Introduction to Computer Science & Engineering

Lecture 12: A Few Basics of Learning Systems

Jeonghun Park

Introduction

if 문이나 while 문 없이 어떠한 데이터를 얻을 수 있는것.

- What is machine learning?
  - Field of study that gives computers the ability to learn without being explicitly programmed
  - A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E

E, T, P 3가지로 컴퓨터 프로그램 진행

jeonghun.park@knu.ac.kr

Quick Example

- Suppose your email program watches which emails you do or do not mark as spam, based on that learns how to better filter spam. What is the task T in this example? And what is the experience E and the performance P? 컴퓨터가 이해함(숫자로 이루어진 value)
  - Classifying emails as spam or not spam Task T
  - Watching you label emails as spam or not Experience E
  - The number of emails correctly classified as spam/not spam Performance P

Machine Learning Algorithms

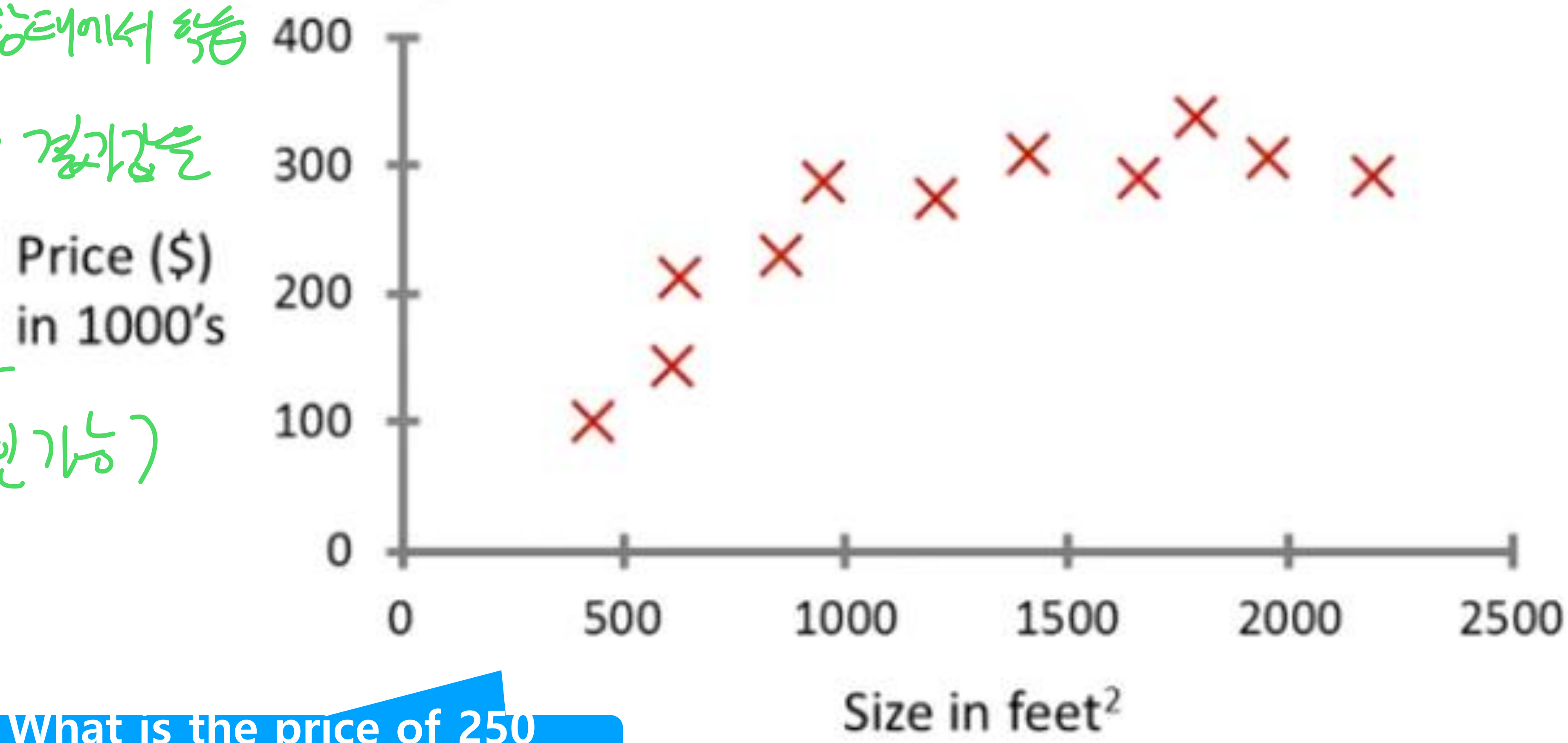
* Information system : 실제 데이터 안쓰고 확률 이용
machine Learning : 실제 Training Data 사용

- Supervised learning
  - ① Regression (연속)
  - ② Classification (이산)

지도식 학습방법

이미 답을 알고있는 상태에서 학습
(Dataset이 이미 결과값을 포함하므로 얼마정도의 오차가 있는지 확인가능)

Housing price prediction.



