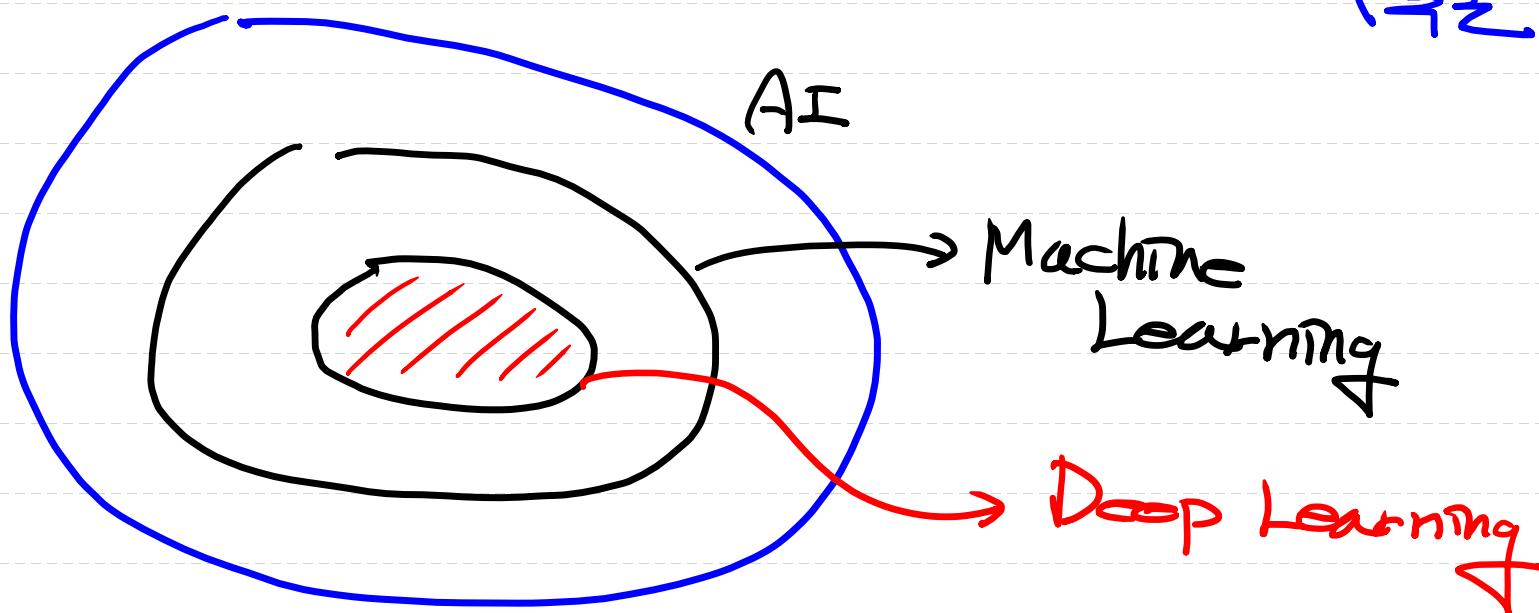
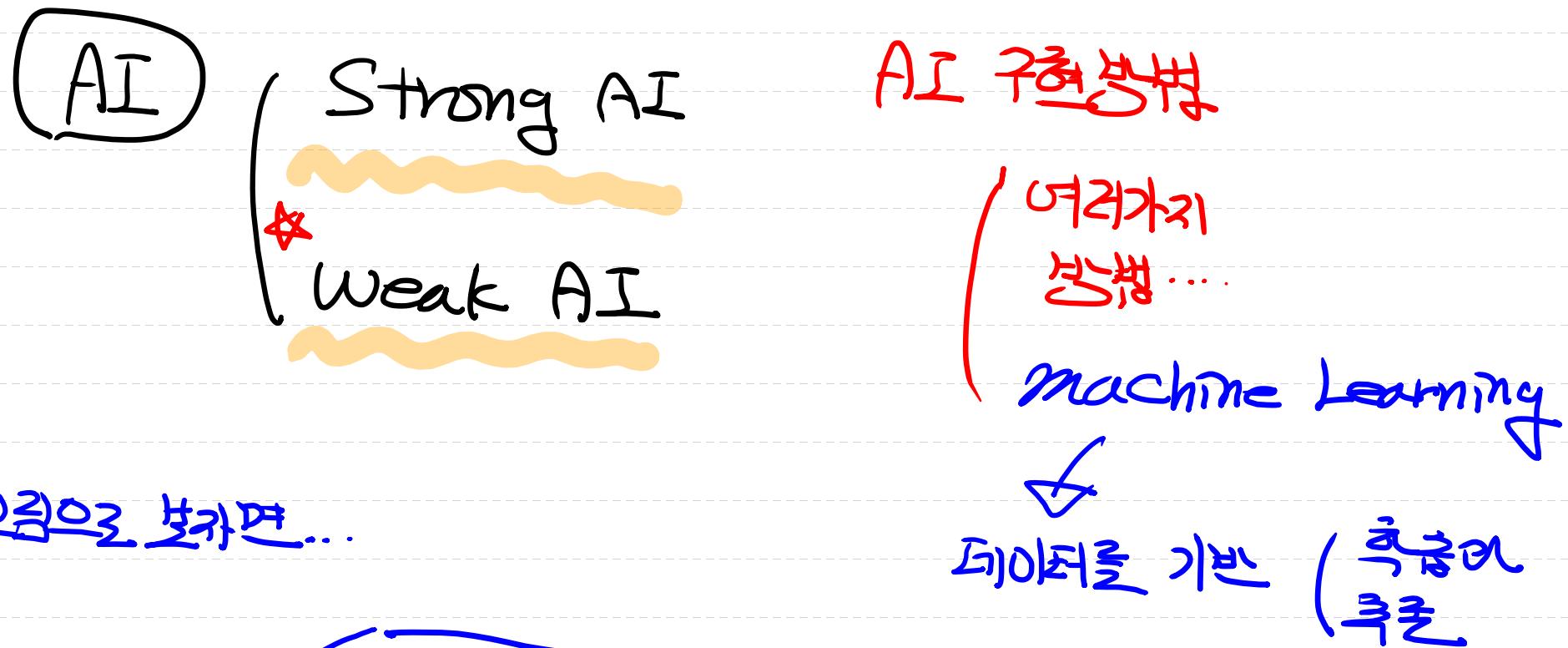


