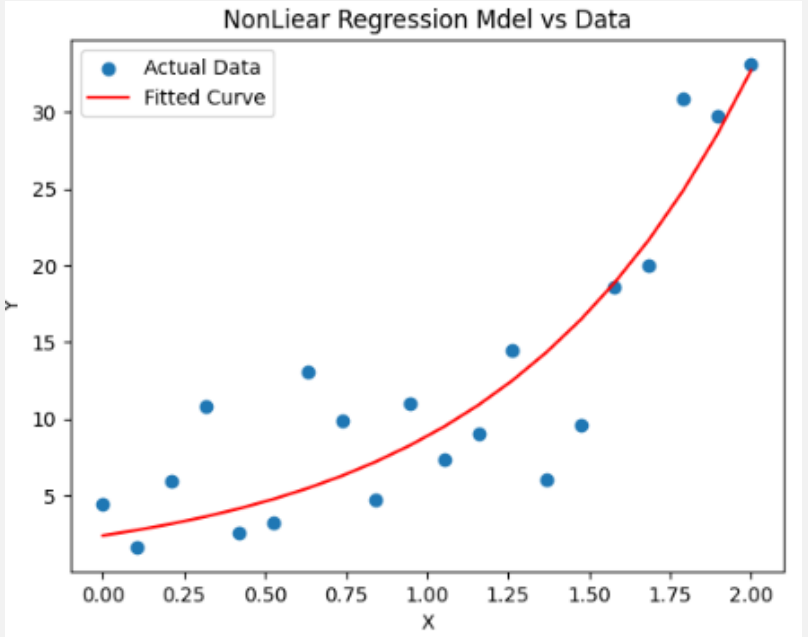
β_1 : X가 증가할 때 Y가 증가, 증가율이 점점 감소
<- 회귀 계수의 직접적 해석 어려움

마이닝 알고리즘 (머신러닝 모델) -2

비선형

- SSR(회귀제곱합)
- SSE(오차제곱합)
- SST(총제곱합)



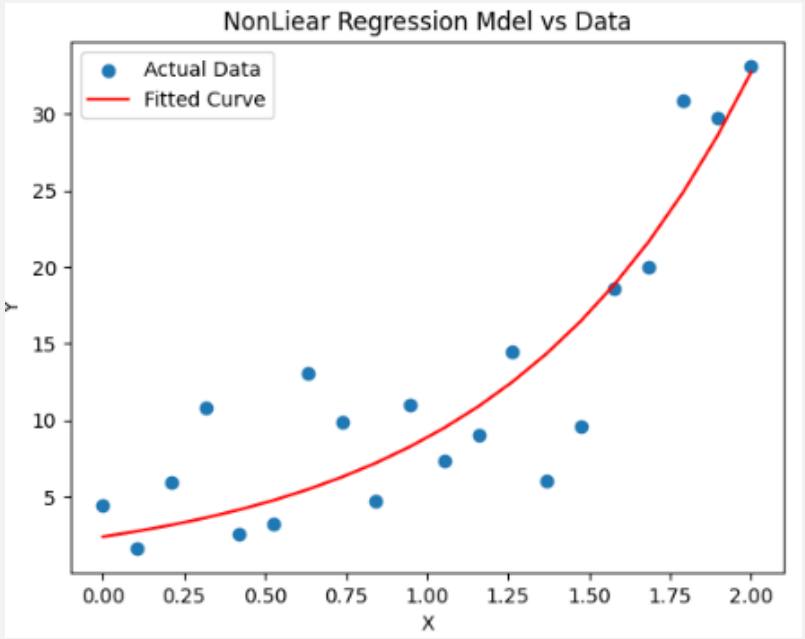


마이닝 알고리즘 (머신러닝 모델) -2

비선형

- **실습**

  2.05.Nonlinear.Regression.ipynb



마이닝 알고리즘 (머신러닝 모델) -2

- **실습**

  2.05.Regression.nn.house.price.ipynb

  2.05.Regression.sklearn.house.price.ipynb

마이닝 알고리즘 (머신러닝 모델) -2

- **실습 : NN으로 구현한 Regression**

  2.05.Regression.nn.house.price.ipynb

$$\hat{y} = w_1 \cdot x_1 + w_2 \cdot x_2 + b$$

선형 회귀(Linear Regression) 모델

활성화 함수

 # deeplearning.ActivationFunction.4th.pptx

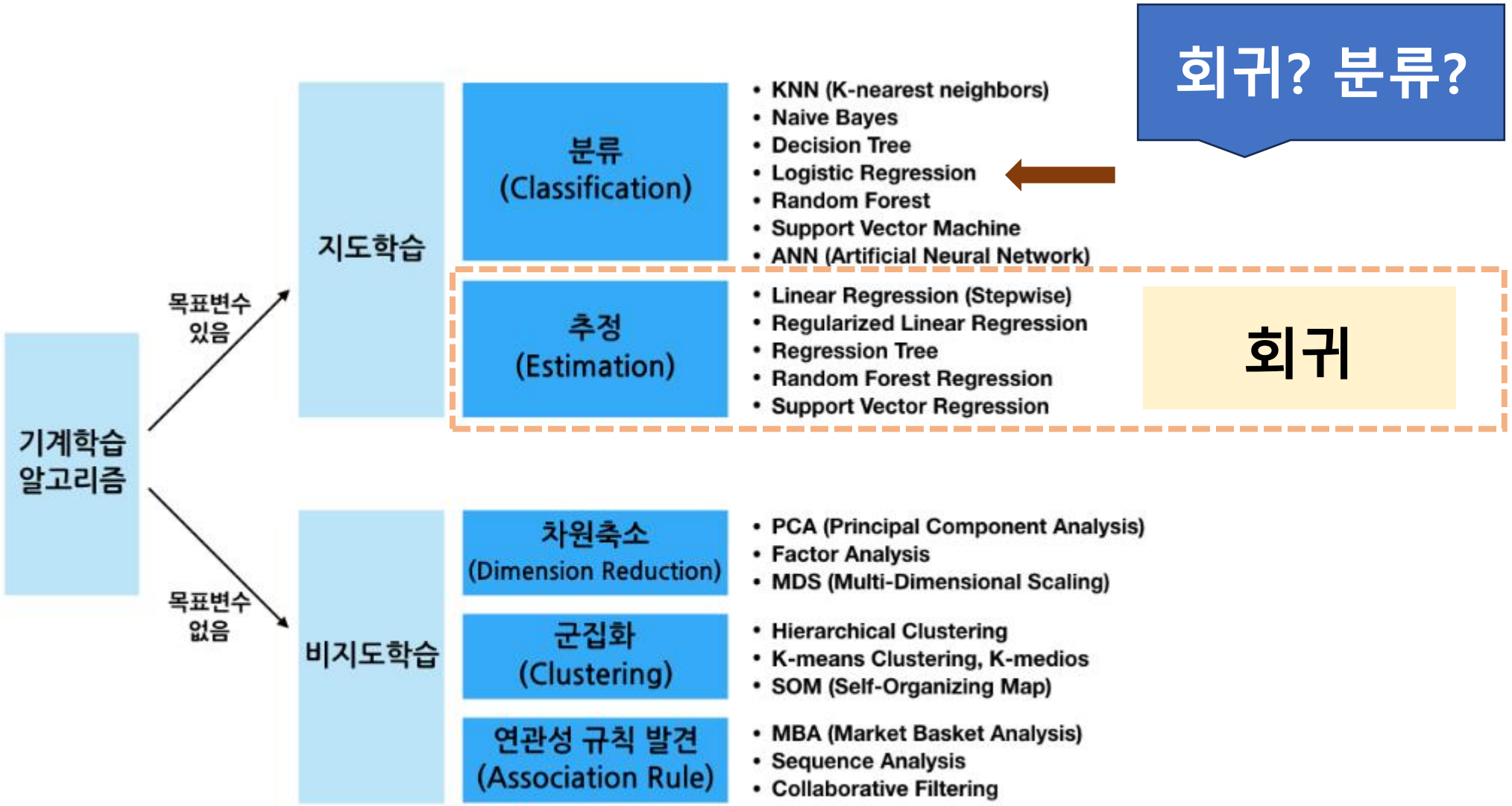
오류 역전파

4.03.NN_4th.pptx

Logistic Regression

독립변수의 선형 결합을 이용하여 사건의 발생 가능성을 예측하는 데 사용되는 통계 기법

마이닝 알고리즘 (머신러닝 모델) -2



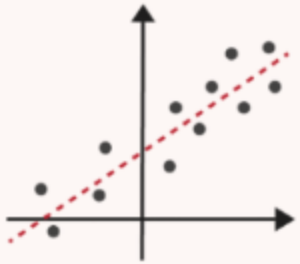
https://dbrang.tistory.com/1520

마이닝 알고리즘 (머신러닝 모델) -2

Regression – Linear, Logistic

Linear regression

- Econometric modeling
- Marketing mix model
- Customer lifetime value



Continuous > Continuous

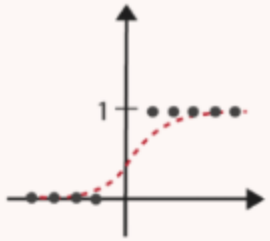
$$y = a_0 + \sum_{i=1}^{N} a_i x_i$$

lm(y-x1 + x2, data)

1 unit increase in
x increases y by a

Logistic regression

- Customer choice model
- Click-through rate
- Conversion rate
- Credit scoring



Continuous > True/False

$$y = \frac{1}{1 + e^{-z}}$$

$$z = a_0 + \sum_{i=1}^{N} a_i x_i$$

glm(y-x1 + x2, data), family = binomial())

1 unit increase in x
increases log odds by a

https://datasciencedojo.com/blog/linear-regression-vs-logistic-regression/

마이닝 알고리즘 (머신러닝 모델) -2

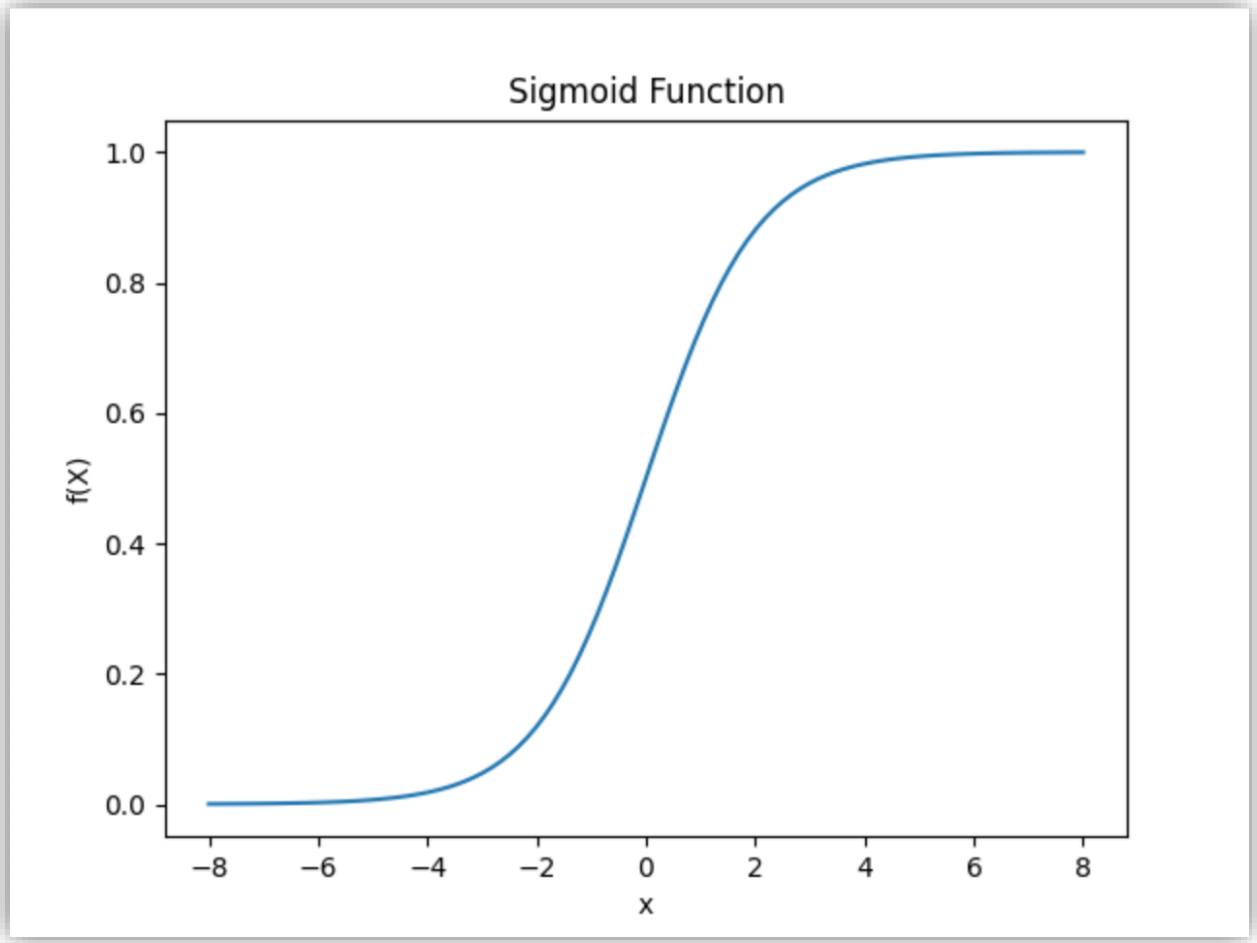
Regression – Linear, Logistic

- **선형 회귀 (Linear Regression)**
  - 연속형 값 예측 ex, 주택 가격, 체중, 온도 등의 값을 예측
  - **모델**: 두 변수 사이의 **선형 관계**를 가정
    독립 변수 X가 변할 때 종속 변수 Y가 일정 비율로 변한다는 가정을 따름
  - **함수**: $Y=b_0+b_1*X$ (b0 절편, b1기울기(회귀 계수))
  - **오차 측정**: 평균 제곱 오차(Mean Squared Error, MSE)로 예측값과 실제값 사이의 차이를 최소화하는 방식으로 학습

- **로지스틱 회귀 (Logistic Regression):**
  - **분류 문제**에 사용 (주로 이진 분류(Binary Classification) 문제)
    출력 값은 특정 클래스에 속할 확률
  - **출력 값**: 확률 출력, 이 확률에 따라 데이터를 특정 클래스(예: 0 또는 1)로 분류
  - **모델**: 두 변수 사이의 **비선형 관계**를 가정. 즉, X가 변할 때 Y가 일정하지 않은 방식으로 변경
  - **함수**: $$P(Y=1|X)=\frac{1}{1+e^{-(b_0+b_1X)}}$$ (P는 Y=1일 확률, e는 자연 로그의 밑)
  - **오차 측정**: 로스 함수(로그 손실 함수 또는 교차 엔트로피)를 사용 모델의 성능을 평가하고 최적화