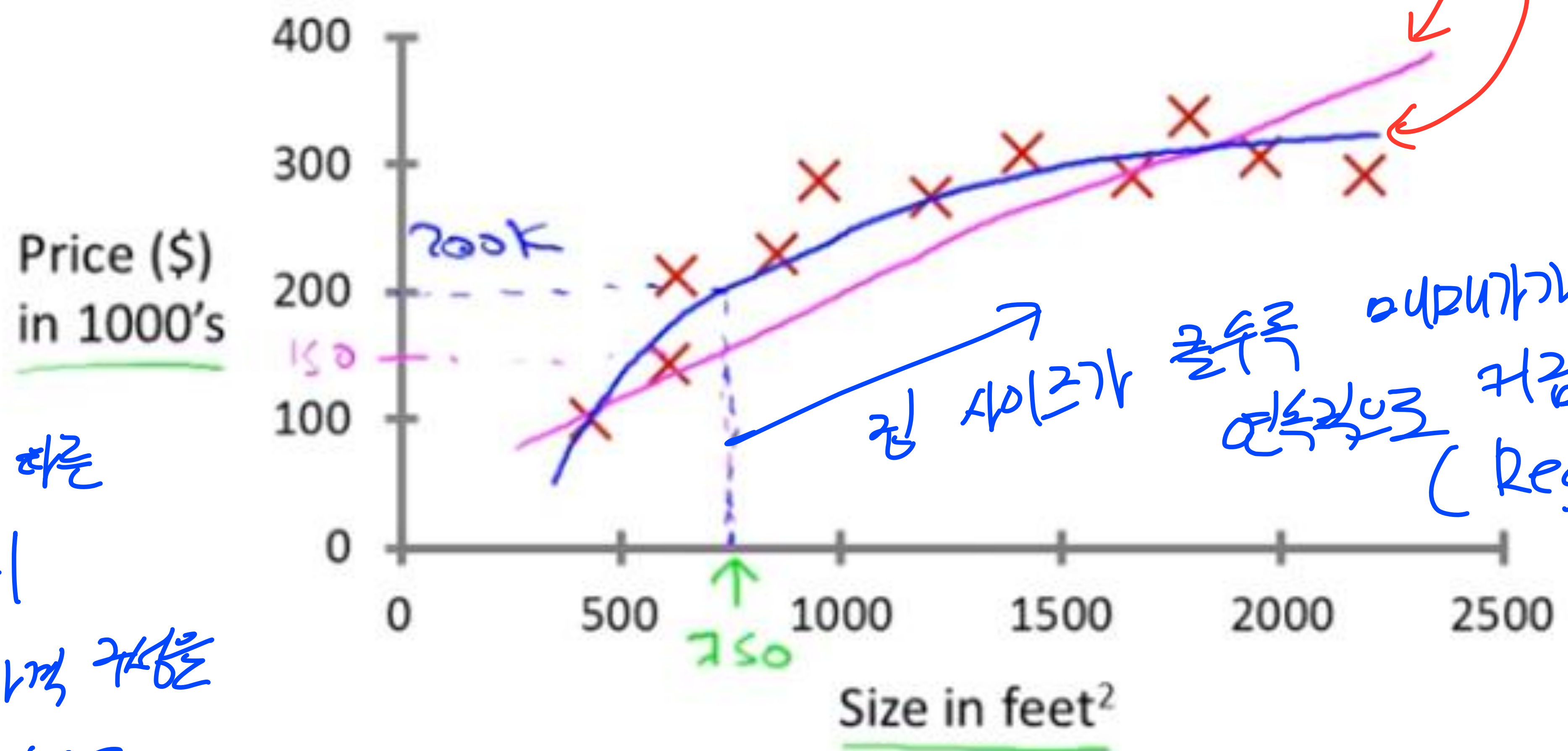
What is the price of 250 feet^2?

KNU KYUNGPOOK NATIONAL UNIVERSITY

Supervised Learning

machine Learning 으로 예상한 value 에 대한 함수를

Housing price prediction.



ex) 집 사이즈에 따른 집의 가격분석 시
평수에 따른 가격 거래를
미리 알고 있으므로
실제 집이 거래된 가격도 정확하게 알수있음.
이를 통해 평수당 가격과 실제거래가격간 cost (오차) 비교가능.