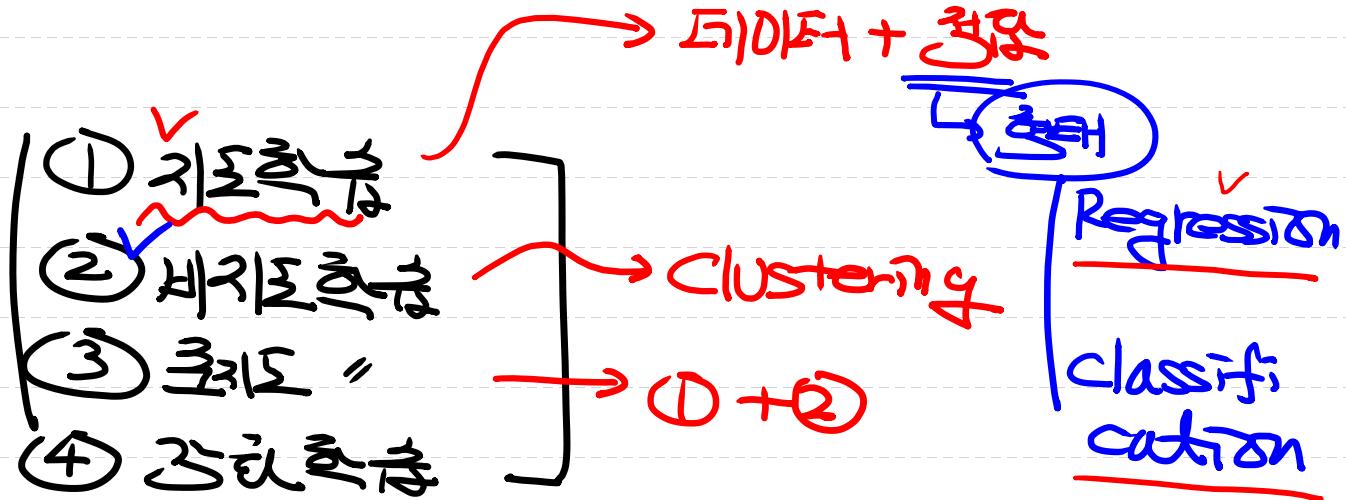
11/29



Machine Learning

데이터의 특성과 규칙

Machine Learning
알고리즘



• Regression

• SVM

• Decision Tree / Random Forest

• Naive Bayes

• KNN

• DBSCAN, k-Means

• ANN → Deep Learning

마분

↳ 주제마분 (종합마분)