마이닝 알고리즘 (머신러닝 모델) -2

Regression – Linear, Logistic



$$f(x) = \frac{1}{1 + e^{-x}}$$

e: 자연 로그의 밑(약 2.718)
x: 입력 값

마이닝 알고리즘 (머신러닝 모델) -2

Regression – Linear, Logistic

- **참고**

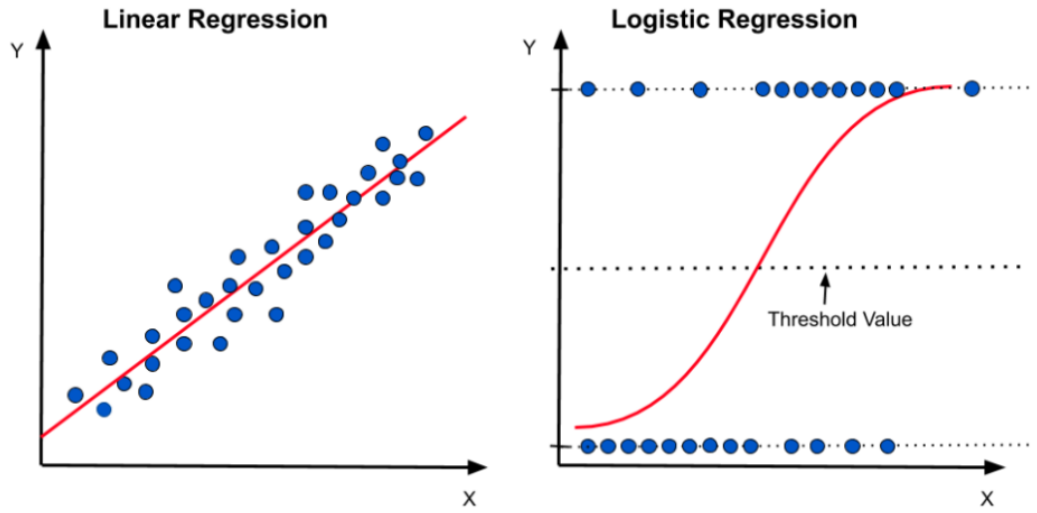
  2.05.Classification.Regression.ipynb

마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

- ✓ 분류 문제를 해결하기 위해 사용되는 통계적 모델링 방법
- ✓ 특히 이진 분류 문제에 자주 사용되며, 두 개의 클래스(예: 0과 1) 중 하나로 결과를 예측
- ✓ 선형 회귀와 유사 but, 종속 변수가 범주형 데이터일 때 사용
- ✓ 독립 변수의 선형 조합을 사용하여 사건의 발생 확률을 추정
- ✓ 모델의 출력은 로지스틱 함수(또는 시그모이드 함수)를 통해 0과 1 사이의 값으로 제한
- ✓ 함수의 출력값은 주어진 입력 데이터가 특정 클래스에 속할 확률을 나타냄

독립 변수의 값이 증가하면
종속 변수의 값도 증가/감소



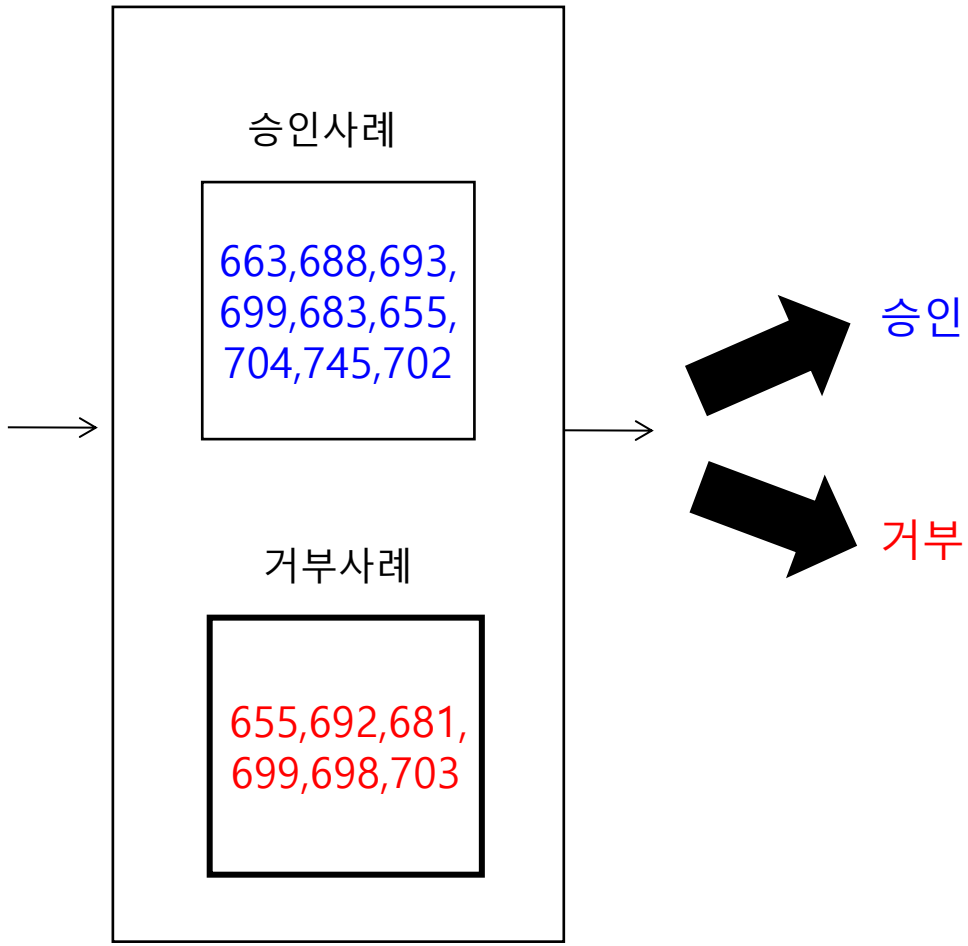
독립변수 값이 크게 변해도
종속 변수는 0과 1사이의 값

마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

- ✓ 분류 문제를 해결하기 위해 사용되는 통계적 모델링 방법
- ✓ 특히 이진 분류 문제에 자주 사용되며, 두 개의 클래스(예: 0과 1) 중 하나로 결과를 예측

ex) 대출 승인



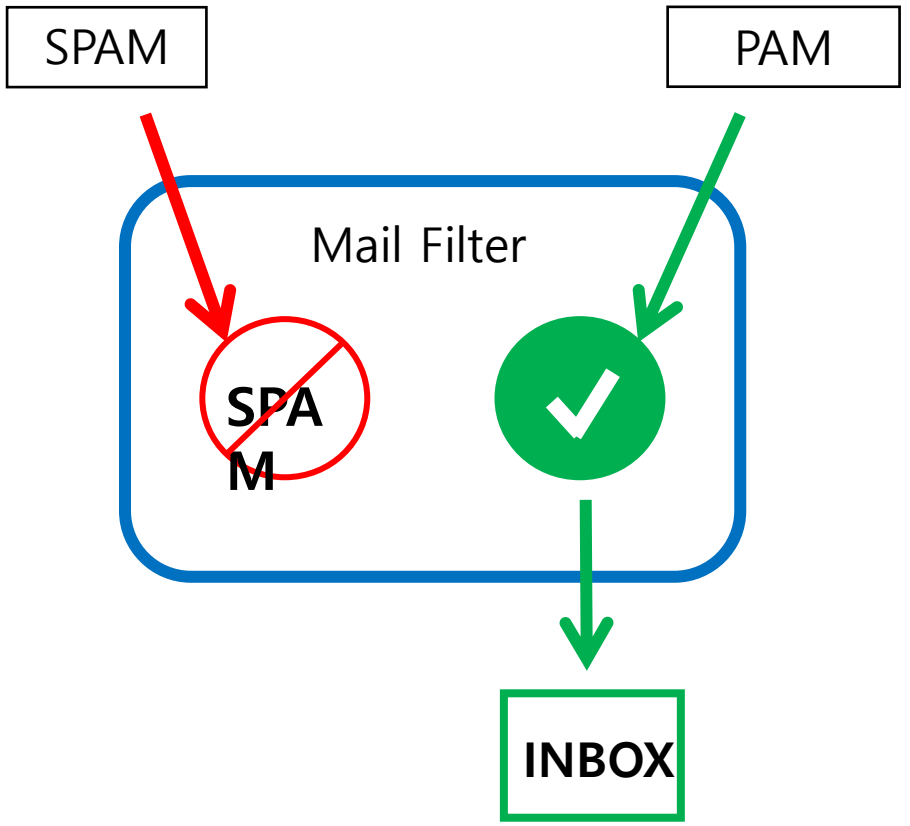
마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

- ✓ 분류 문제를 해결하기 위해 사용되는 통계적 모델링 방법
- ✓ 특히 이진 분류 문제에 자주 사용되며, 두 개의 클래스(예: 0과 1) 중 하나로 결과를 예측

ex) Spam 필터링

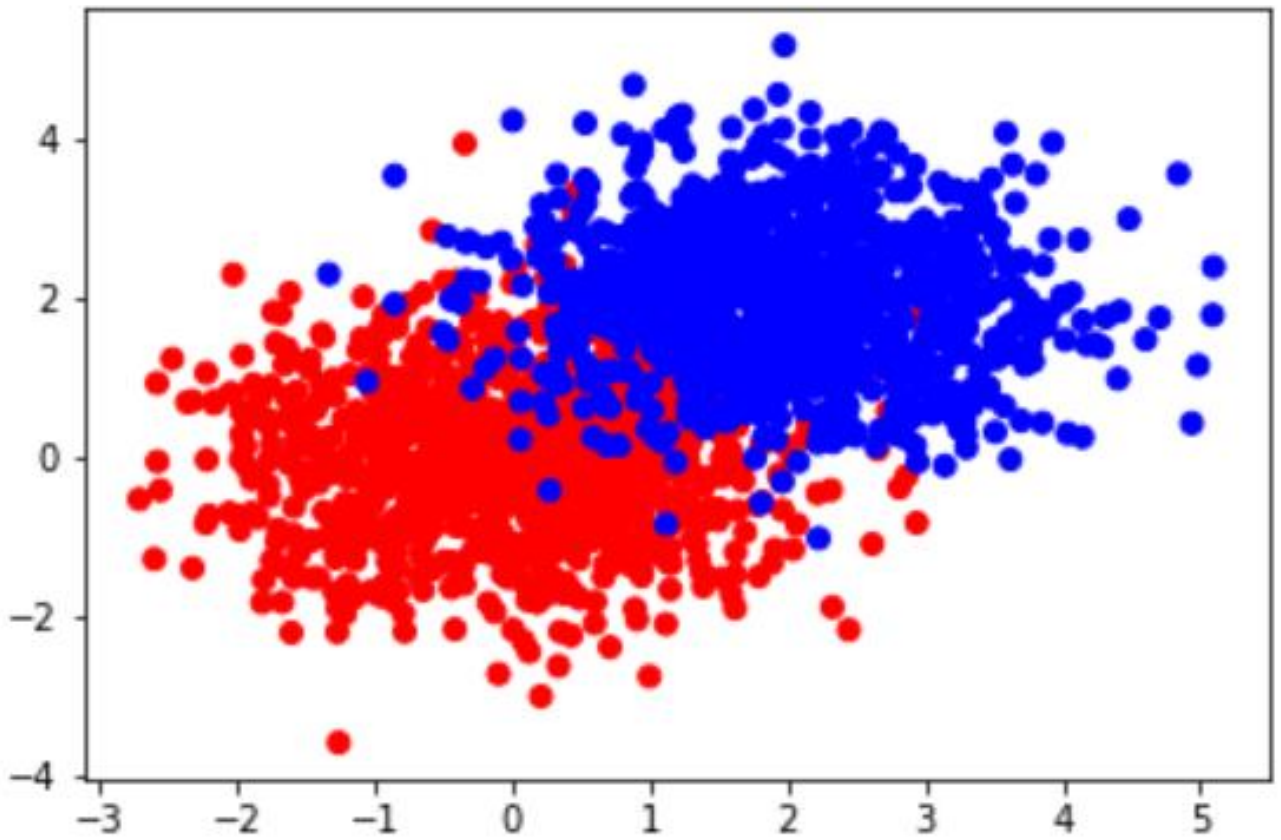
	SPAM	PAM
Free	8	2
Cash	8	1
Can	2	9
Won	7	2
Never	10	1
Chance	9	3
Sorry	0	8
Need	2	9
Click	10	0
Can	3	8



마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

2진 분류



마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

로지스틱 함수

시그모이드 함수(Sigmoid function):

- ✓ 입력 값을 받아서 출력 값을 0과 1 사이의 확률로 변환
- ✓ 선형 회귀 모델의 출력을 확률로 해석

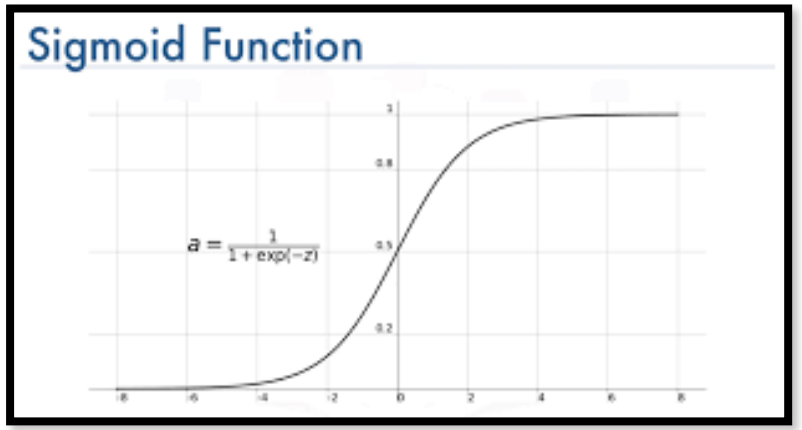
$$sigmoid(x) = \frac{1}{1 + e^{-x}}$$

마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

시그모이드 함수(Sigmoid function):

- ✓ **입력 값을 받아서 출력 값을 0과 1 사이의 확률로 변환**
- ✓ **선형 회귀 모델의 출력을 확률로 해석**



$$sigmoid(x) = \frac{1}{1+e^{-x}}$$

- **형태 곡선**:

  > S 형태의 곡선

  > 함수의 입력값이 양의 무한대로 가면 출력값이 1에 수렴,
  입력값이 음의 무한대로 가면 출력값이 0에 수렴

- **결정 경계**:

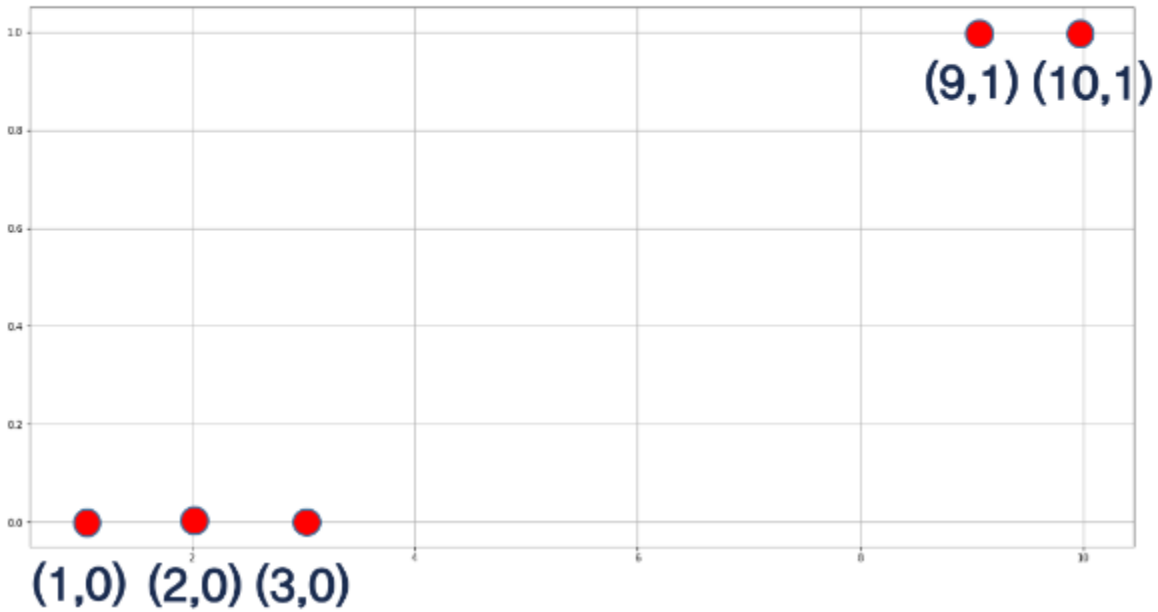
  > 로지스틱 회귀에서는 시그모이드 함수의 출력값이 0.5 이상이면 해당 데이터를 클래스 1로,
  그렇지 않으면 클래스 0으로 분류 ($\sigma(z)$=0.5 <- 결정 경계(decision boundary))