Regression

연속적인 output

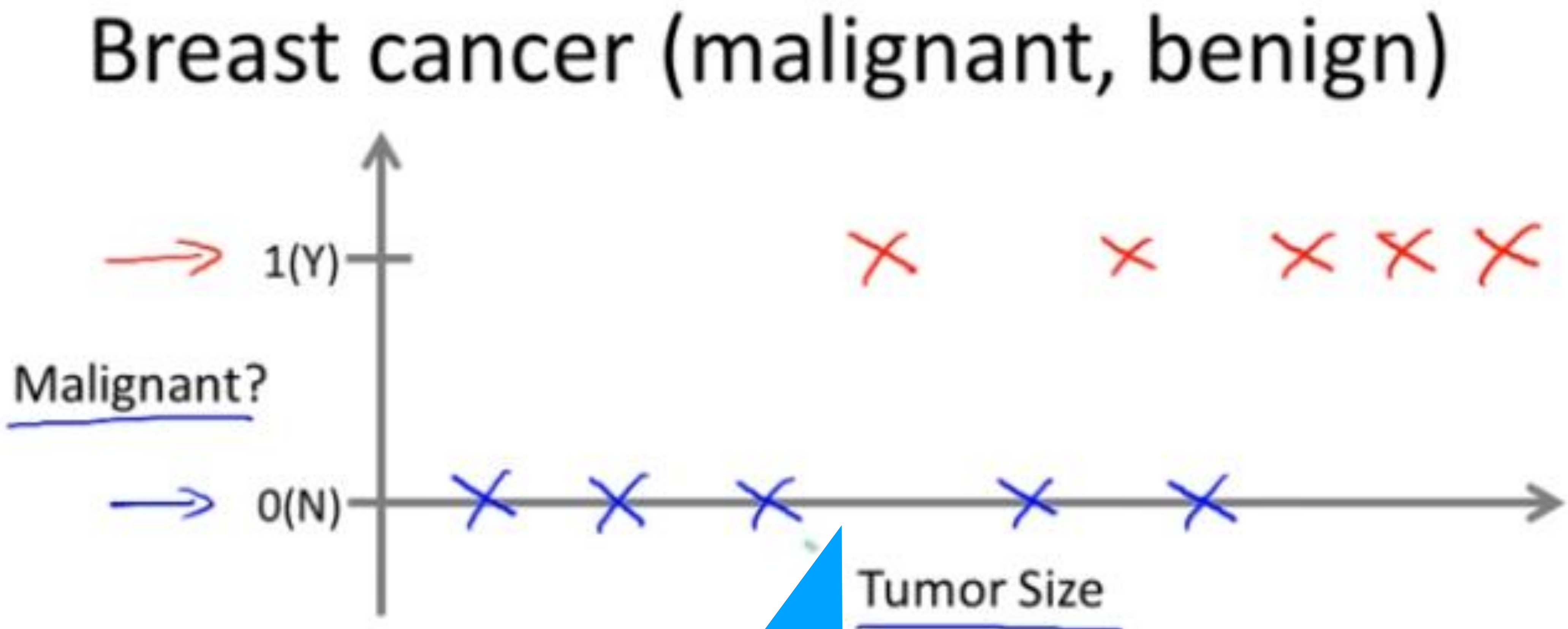
- Right answers given
  - Called training set

실측된 데이터로 만든 함수

- The example is included in “regression”
  - Predict continuous valued output

Supervised Learning

집의 구조에 따라 아파트인지 주택인지 구분되는 건 Classification



Would it be malignant if the tumor size is this much?

암종 사이즈를 기준으로
악성인지 아닌지 구분.