- 기본마분공식
- 주제마분

프로그램 코드까지

알아보았나요

Regression

① 풍조를 구하는 기분이기 때문에 풍조의 의미를 가진 Data의 대체

Regression Model이 되는 모듈이 있어요

② 드레스를 가족

③ 가敝 model을 고려한 선형회귀모델 (Classical Linear)

$$\hat{y} = \beta_0 + \sum_{i=1}^n (\beta_i) x_i$$

Regression Models

Regression

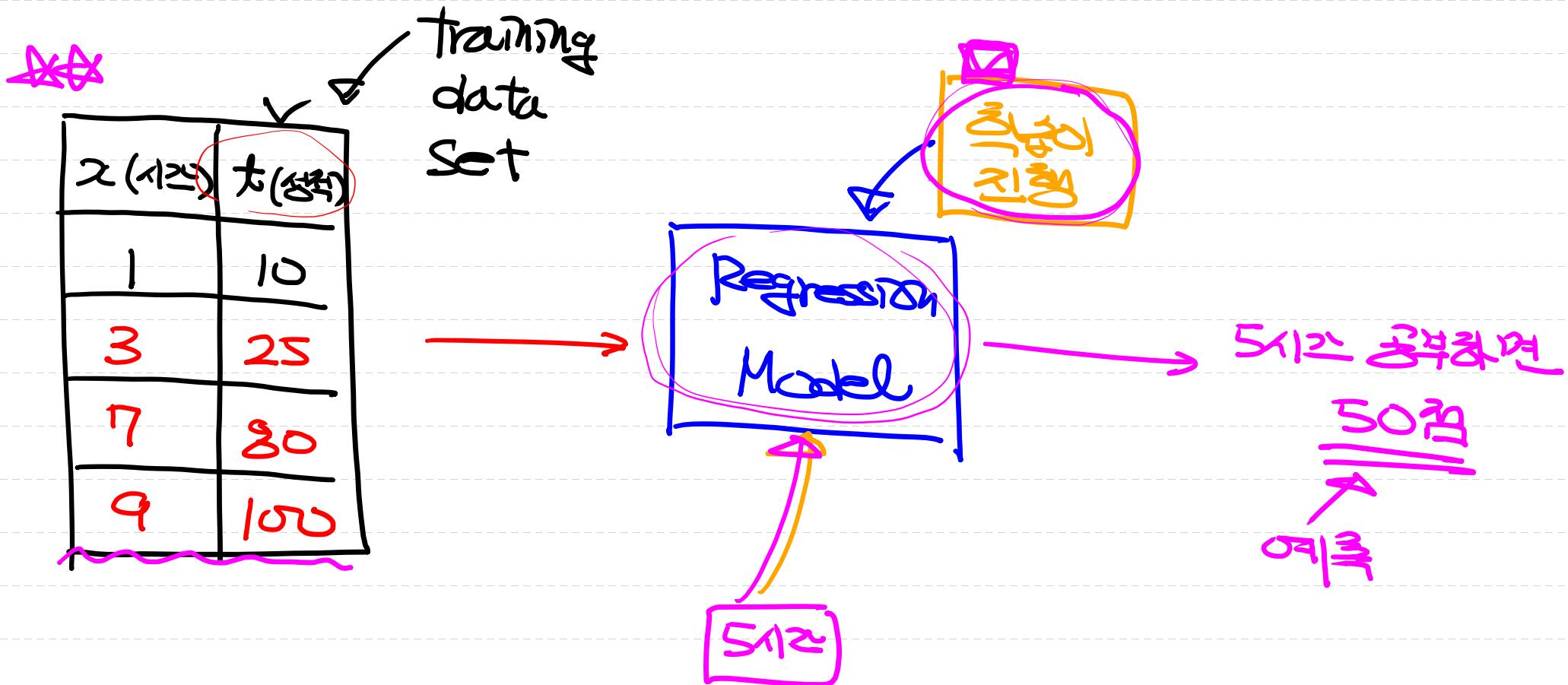
* Linear Regression

target

(Label의 종류가)