마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

- **이진 분류 모델링**
  로지스틱 회귀를 통해 새로운 X 값에 대해 Y 값이 0(Fail)인지 1(Pass)인지 예측
  두 클래스(0과 1)가 X 값에 따라 구분

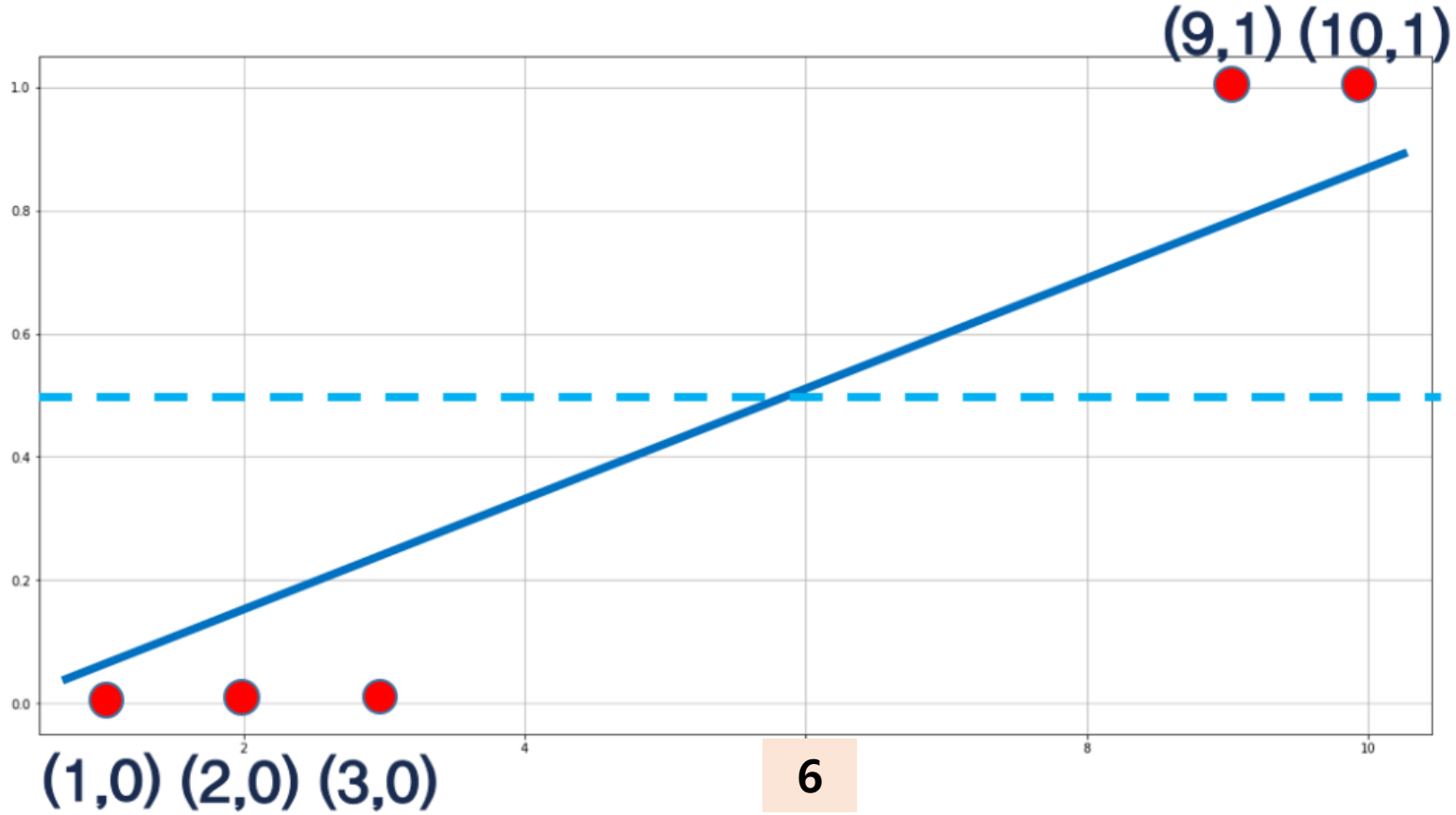
X 데이터	Y 데이터
10	Pass (1)
9	Pass (1)
3	Fail (0)
2	Fail (0)
1	Fail (0)
5	???



마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

- **선형 분류**
  예측 값이 직선 형태로 나타남
  0과 1 사이를 벗어나는 결과

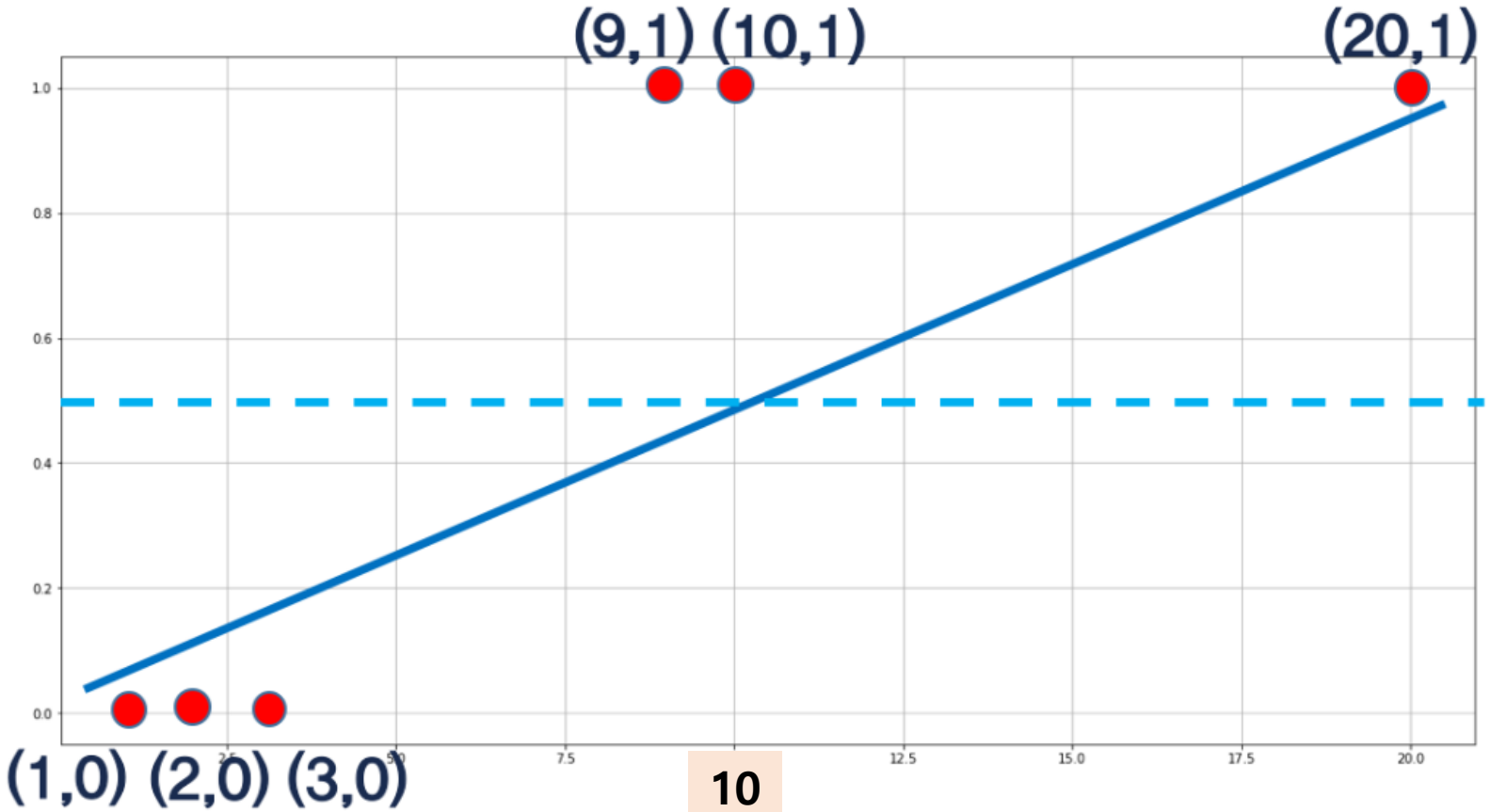


마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

- **선형 분류**
  예측 값이 직선 형태로 나타남
  0과 1 사이를 벗어나는 결과 – 어떻게???

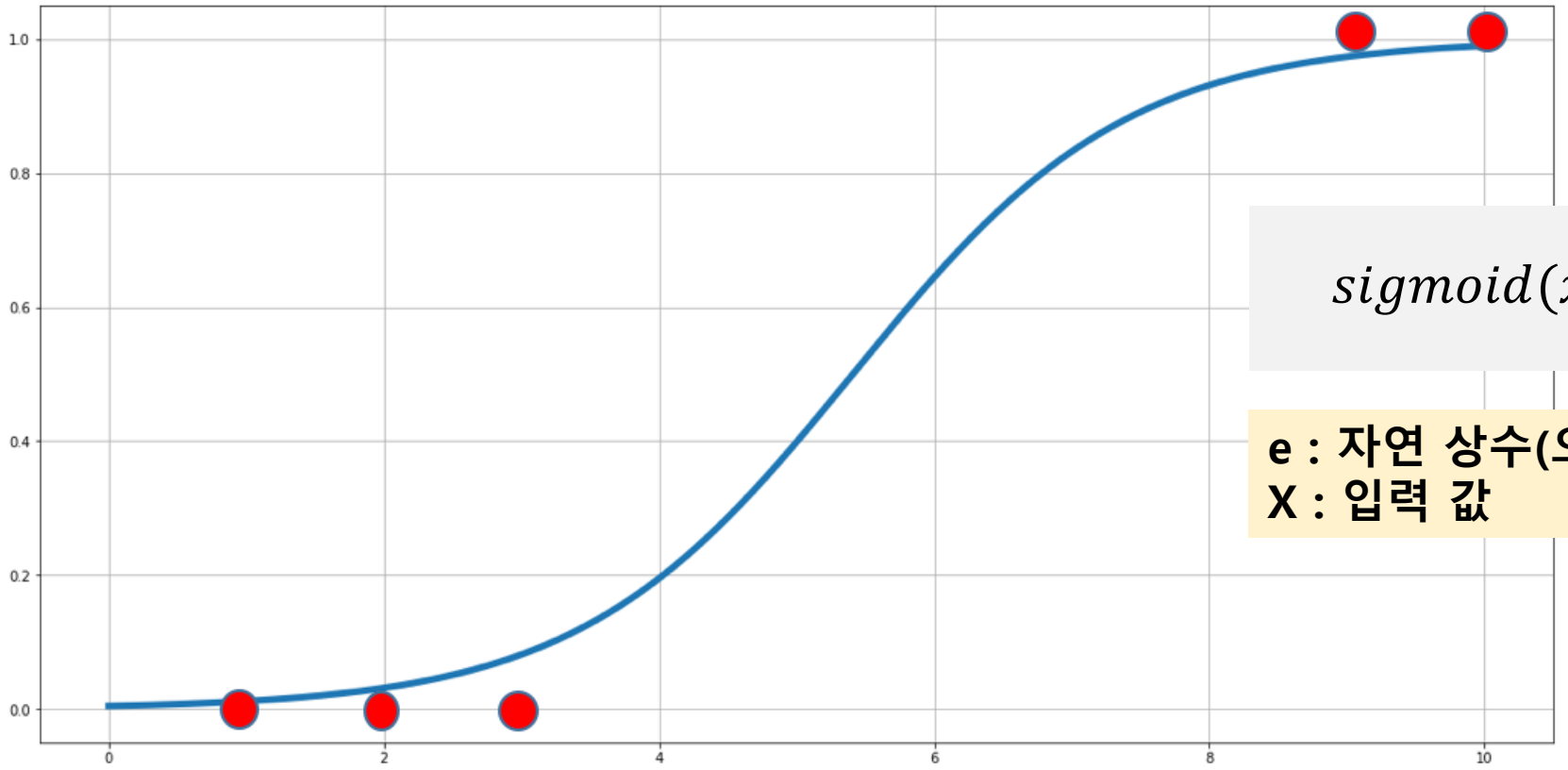
X 데이터	Y 데이터
10	Pass (1)
9	Pass (1)
3	Fail (0)
2	Fail (0)
1	Fail (0)
5	???