jeonghun.park@knu.ac.kr

Classification

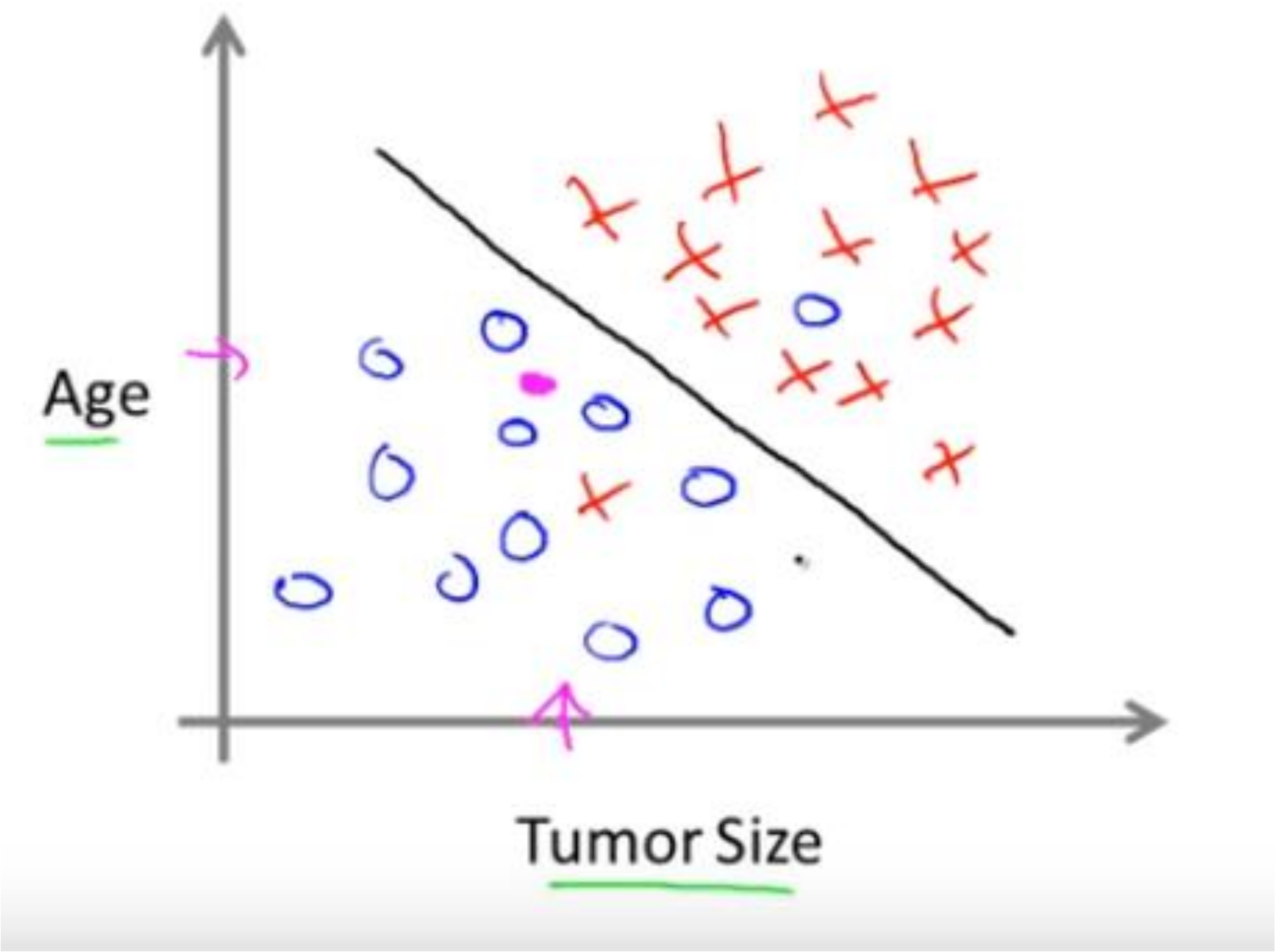
이산적인 output

- Discrete valued output
- Could be multiple
  - For example, cancer type 1, 2, 3, 4,…

jeonghun.park@knu.ac.kr

Classification

- There can be multiple types of data



Unsupervised Learning

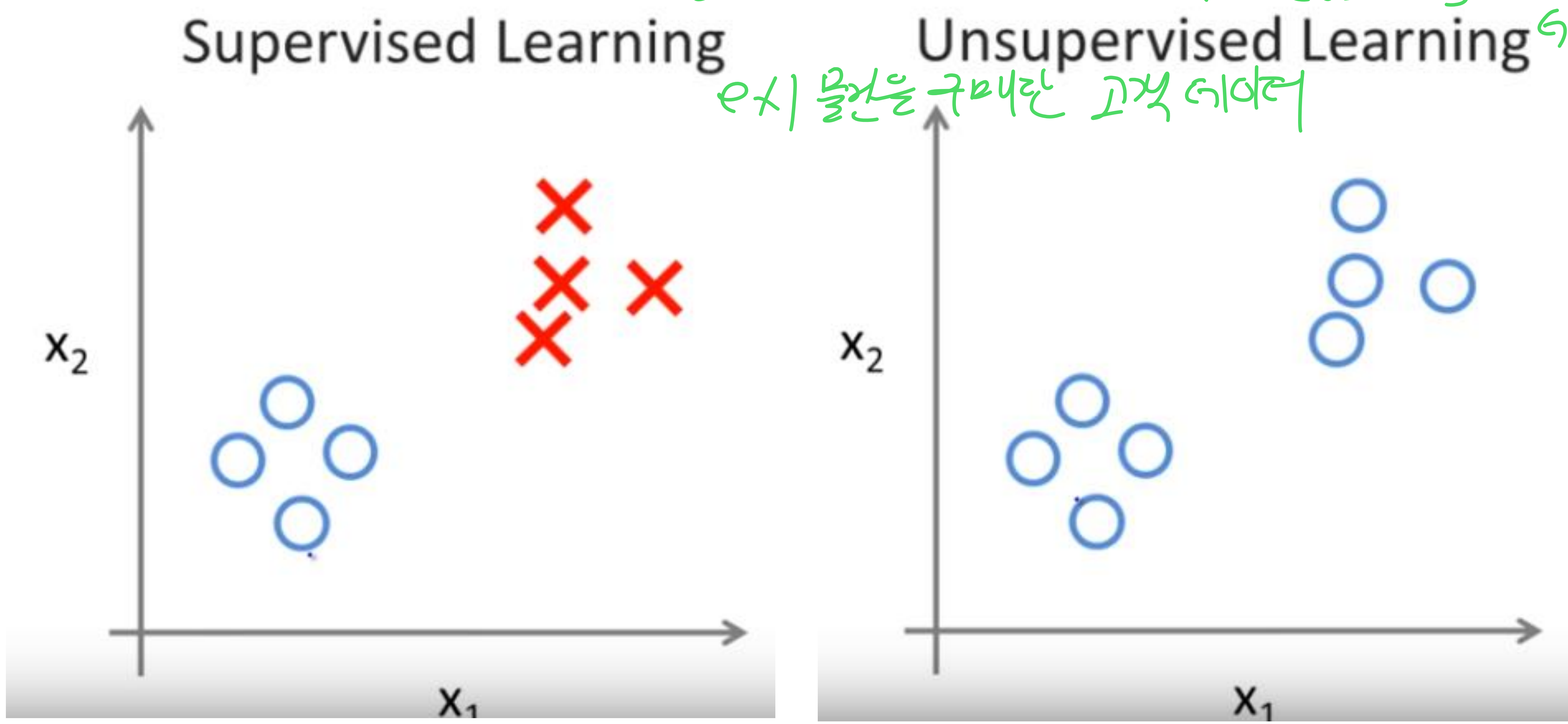
자율학습 => 우리가 답을 알고있지 않은 상태에서 학습하는경우
(Data set 이 어떤 결과가 도출될지 모를때사용)