숫자

Classification (logistic Regression)



$$\hat{y} = \beta_0 + \sum_{i=1}^n \beta_i x_i$$

독립변수 1개 $\rightarrow \hat{y} = \beta_0 + \beta_1 x_1$

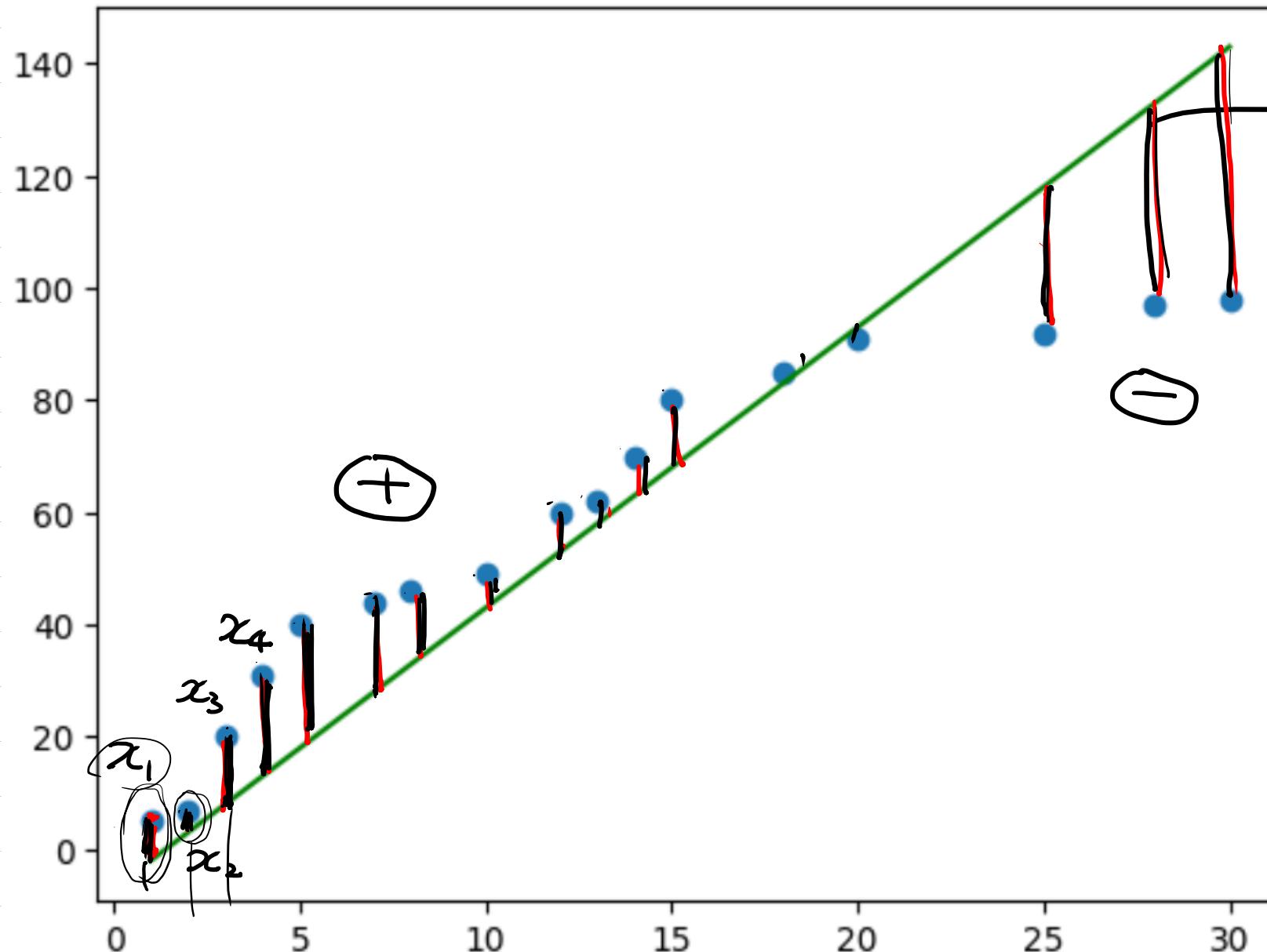
* $\hat{y} = ax + b \rightarrow$ 직선의 회귀

가중치 절편

* Linear
Regression
Model

* $\hat{y} = \underbrace{wx}_\text{weight} + \underbrace{b}_\text{bias}$

① 오류 (Error) \rightarrow $t - \hat{y}$ = $t - (w_x + b)$



(예측값)
이 모델이
우리 데이터를
잘 표현하나요?
어떻게 했을
까요?

오류를 하나의 숫자로 계산하려 해요 → 이 흐름을 loss 흐름
라고 불리요
loss 흐름을 만드는 방법

- ① error를 고해요 → 부호파문에 이어진 loss값으로 사용하기 힘들어요
- ② error의 절대값의 합 → "ok. 좋아요." loss값을 이용해보자