마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

- **시그모이드 함수**
  시그모이드 함수를 적용 출력을 0과 1 사이의 값으로 제한
  X 값이 커질수록 Y 값이 1에 가까워지며, X 값이 작을수록 Y 값이 0에 가까워지는 비선형 관계를 형성



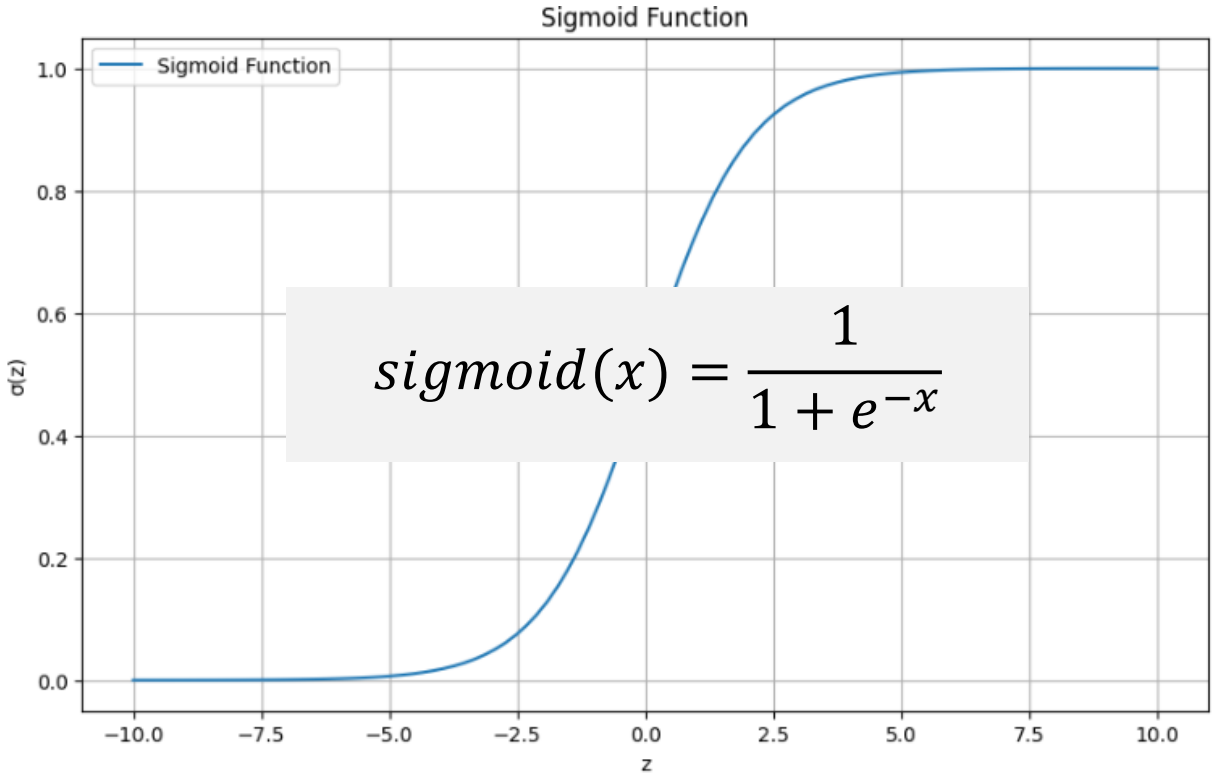
$$sigmoid(x) = \frac{1}{1 + e^{-x}}$$

e : 자연 상수(오일러 수, 약 2.718)
X : 입력 값

마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

- **시그모이드 함수**
  시그모이드 함수를 적용 출력을 0과 1 사이의 값으로 제한
  X 값이 커질수록 Y 값이 1에 가까워지며, X 값이 작을수록 Y 값이 0에 가까워지는 비선형 관계를 형성



$$sigmoid(x) = \frac{1}{1 + e^{-x}}$$

```python
import numpy as np
import matplotlib.pyplot as plt

# 시그모이드 함수 정의
def sigmoid(z):
    return 1 / (1 + np.exp(-z))

# 입력값 z의 범위를 정의 (-10에서 10까지)
z = np.linspace(-10, 10, 100)

# 시그모이드 함수 적용
sigma_z = sigmoid(z)

# 시그모이드 함수 그래프 그리기
plt.figure(figsize=(10, 6))
plt.plot(z, sigma_z, label='Sigmoid
Function')
plt.title('Sigmoid Function')
plt.xlabel('z')
plt.ylabel('σ(z)')
plt.grid(True)
plt.legend()
plt.show()
```

마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

- **실습**

  2.05.03.logi.titanic.ipynb

  2.05.03. logi.bc.p.ipynb

```
data = load_breast_cancer()
```

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points	mean symmetry	mean fractal dimension	...	worst radius	worst texture	worst perimeter	worst area
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.2419	0.07871	...	25.38	17.33	184.60	2019.0
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812	0.05667	...	24.99	23.41	158.80	1956.0
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.2069	0.05999	...	23.57	25.53	152.50	1709.0
3	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.2597	0.09744	...	14.91	26.50	98.87	567.7
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.1809	0.05883	...	22.54	16.67	152.20	1575.0

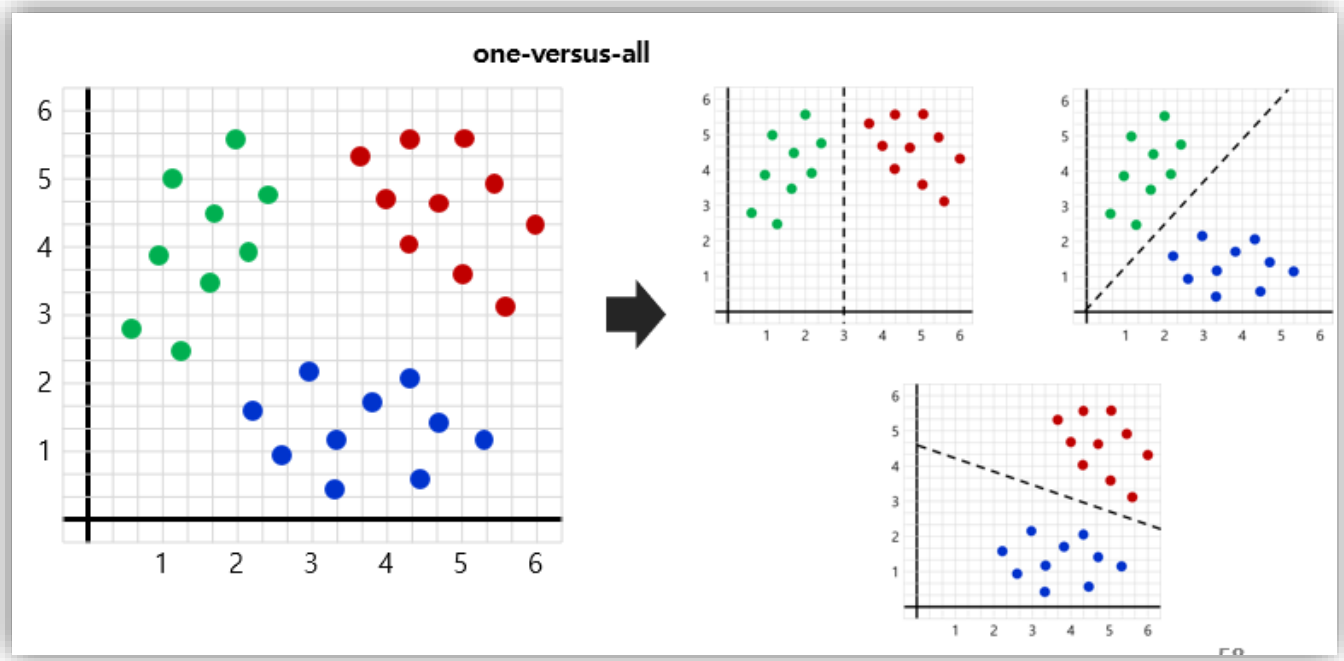
	target
0	0
1	0
2	0
3	0
4	0

마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

- **실습 - 심화**

  2.05.03.one_v_all_iris.ipynb



- **실습 - 심화**

  2.05.03.logi_nonlinear.ipynb

마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

- 모델 평가

예측 \ 실제	Positive	Negative
Positive	True Positive	False Positive
Negative	False Negative	True Negative

Accuracy(정확도) :
분류 모델의 평가 지표

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

True Positive: 실제 Positive인 정답을 Positive라고 예측 (True)
True Negative: 실제 Negative인 정답을 Negative라고 예측 (True)
False Positive: 실제 Negative인 정답을 Positive라고 예측 (False) – Type I error
False Negative: 실제 Positive인 정답을 Negative라고 예측 (False) – Type II error

마이닝 알고리즘 (머신러닝 모델) -2

Regression – Logistic

- **모델 평가**

4.01.모델평가.Model.Evaluation.4th.pdf

마이닝 알고리즘 (머신러닝 모델) -2

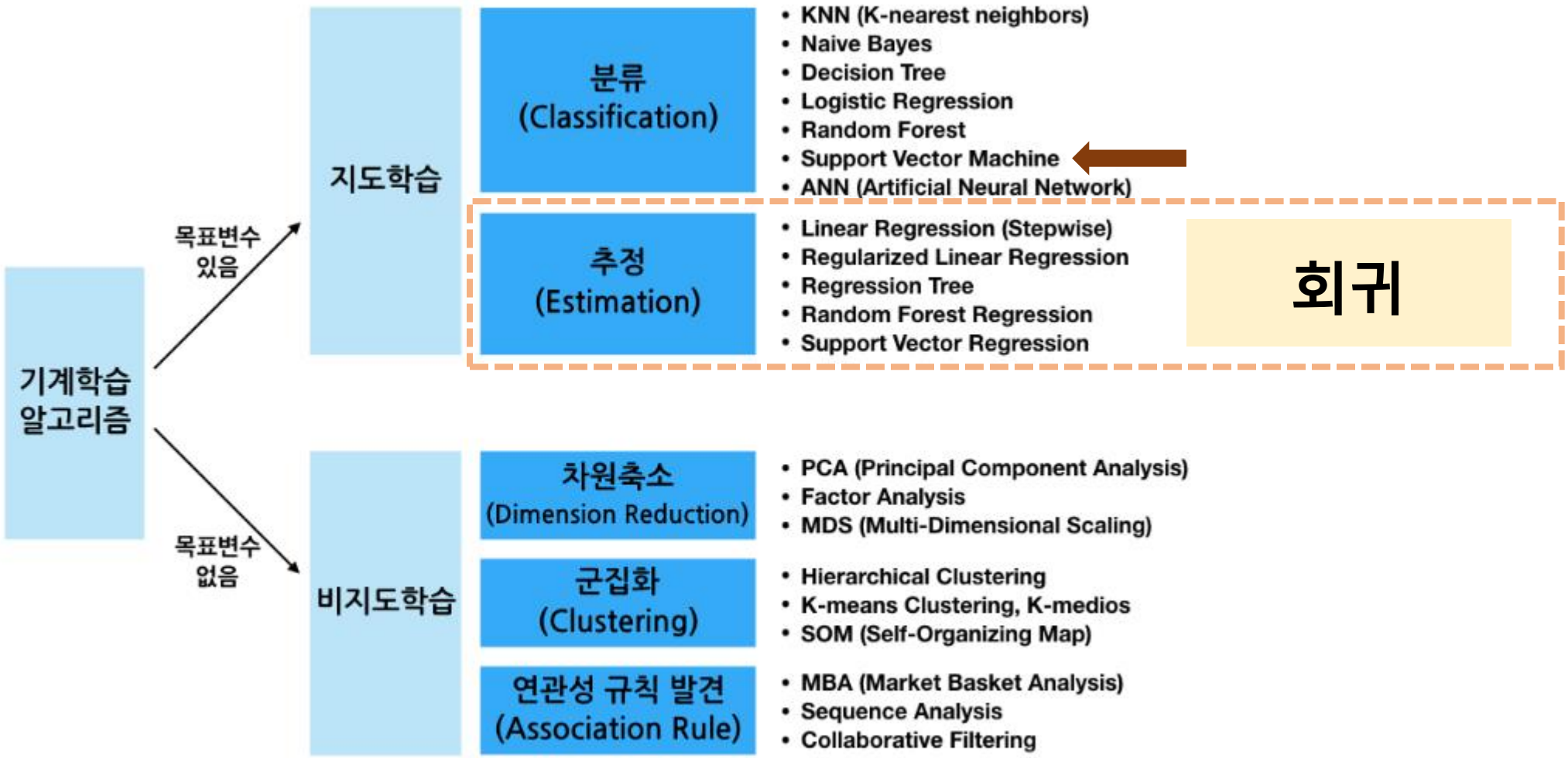
Regression – Logistic

- **모델 평가**

Algorithms for the Travelling Salesman Problem

SVM

마이닝 알고리즘 (머신러닝 모델) -2



https://dbrang.tistory.com/1520

마이닝 알고리즘 (머신러닝 모델) -2

SVM

서포트 벡터 머신(SVM, Support Vector Machine)

✓ 지도 학습의 일종으로 분류(classification), 회귀(regression) 문제에 사용될 수 있는 강력하고 유연한 머신러닝 모델

✓ SVM 모델링은 데이터 포인트를 **고차원 공간**으로 매핑하고, 이 공간에서 분류를 가장 잘 할 수 있는 **결정 경계(decision boundary)**인 초평면(hyperplane)을 찾는 과정

✓ 이 결정 경계는 **서포트 벡터(support vectors)**라는, 서로 다른 클래스의 가장 가까운 데이터 포인트들에 의해 결정

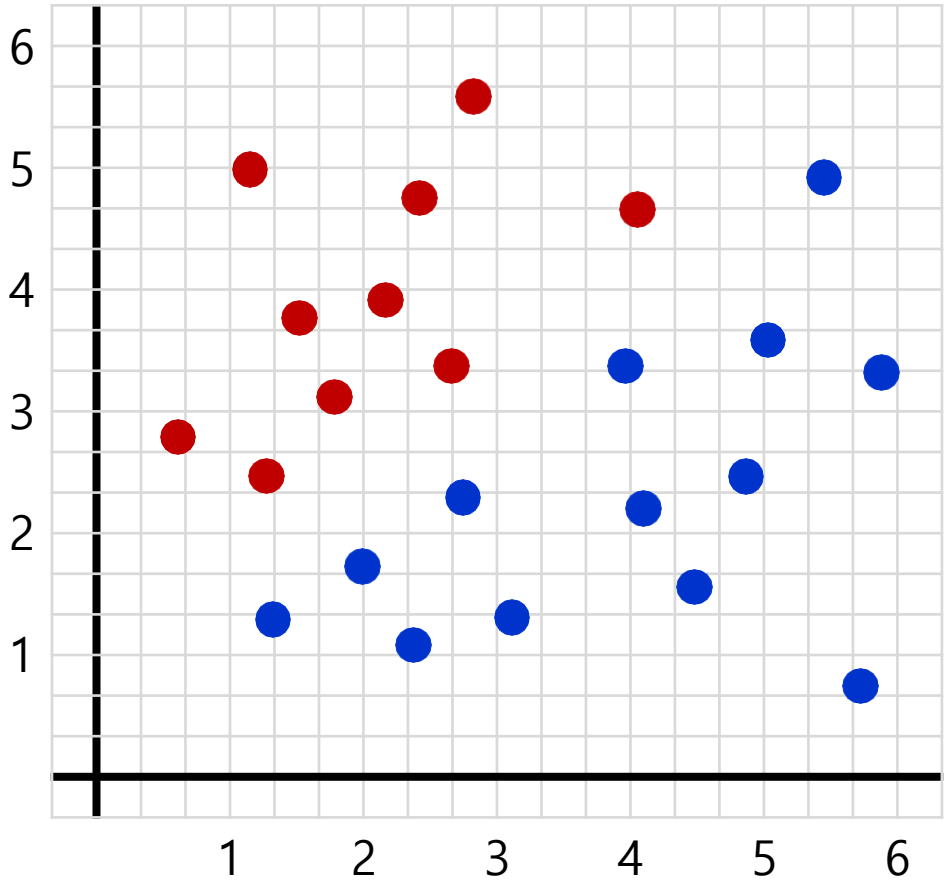
SVM의 주요 개념

- **서포트 벡터(support vectors):**
  두 클래스 사이의 경계에 위치한 데이터 포인트들, 결정 경계의 위치와 방향을 정의하는 데 사용
- **결정 경계(decision boundary):** 서로 다른 클래스를 분리하는 초평면.
  SVM 모델은 이 결정 경계를 최적화하여 클래스 사이의 마진을 최대화
- **마진(margin):** 결정 경계와 가장 가까운 서포트 벡터 사이의 거리
  SVM은 이 마진을 최대화하는 결정 경계를 찾는 과정

마이닝 알고리즘 (머신러닝 모델) -2

SVM

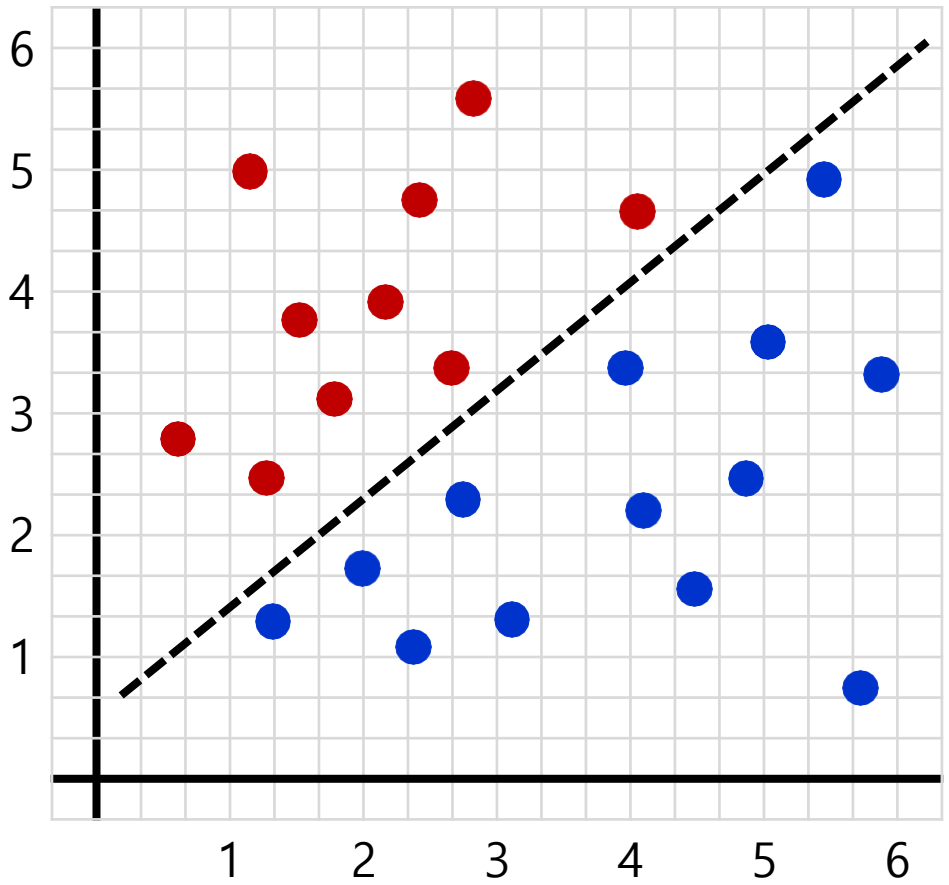
Support Vector Machin?