Lable 이 있어서 구분됨

Lable 구분이 안됨

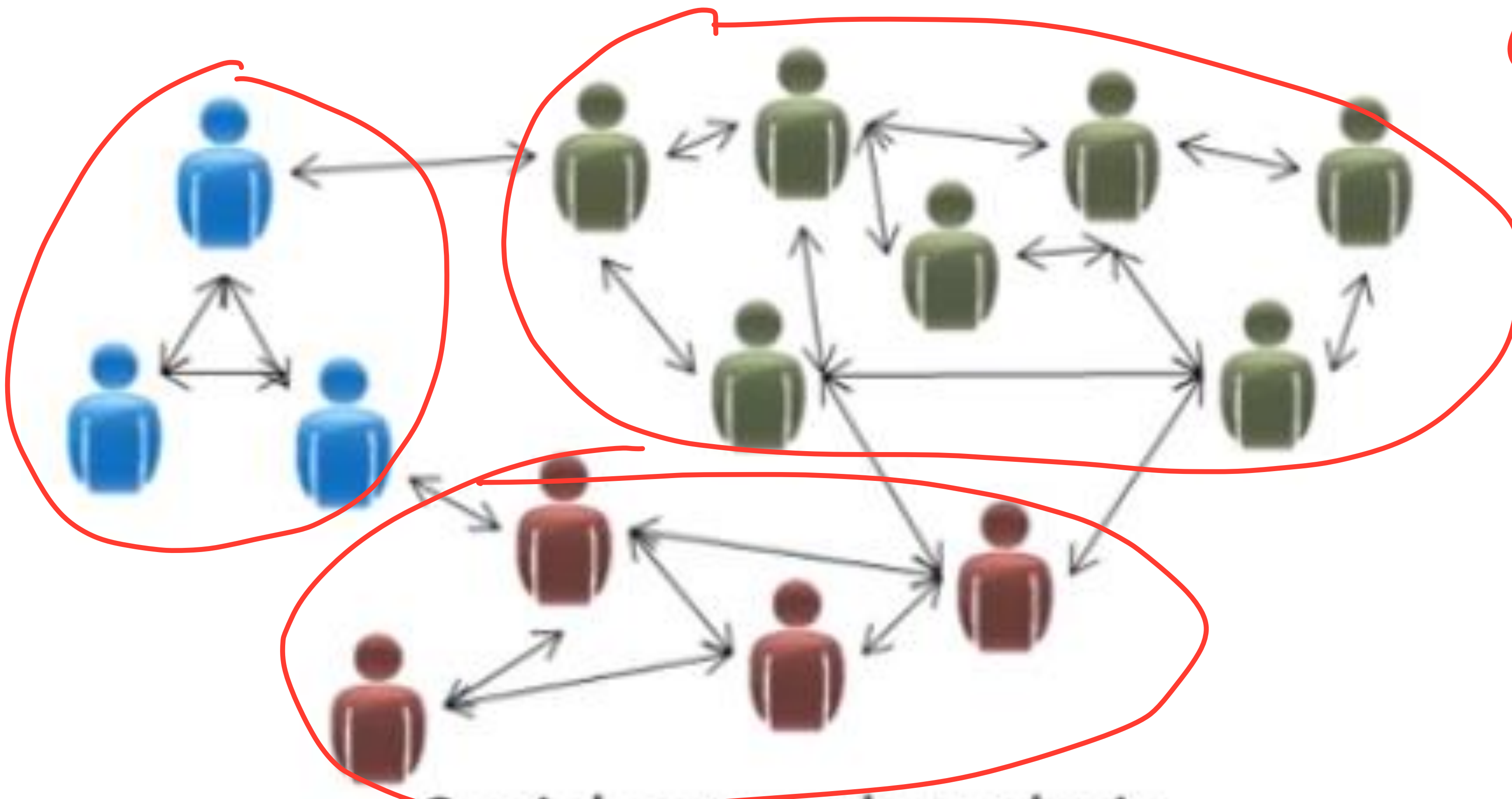
해비안 비슷한 주제로 분류하는것 (clustering 이나 Grouping)