계산할 수도 있으려요
- ③ ~~error의 제곱의 평균~~ → **MSE**
~~제곱 평균 오류~~
MSE (Mean
squared
Error)

Model \rightarrow $y = \boxed{w_x + b}$

Error \rightarrow $t - y = \underbrace{t - (w_x + b)}$

Loss \rightarrow $\frac{(t_1 - y_1)^2 + (t_2 - y_2)^2 + \dots + (t_n - y_n)^2}{n}$

Loss \rightarrow $\frac{[t_1 - (w x_1 + b)]^2 + [t_2 - (w x_2 + b)]^2 + \dots}{n}$

Loss \rightarrow $\frac{1}{n} \sum_{i=1}^n [t_i - (w x_i + b)]^2$

Loss는
모델 예측과 실제
결과 오차값이라고
할 수 있네요

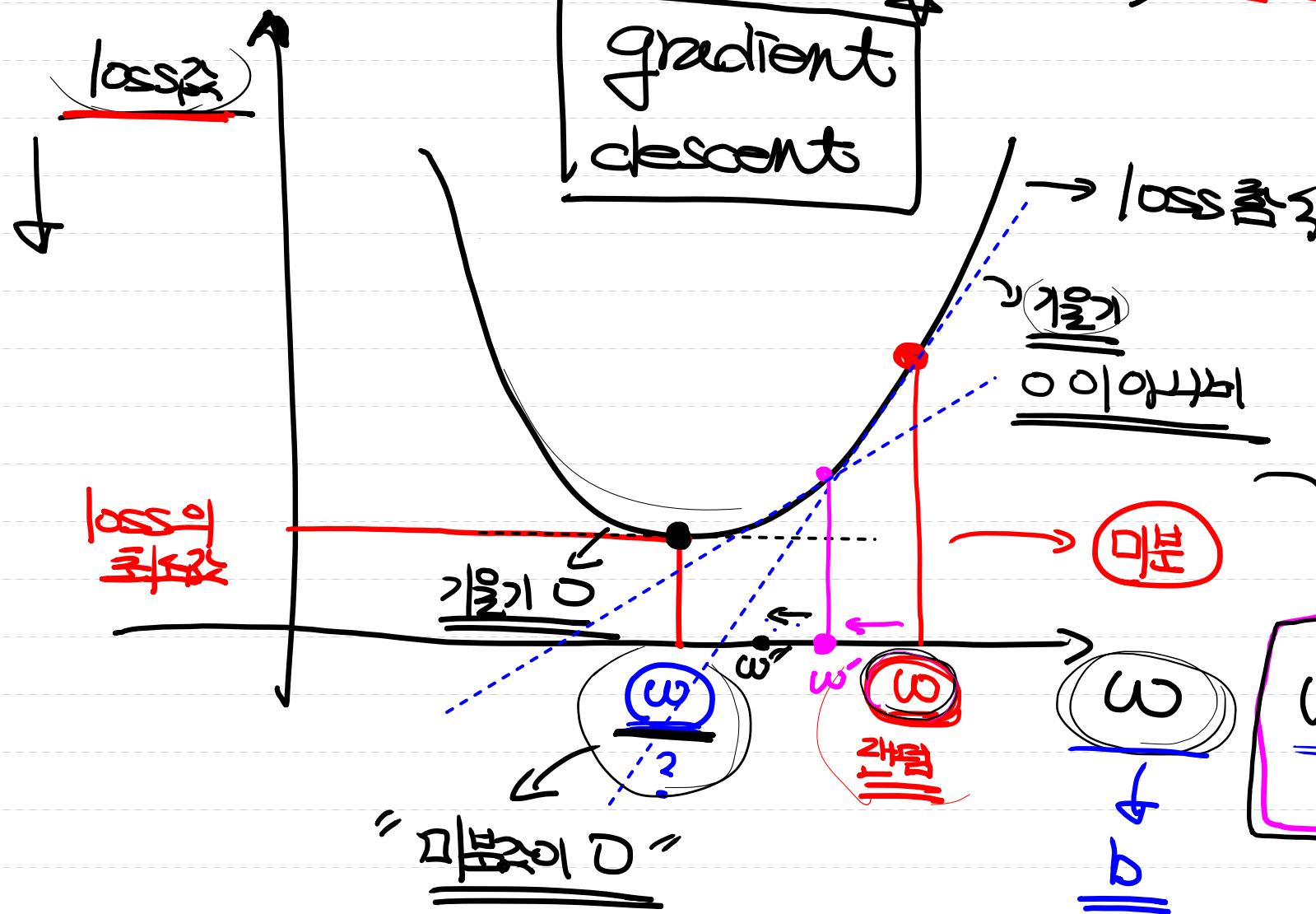
loss function, $E(w, b) = \frac{1}{n} \sum_{i=1}^n [t_i - (w x_i + b)]^2$