마이닝 알고리즘 (머신러닝 모델) -2

SVM

Support Vector Machin?



Find best linear classifier(i.e. hyper plane)

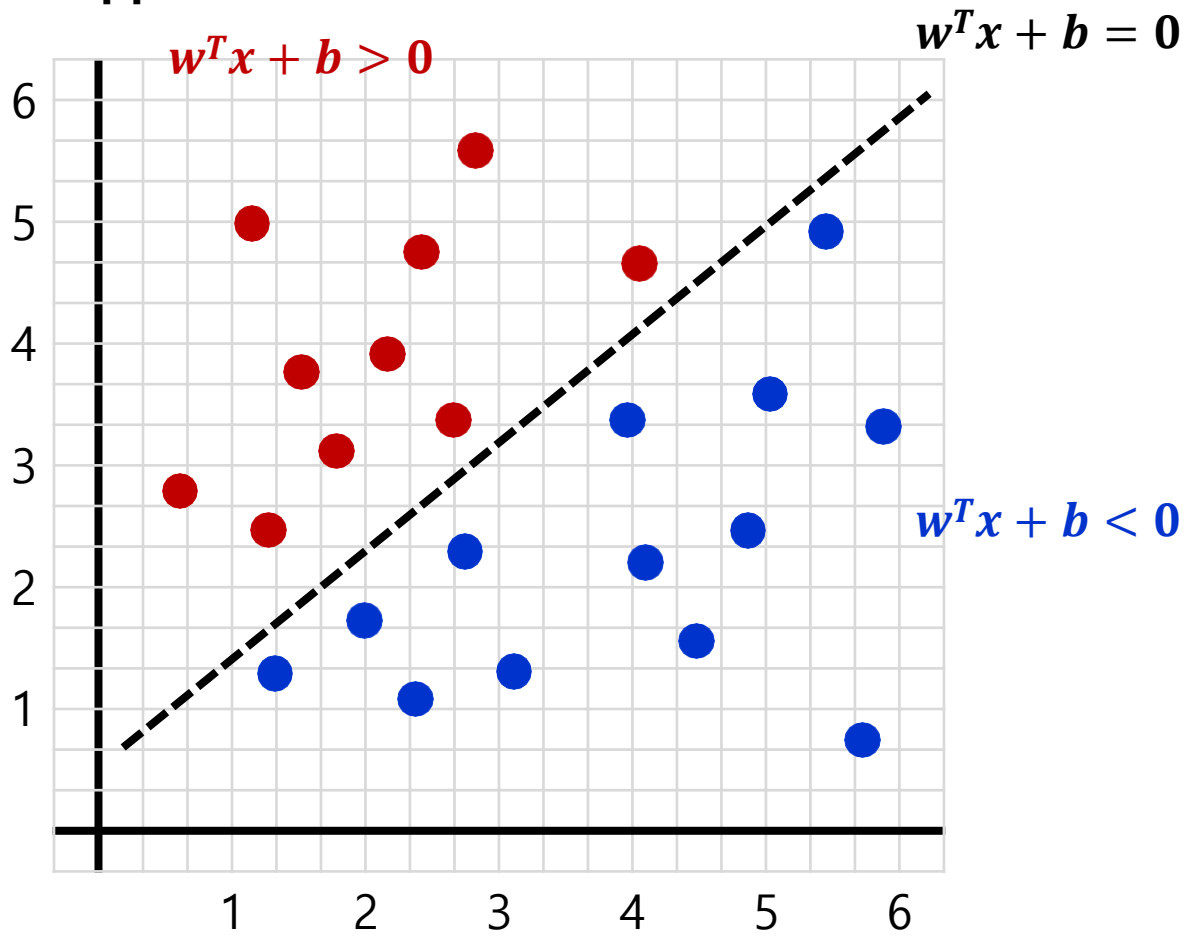
마이닝 알고리즘 (머신러닝 모델) -2

SVM

- w: 특성 공간에서의 결정 경계의 방향을 나타내는 가중치 벡터
- x: 입력 데이터의 특성 벡터
- b: 편향(bias)으로서 결정 경계의 위치를 조정
- $w^T x + b$: w 벡터와 x 벡터의 내적에 편향을 더한 값

결정 경계는 $w^T x + b = 0$ 로 정의
입력 데이터 x가 결정 경계 위에 있을 때의 조건

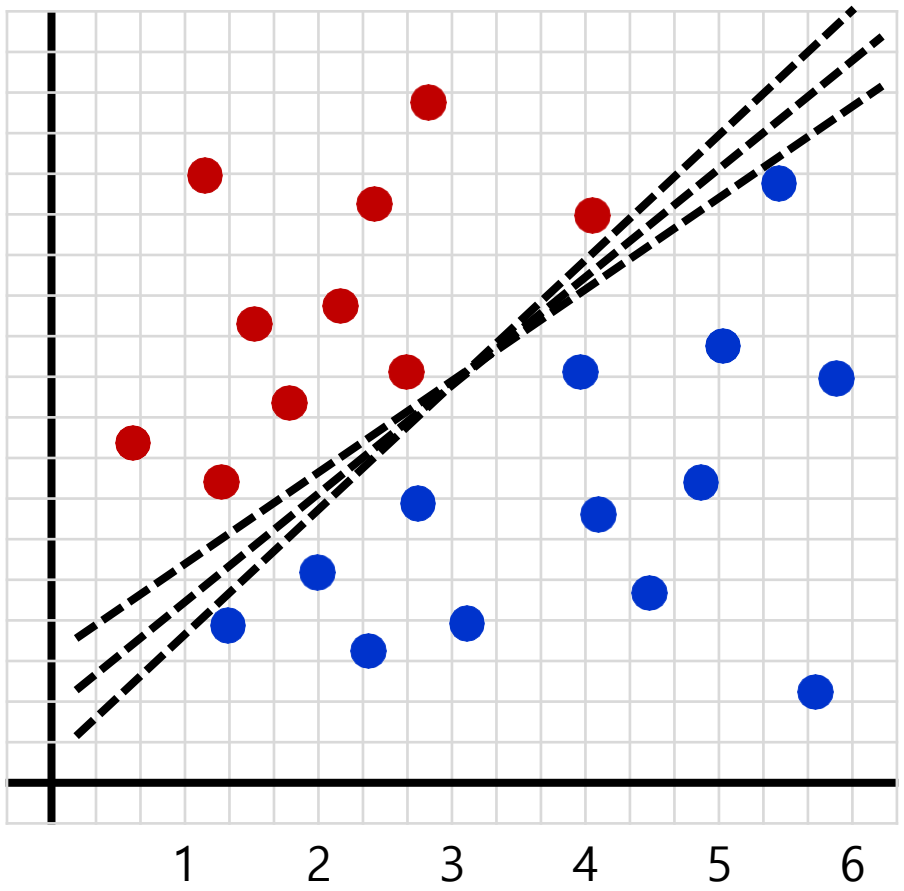
Support Vector Machin?



Find best linear classifier(i.e. hyper plane)

마이닝 알고리즘 (머신러닝 모델) -2

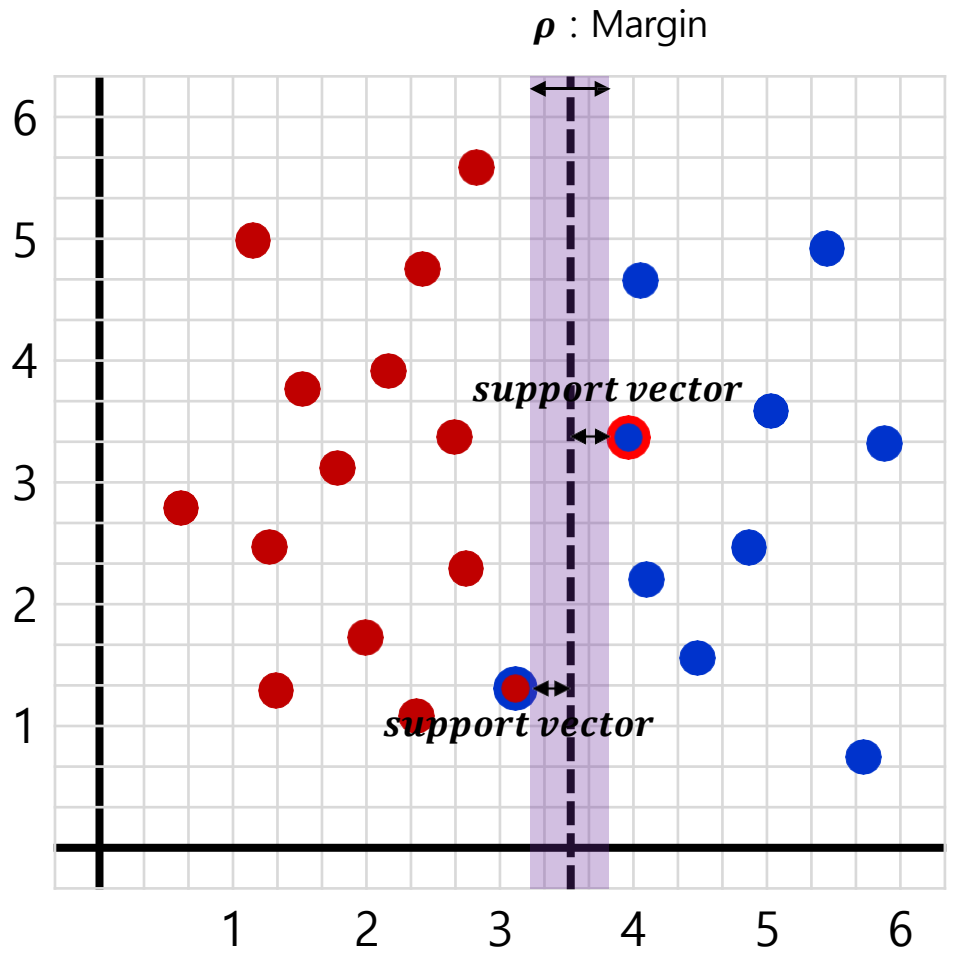
SVM



Which hyper plane is the best classifier?