ex) 물건을 구매한 고객 데이터



Clustering

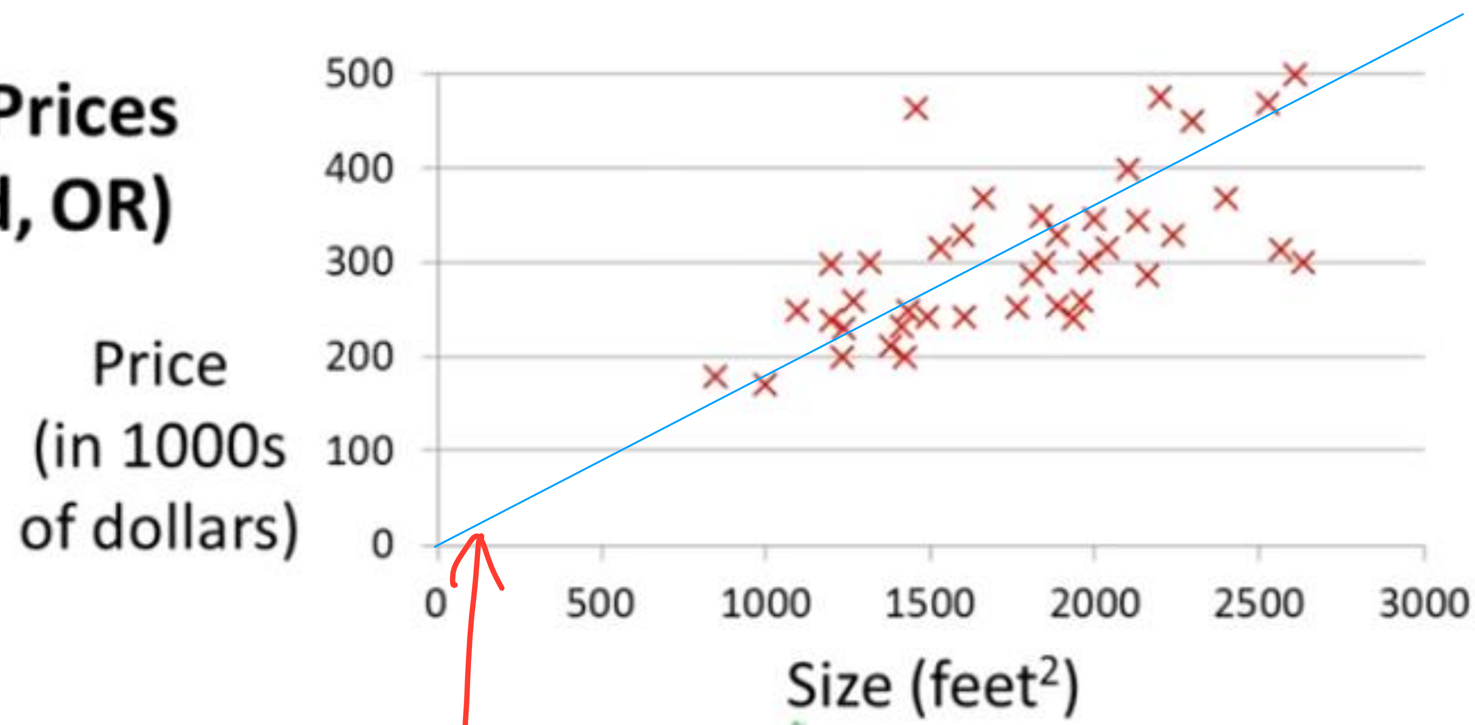
Group Name은 모르지만
Group 별로 구분관리
(Facebook)



Social network analysis

Linear Regression

Housing Prices (Portland, OR)

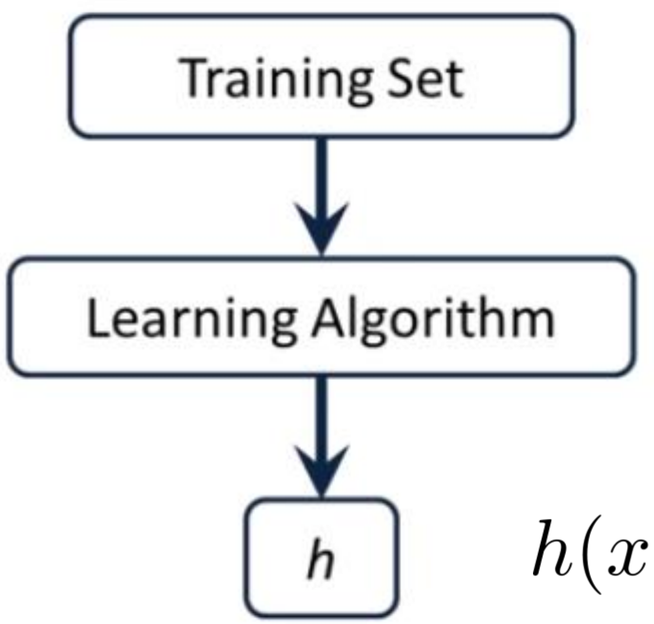


직선이 사이즈에 따른 집값

따라서 이 직선을 찾는것이 중요한데

우리가 직접 찾는게 아니라

알고리즘(머신)이 학습해서 찾아줄것임.



$$h(x) = ax + b$$

a, and b mean the parameters of our function

Linear Regression

* 실제 정확한 값은 Supervised 방식이라 알고있고, 그에 따른 h(x) = Ax+b 를 얻었으므로 실제정확한 값과 이 함수의 결과값을 비교하는게 Cost (오차)

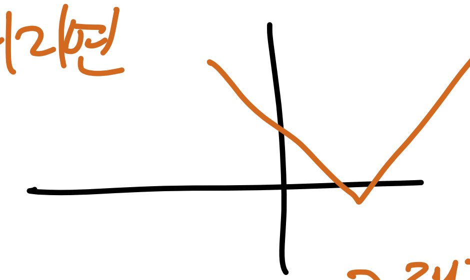
- So, how to determine the parameters? 수학적으로 minimize 해서 찾을수있으므로

- Principle

  Cost = (예측값 - 실제값)^2 마이너스 방지

  - Choose the parameters so that h(x) is close to the training data $\mathbf{t} = [t_1, t_2, ..., t_N]$ for $\mathbf{x} = [x_1, x_2, ..., x_N]$

* 제곱이 없으면 $h(x_i) < t_i$ 시 관리가 안되며 2제곱 다고 abs 취해버리면 이런 그래프나와서 관리가 안됨



- Find: $h(x_i) = ax_i + b$

  - $\min \sum_{i=1}^{N} (\boxed{h(x_i)} - t_i)^2$ => 최소값 구함.