Loss(w, b) → 이하를 줄기 위해 b 를 상수로 취급

↳ Loss는 w 의 2차 평균

$y = wx + b$

↳ 포물선 그래프

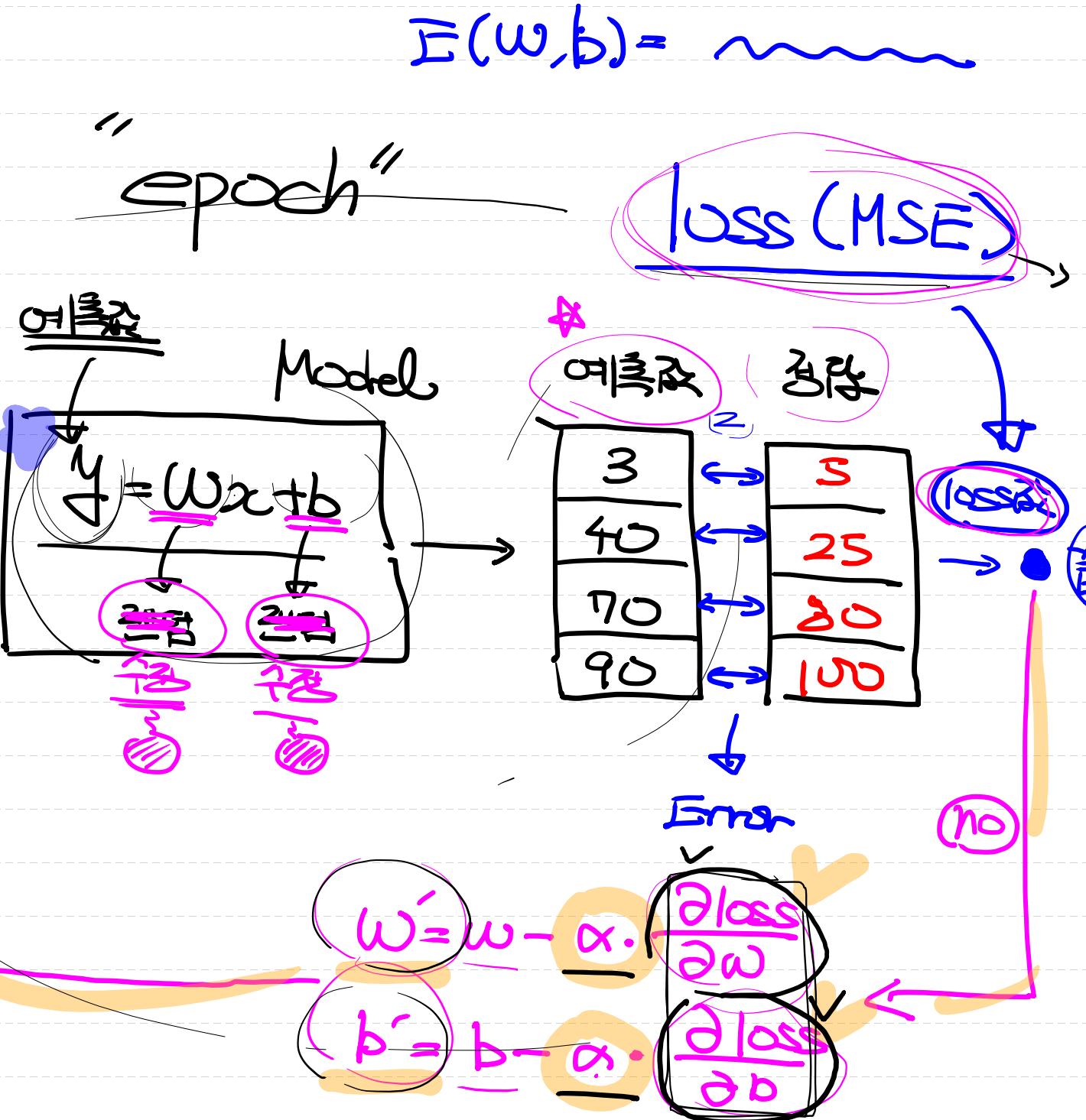
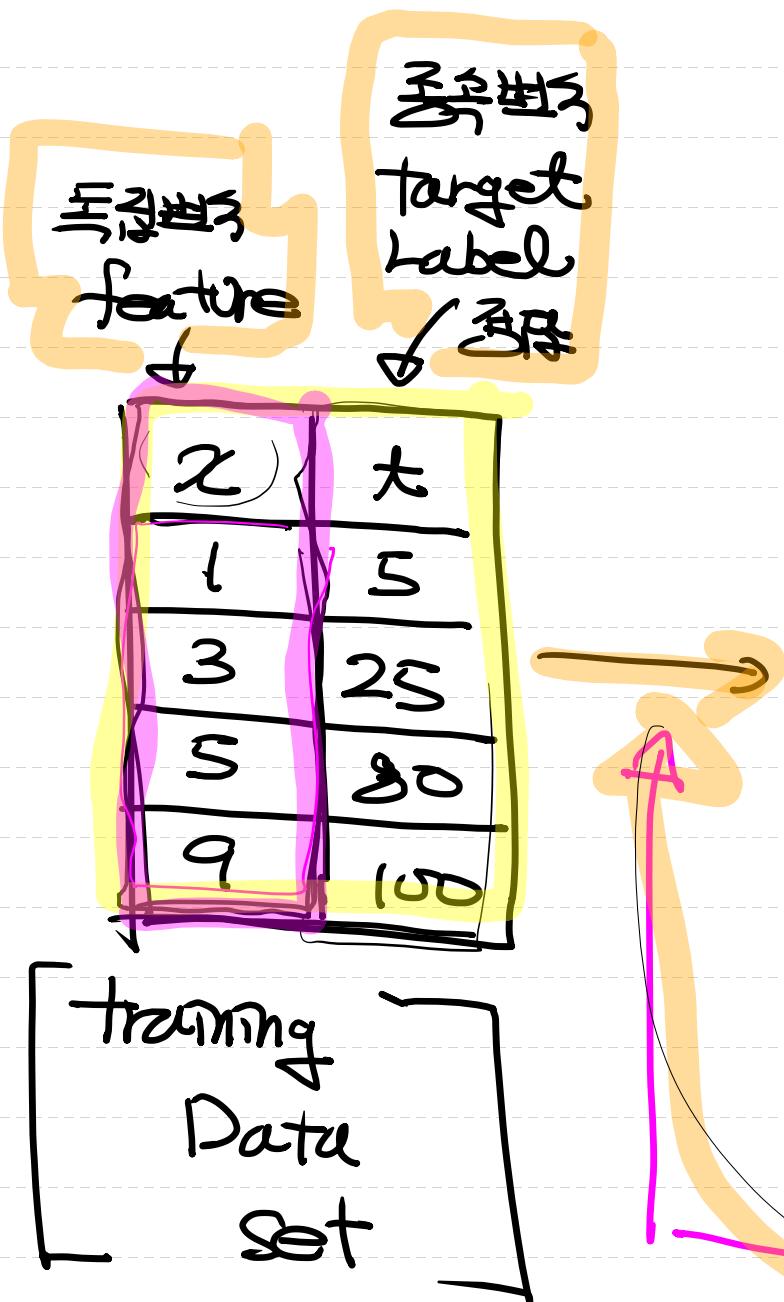