마이닝 알고리즘 (머신러닝 모델) -2

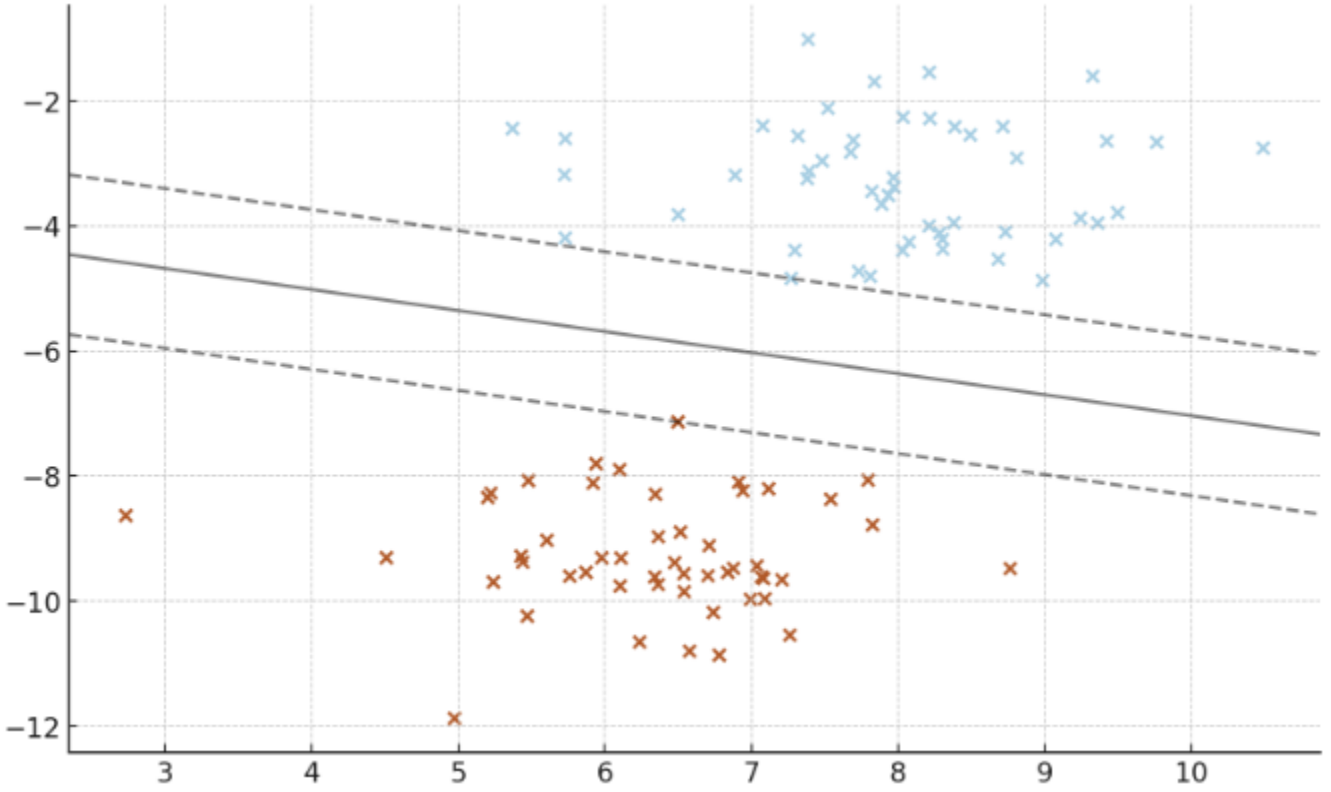
SVM



closest to the hyperplane are termed *support vector*

마이닝 알고리즘 (머신러닝 모델) -2

SVM



1 2 3 4 5 6

closest to the hyperplane are termed $support\ vector$

마이닝 알고리즘 (머신러닝 모델) -2

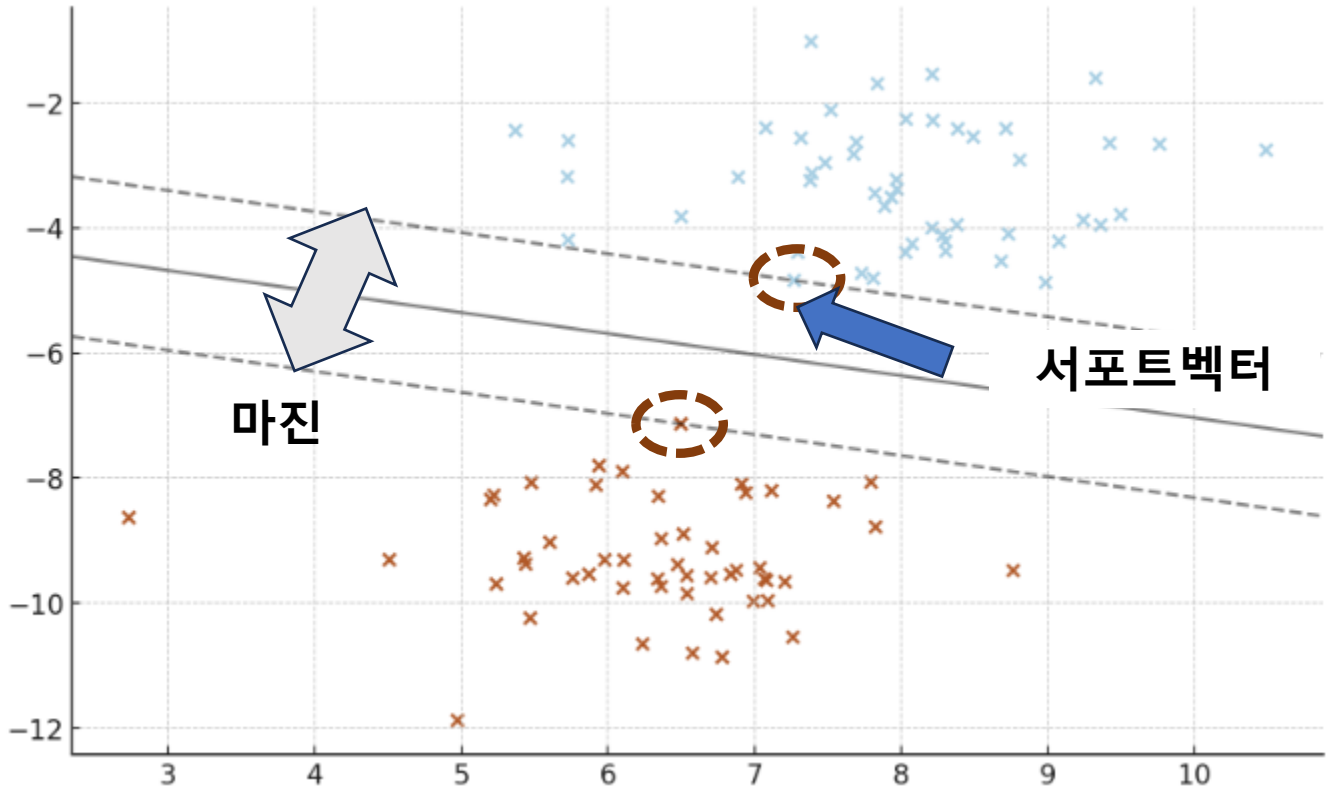
SVM

2_05_02_svm.plot.ipynb

```
from sklearn import datasets
from sklearn.svm import SVC
import numpy as np
import matplotlib.pyplot as plt

# 임의의 데이터셋 생성
X, y = datasets.make_blobs(n_samples=100, centers=2, random_state=6)

# SVM 분류기 모델 생성 및 학습
clf = SVC(kernel='linear', C=1000)
clf.fit(X, y)

# 데이터 포인트와 결정 경계 시각화
plt.scatter(X[:, 0], X[:, 1], c=y, s=30, cmap=plt.cm.Paired)

# 결정 경계
ax = plt.gca()
xlim = ax.get_xlim()
ylim = ax.get_ylim()

# 그리드 생성
xx = np.linspace(xlim[0], xlim[1], 30)
yy = np.linspace(ylim[0], ylim[1], 30)
YY, XX = np.meshgrid(yy, xx)
xy = np.vstack([XX.ravel(), YY.ravel()]).T
Z = clf.decision_function(xy).reshape(XX.shape)

# 결정 경계와 마진 플로팅
ax.contour(XX, YY, Z, colors='k', levels=[-1, 0, 1], alpha=0.5,
           linestyles=['--', '-', '--'])

# 서포트 벡터 플로팅
ax.scatter(clf.support_vectors_[:, 0], clf.support_vectors_[:, 1], s=100,
           linewidth=1, facecolors='none', edgecolors='k')

plt.show()
```