  실제데이터

- We define the cost function $J(a, b) = \sum_{i=1}^{N} (h(x_i) - t_i)^2$

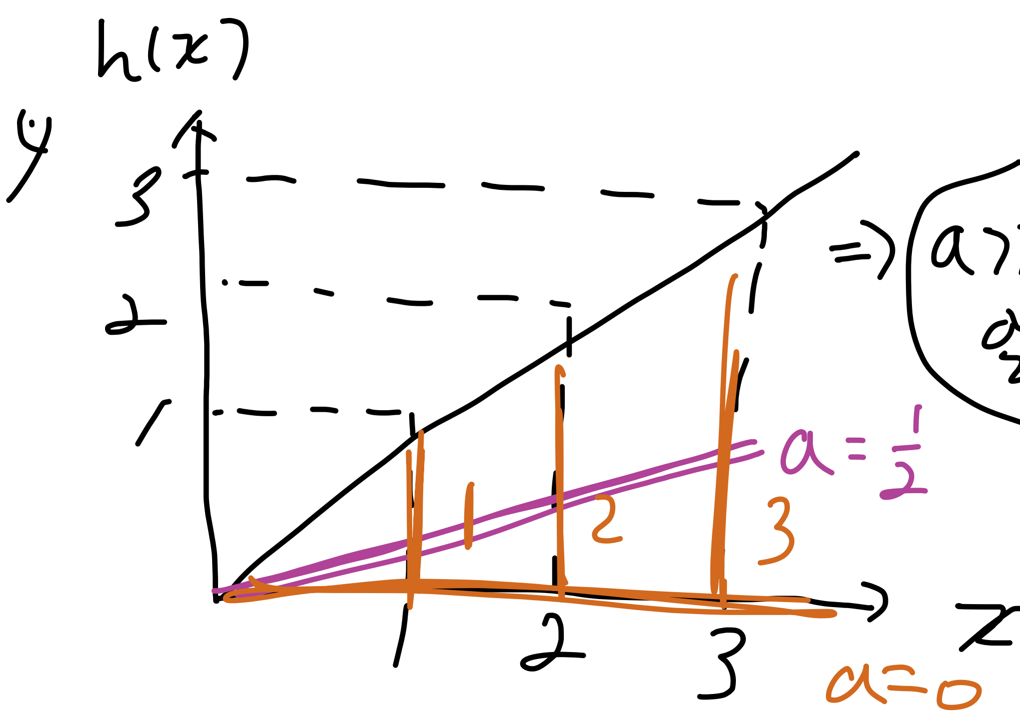
  Cost 함수는 J로 표시

  $\frac{1}{N}$ 곱해주면 평균

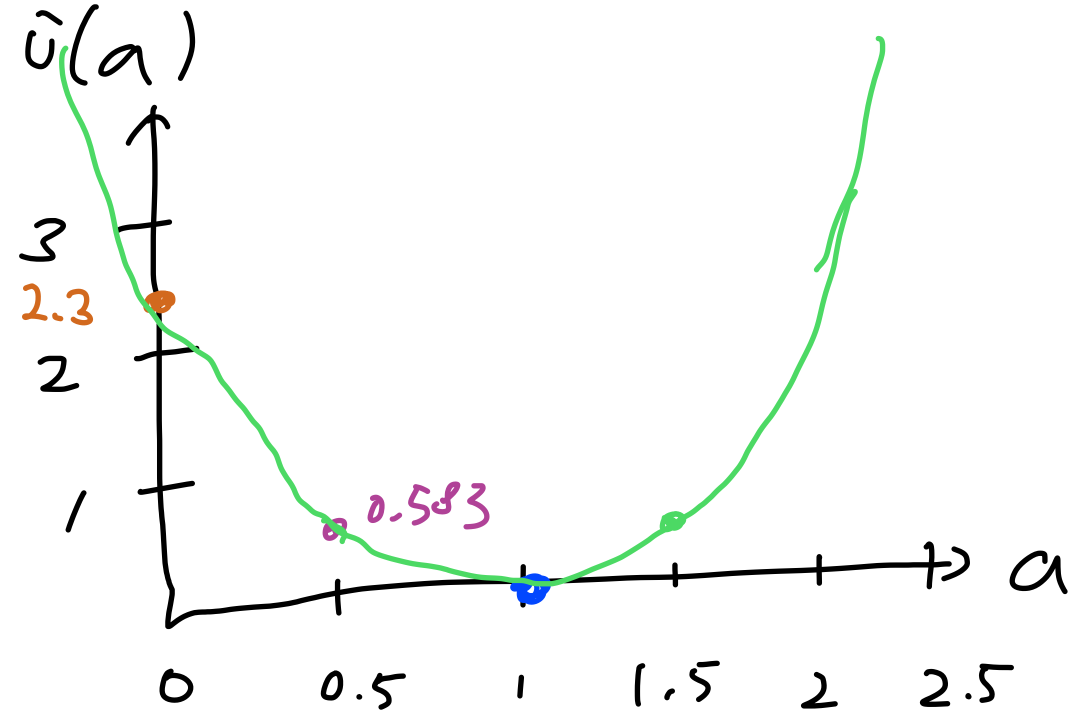
  $2N$ 미분을 하게되면 제곱이 미분되어서 2 곱해지므로 나눠줌

jeonghun.park@knu.ac.kr

$h(x) = ax$ 라 가정하면



=> a가 1임을 알수있다



실제결과인 t_i와 $h(x)$가 동일하면 cost가 0일것.

$$J(a) = \sum_{i=1}^{N} (h(x) - t_i)^2$$

$h(x) = ax$ 이므로

$J(1) = 0$ 임!