loss는 작으면
직을 줄여 줄까요!

↳ loss가 최소가 되는
w와 b를 찾으면

도대!!

• 그림으로 그려보자.



○ 구현특연이km 솔직보아요!!

* $y = w_2x + b$ → Model

x	t
x_1	10
x_2	40
x_3	60
x_4	90

$x_1(w) + b \rightarrow y_1$
 $x_2(w) + b \rightarrow y_2$
 $x_3(w) + b \rightarrow y_3$
 $x_4(w) + b \rightarrow y_4$

→ training data set

방법론적(0)학습 과정
반복(2)학습 (loop)

$F = X \cdot W + b$

~~\times~~ $w + b = F$

x_1
 x_2
 x_3
 x_4
 (4×1)

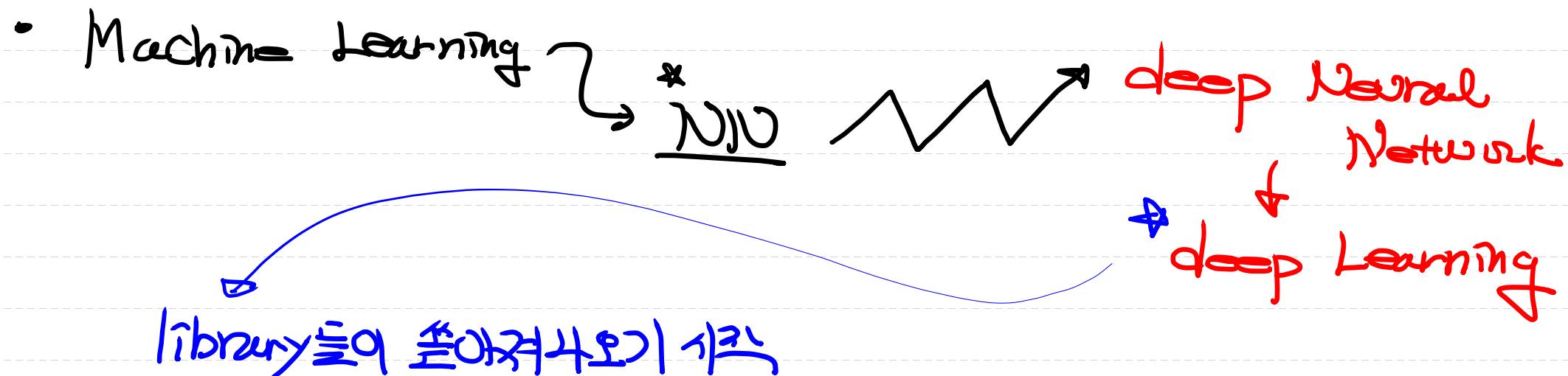
$w + b =$

y_1
 y_2
 y_3
 y_4
 (4×1)

$(1 \times 1) \rightarrow (4 \times 1)$

Python 풀이

- Tensorflow → google 2015년에 발표
 - Tensorflow 1.x
 - ① open source library
 - ② 구조계산용 library
 - ③ data flow Graph를 이용해 Tensor를
다뤄온다
- PyTorch → facebook에서 개발한 deep Learning library



- 2013 → Caffe
- 2014 → Torch3
- 2015 → MXNet, Tensorflow, Keras
- 2017 → CNTK(MS)
- :
- Tensorflow 2.x 버전 ↗ Modular
- ↳ keras를 기본으로 구현 (훨씬 더 쉽죠)
- “인공지능 훈련”
- ✓ ↗ 사용하기 훨씬 쉬워요
-

Tensorflow keras 구현

$$Y = X \cdot W + b$$