마이닝 알고리즘 (머신러닝 모델) -2

SVM



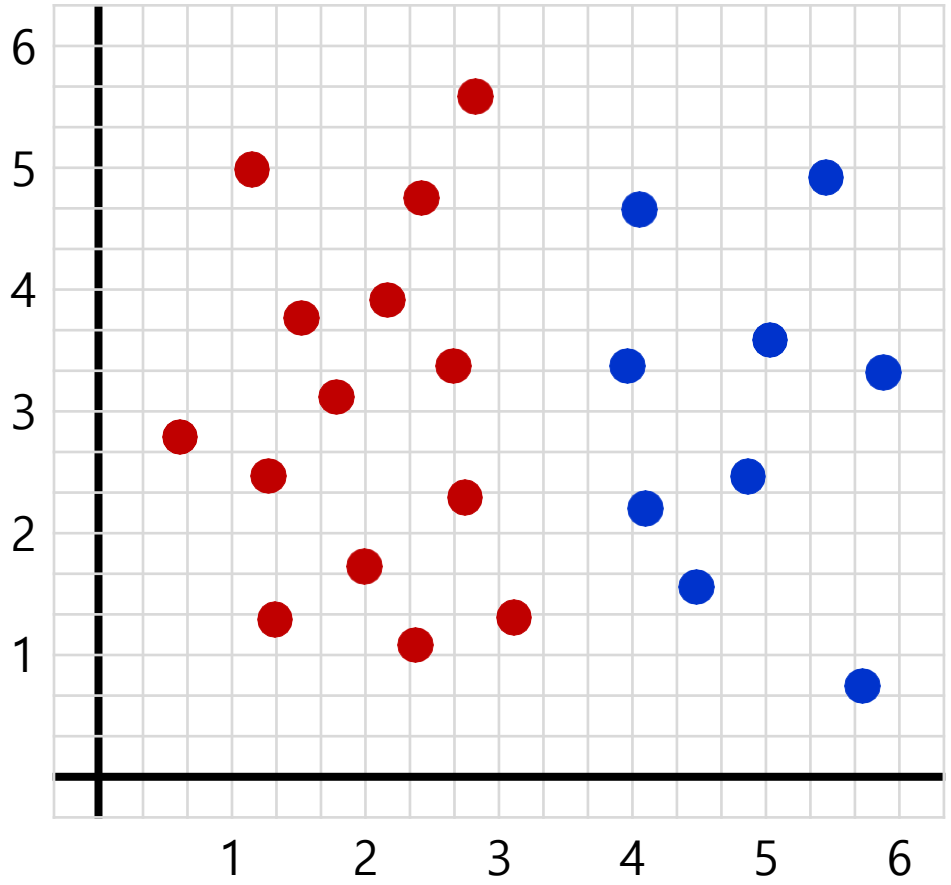
1 2 3 4 5 6

closest to the hyperplane are termed *support vector*

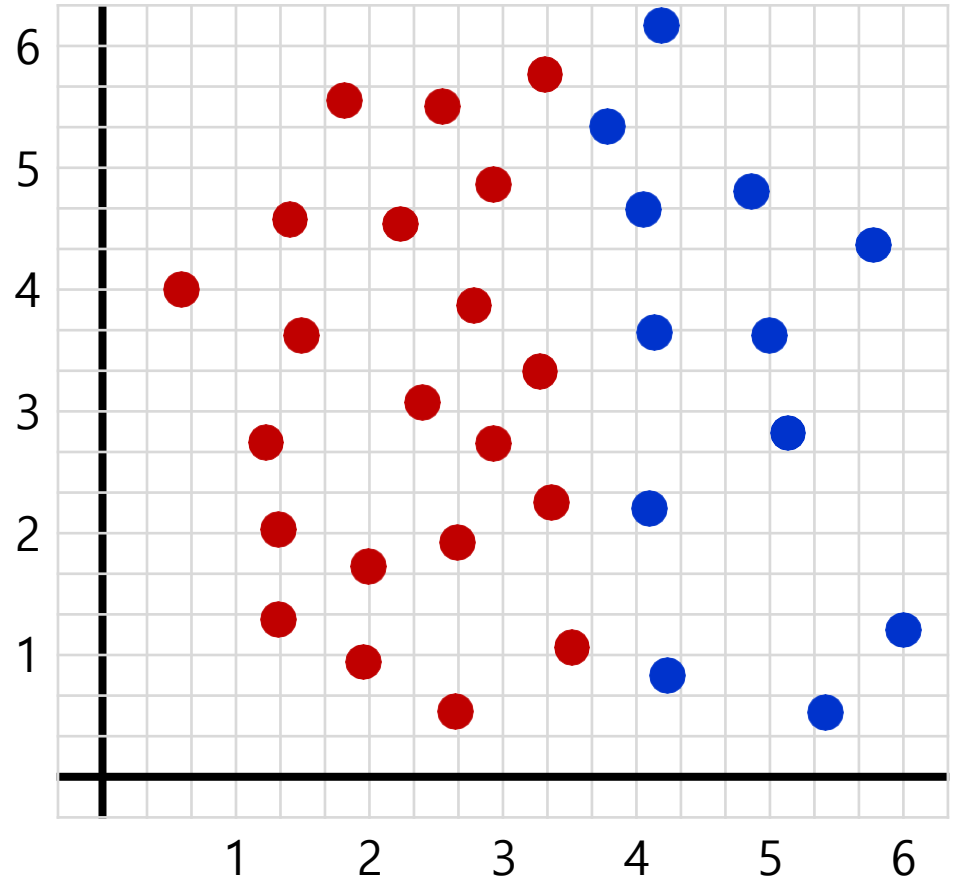
마이닝 알고리즘 (머신러닝 모델) -2

SVM

Data set



Training Set

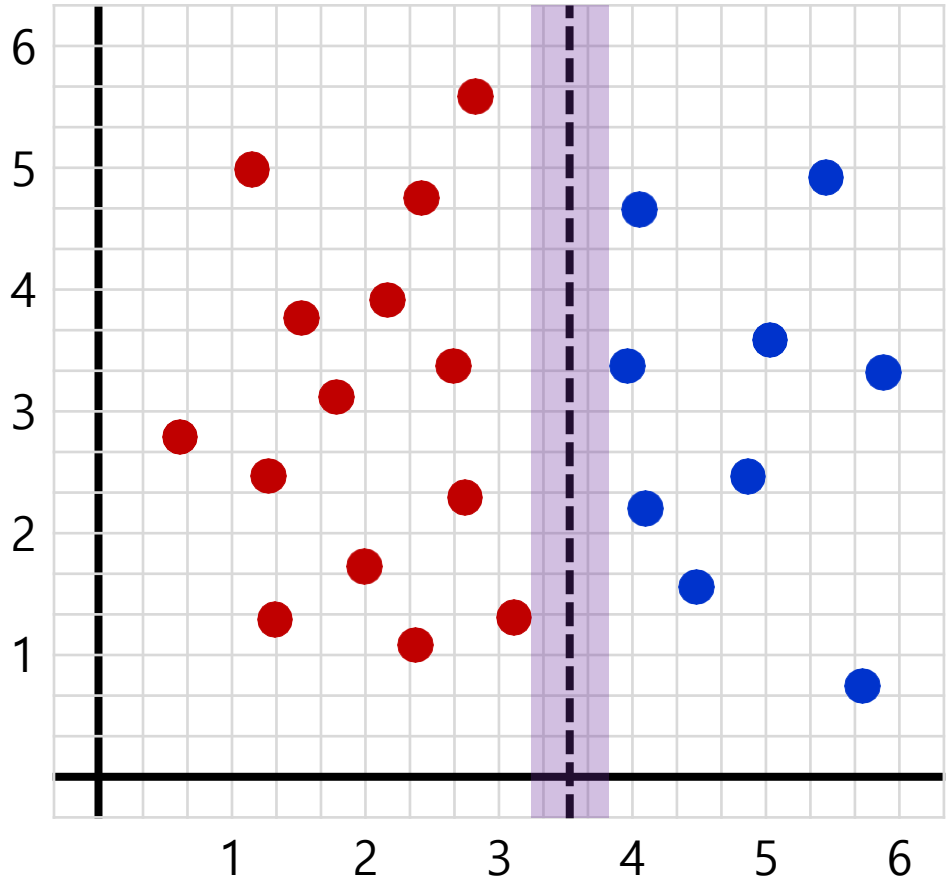


Test set

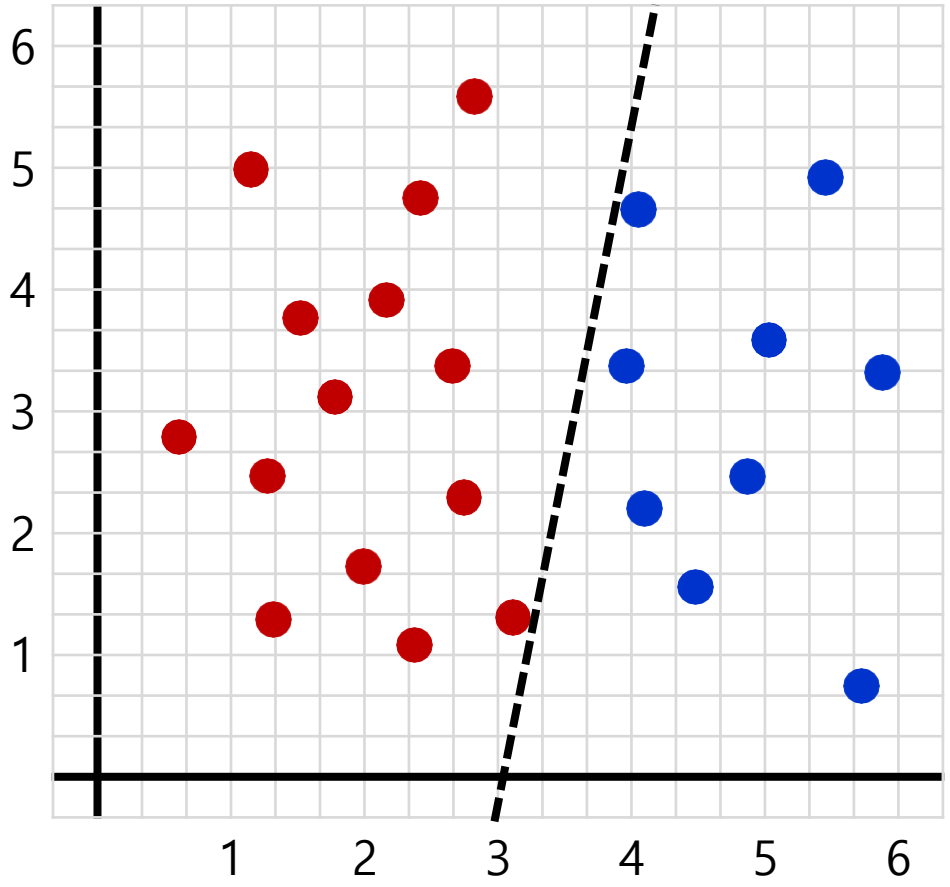
마이닝 알고리즘 (머신러닝 모델) -2

SVM

Training SVM with different Margin length



Training Set, Margin Max

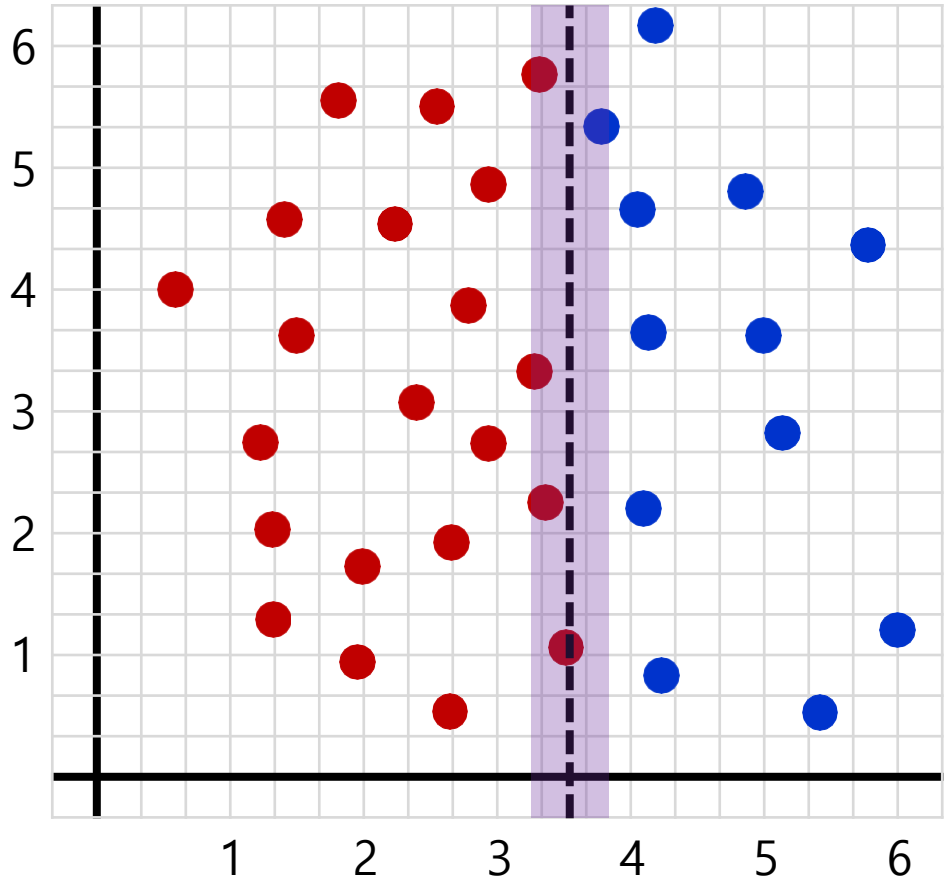


Training Set , Margin Min

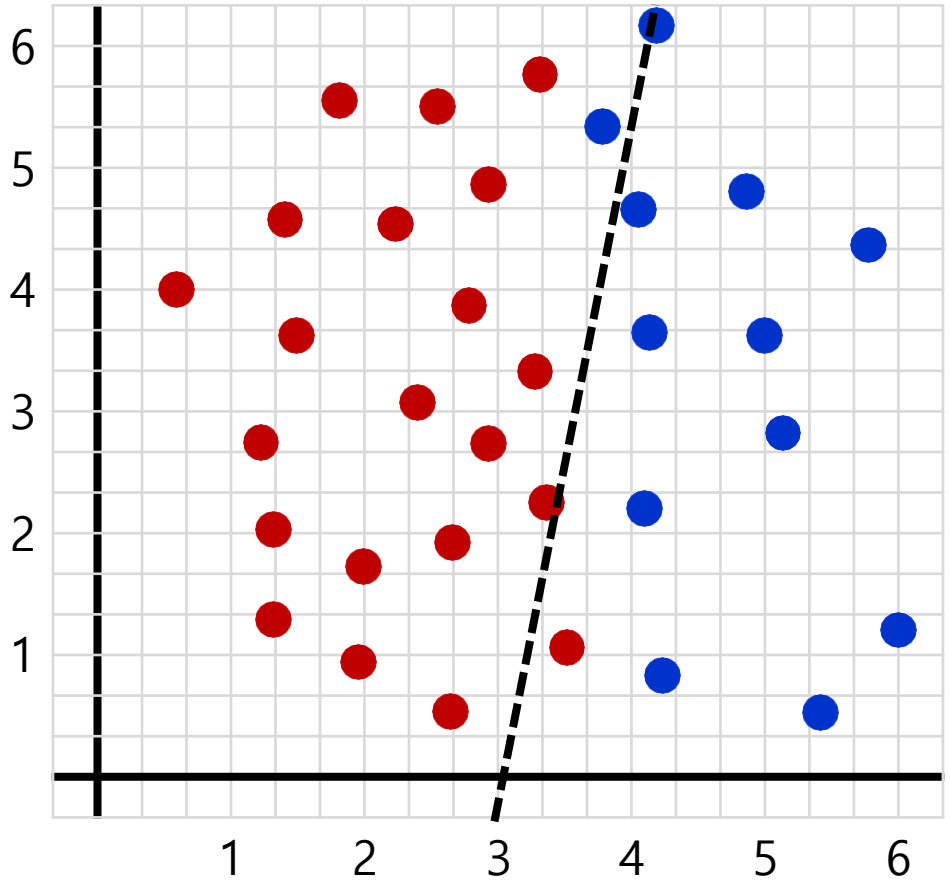
마이닝 알고리즘 (머신러닝 모델) -2

SVM

SVM show different performances



Test Set

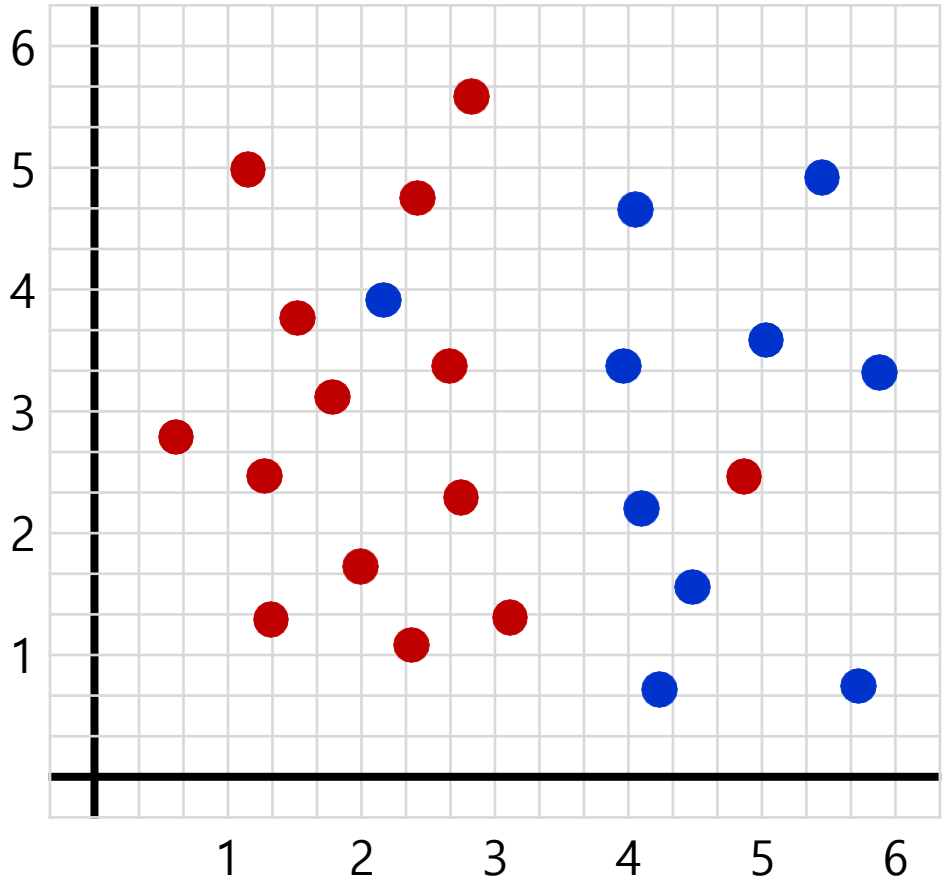


Test set

마이닝 알고리즘 (머신러닝 모델) -2

SVM

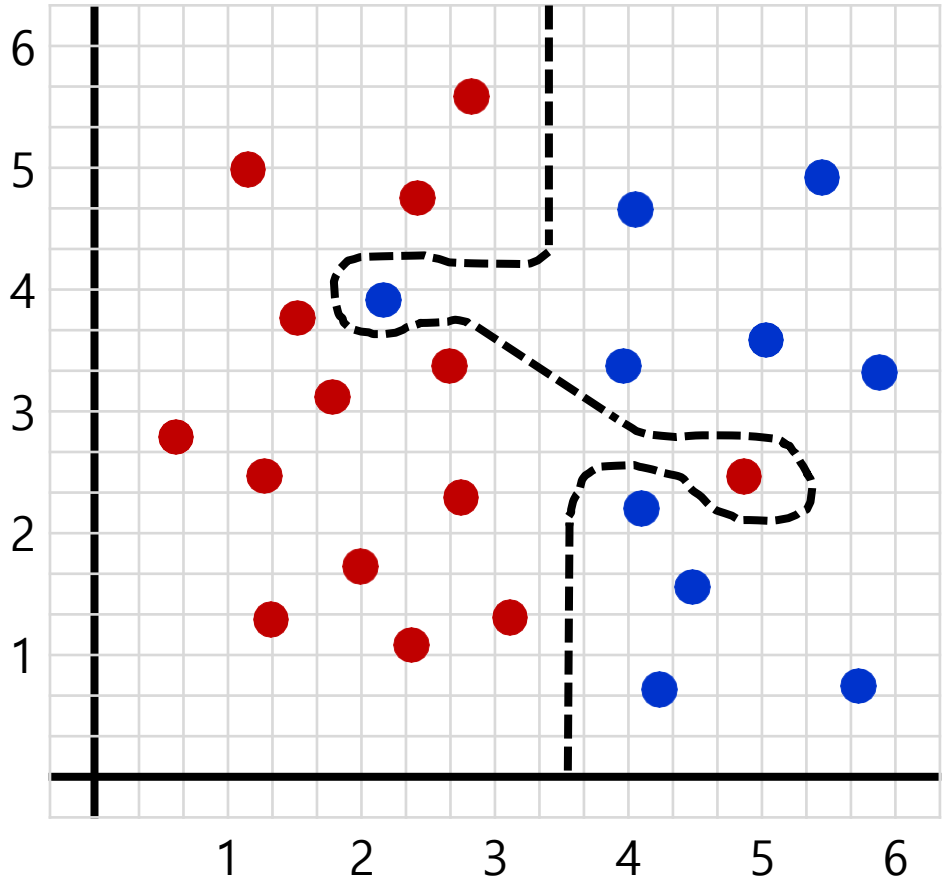
Soft Margin Classification



마이닝 알고리즘 (머신러닝 모델) -2

SVM

Soft Margin Classification

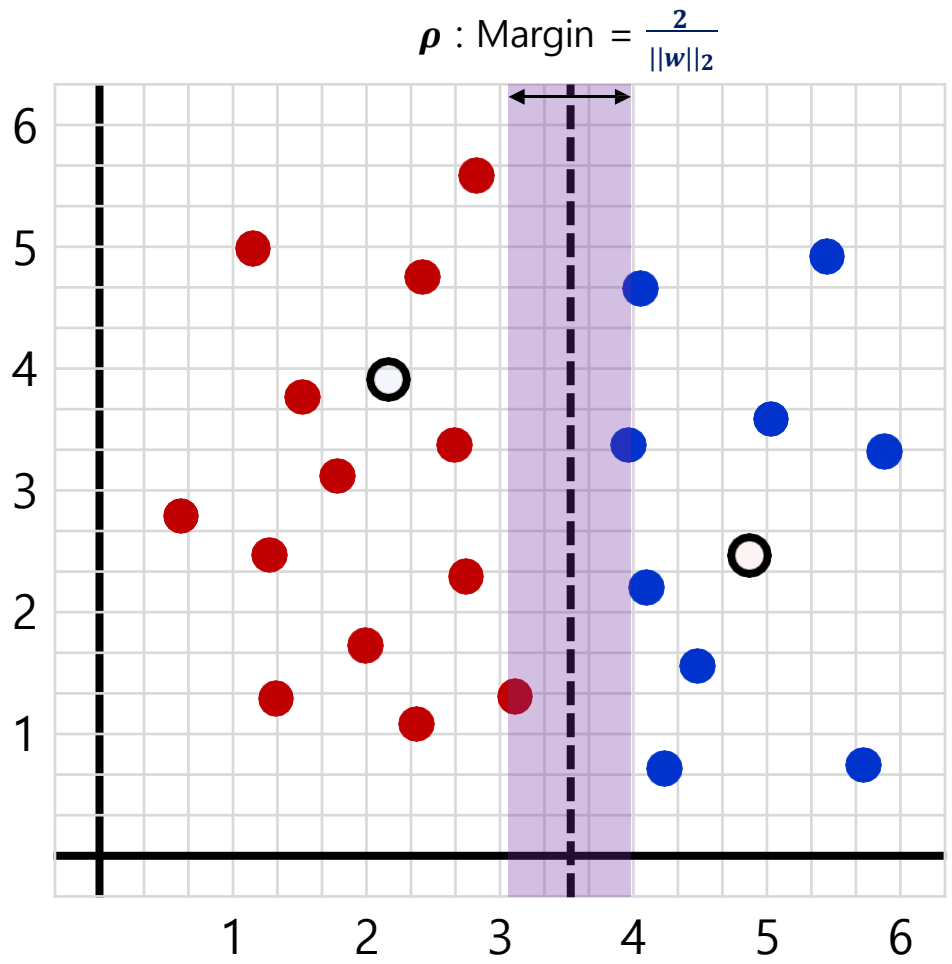


Overfitting at Train data set

마이닝 알고리즘 (머신러닝 모델) -2

SVM

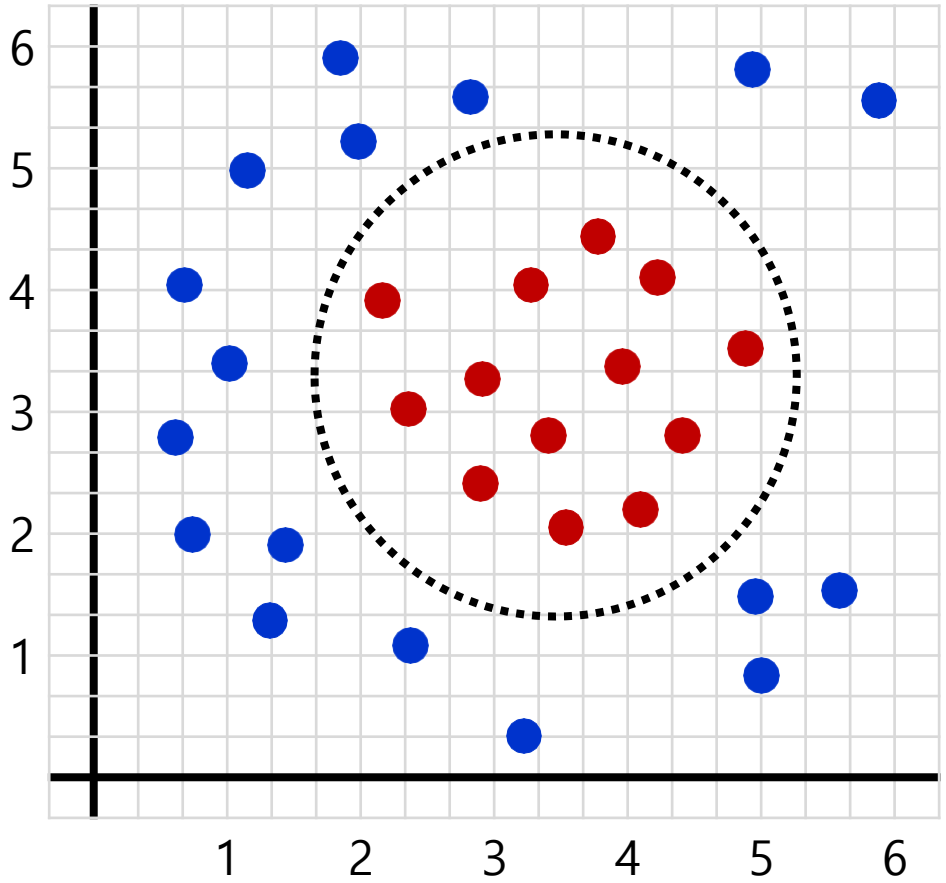
Soft Margin Classification



마이닝 알고리즘 (머신러닝 모델) -2

SVM | **Kernel Function**

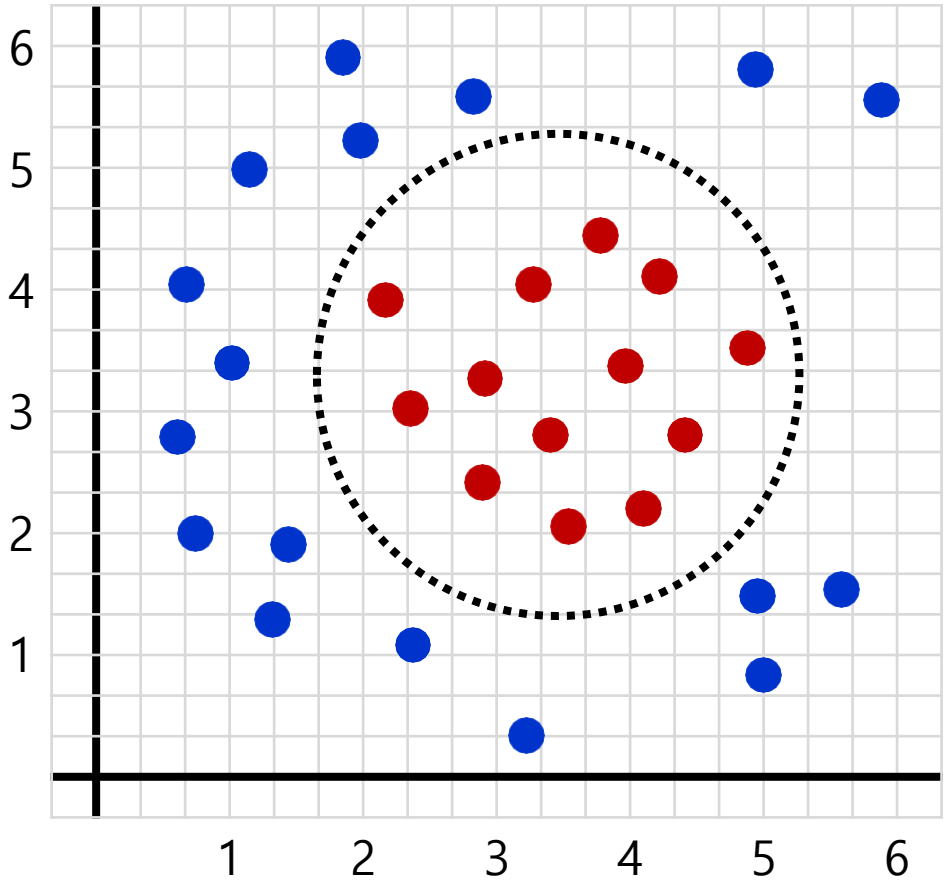
Non-Linear Support Vector Machine



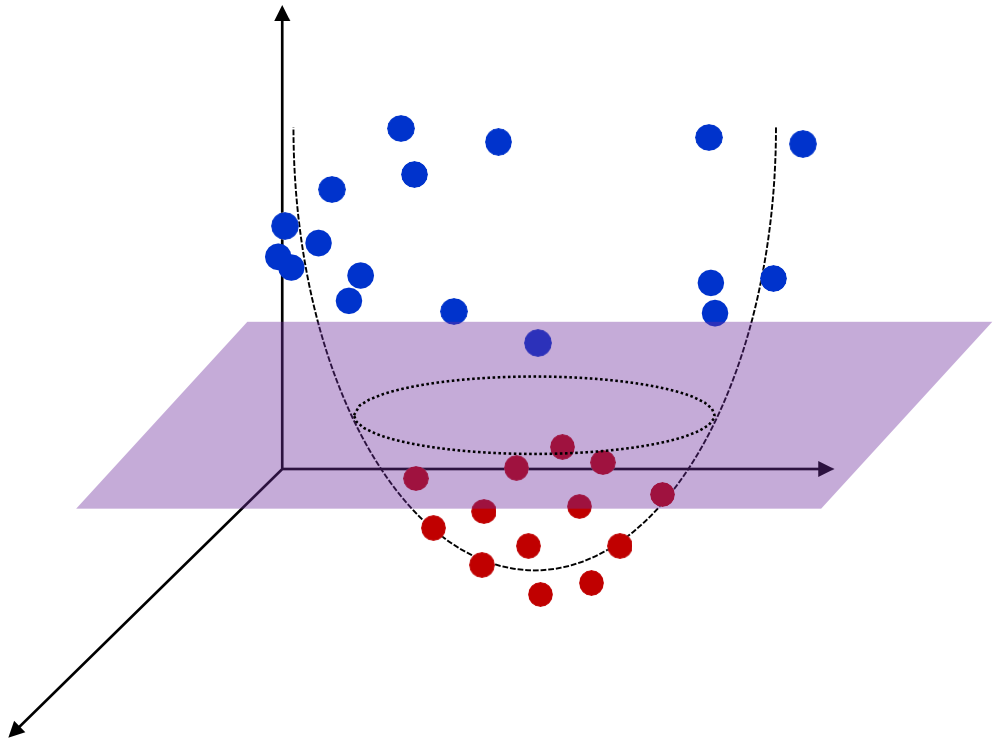
마이닝 알고리즘 (머신러닝 모델) -2

SVM | **Kernel Function**

Non-Linear Support Vector Machine



$\Phi(x)$



마이닝 알고리즘 (머신러닝 모델) -2

SVM **Kernel Trick** 비선형 데이터를 분류할 때 사용하는 핵심 개념

SVM, 최적의 초평면(Optimal Hyperplane)을 찾아 데이터를 분류하는 알고리즘

선형적으로 분리 가능한 데이터에 대해서는, 두 클래스를 나누는 경계를 찾는 것이 비교적 간단

But, 현실에서 많은 데이터는 **비선형적**으로 분포 때문에, 단순한 직선 또는 평면으로는 분류 불 가능

- **비선형 데이터 문제:**

  데이터가 비선형적으로 분포되어 있다면, 선형적인 경계(초평면)를 찾는 것은 불가능

  ex, 원형으로 분포된 두 클래스의 선형 분리

  -> 이의 해결을 위해 데이터를 **고차원 공간**으로 변환 -> 고차원 공간에서 선형적 분리

- **고차원으로의 변환:**

  ex, 2차원 평면에서 원형으로 분포된 두 클래스를 3차원 공간으로 매핑 -> 쉽게 분리

  but, **계산 복잡도** 고려해야.

  데이터의 차원을 계속해서 높이면 계산량이 기하급수적으로 증가 (특히 대규모 데이터셋)

마이닝 알고리즘 (머신러닝 모델) -2

SVM | **Kernel Trick**

- **계산 복잡도 -> 커널 트릭**을 해법으로
  고차원으로 변환된 데이터를 직접적으로 계산하지 않고
  -> 커널 트릭(Kernel Trick) 사용 : 고차원 공간의 점 간의 내적(inner product) 효율적 계산

- **커널 트릭**
  커널 함수 K(x,z) : (**원래) 공간에서** 두 벡터 x와 z의 내적을 계산 함수
  실제로 데이터를 고차원 공간으로 매핑하는 함수 Φ(x)와 Φ(z)간의 내적을 구한 결과와 동일한 값 반환
  !!! **고차원으로 실제로 매핑하지 않고도** 마치 고차원에서 계산한 것처럼 효율적으로 처리

- **수학 표현**

$\Phi(x)$ → $K(x,z) = \Phi(x)^{\top}\Phi(z)$

고차원 변환 매핑 함수

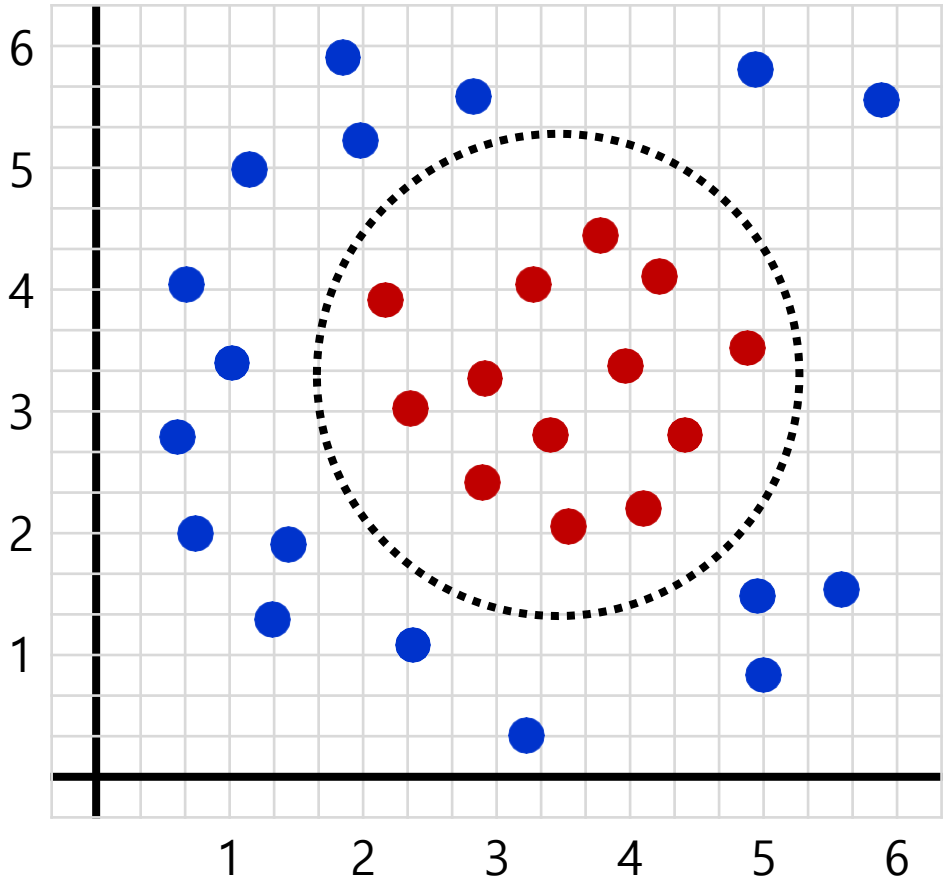
커널 트릭 방식
데이터를 실제로 고차원으로 변환하지 않고, **저차원 공간에서**
두 데이터 간의 내적을 계산하여
마치 고차원에서 계산한 것과 동일한 효과 얻음

마이닝 알고리즘 (머신러닝 모델) -2

SVM | **Kernel Function**

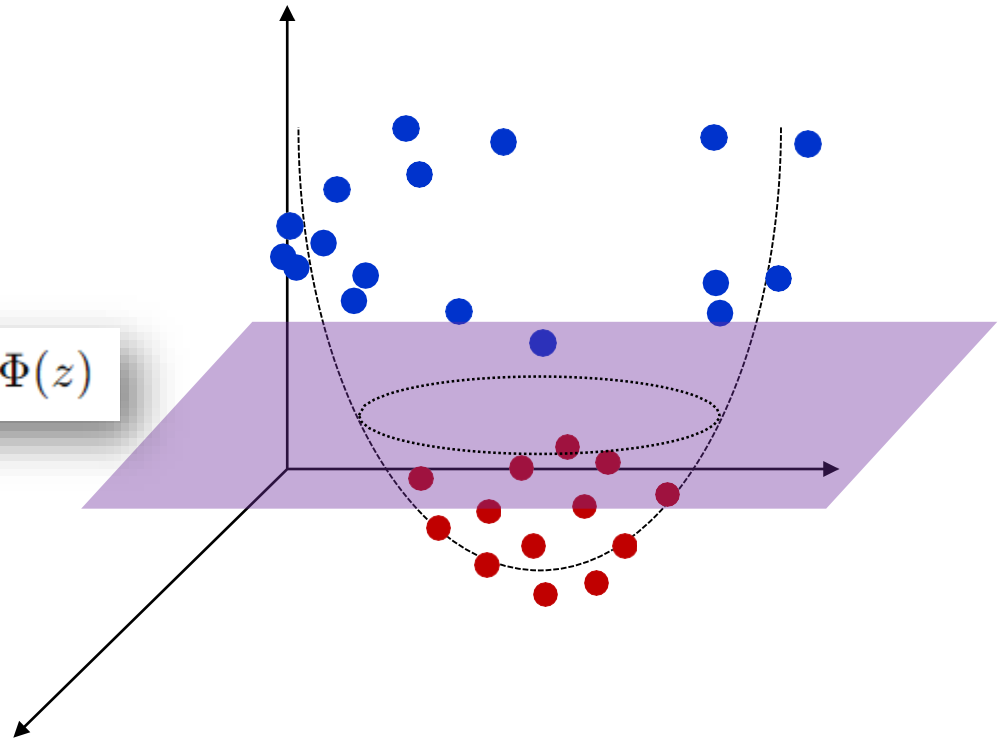
커널 트릭(Kernel Trick) : 저차원 공간(low dimensional space)을 고차원 공간(high dimensional space)으로 매핑해주는 작업

Non-Linear Support Vector Machine



$\Phi(x)$

$$K(x,z) = \Phi(x)^{\top}\Phi(z)$$



마이닝 알고리즘 (머신러닝 모델) -2

SVM | Kernel Trick

- 실습

  2.05.03.svm.KernelTrick.ipynb

THANK YOU