$a = \frac{1}{2}$ 이면

$$J\left(\frac{1}{2}\right) = \sum_{i=1}^{3} \left(\frac{1}{2}x - t_i\right)^2$$

$$= \frac{1}{2 \cdot 3}\left[\left(\frac{1}{2}-1\right)^2 + \left(\frac{2}{2}-2\right)^2 + \left(\frac{3}{2}-3\right)^2\right]$$

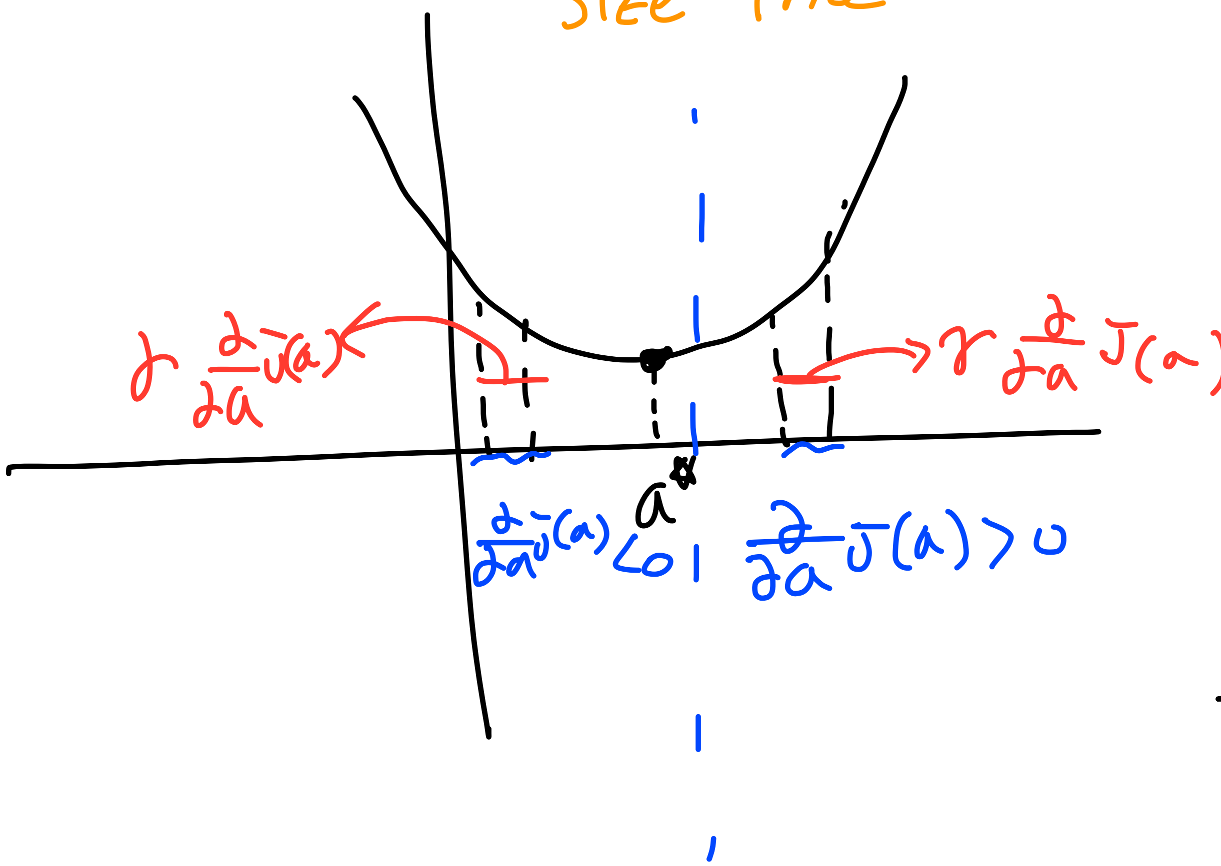
$$= \frac{1}{6} \cdot \left(\frac{1}{4} + 1 + \frac{9}{4}\right) = \frac{14}{24} \simeq 0.583$$

$a = 0$ 이면

$$J(0) = \sum_{i=1}^{N} (t_i)^2$$

$$\frac{1}{2 \cdot N}\left(1^2 + 2^2 + 3^2\right) = \frac{14}{6} \simeq 2.3$$

$$J(a) = \sum_{i=1}^{N} (\underbrace{ax_i}_{\text{Size}} - \underbrace{t_i}_{\text{Price}})^2 \qquad (h(x) = ax \text{ 일 때})$$



$$a_{\text{new}} = a - \gamma \frac{\partial}{\partial a} J(a, b)$$

하면할수록 a^*에 가까워짐.

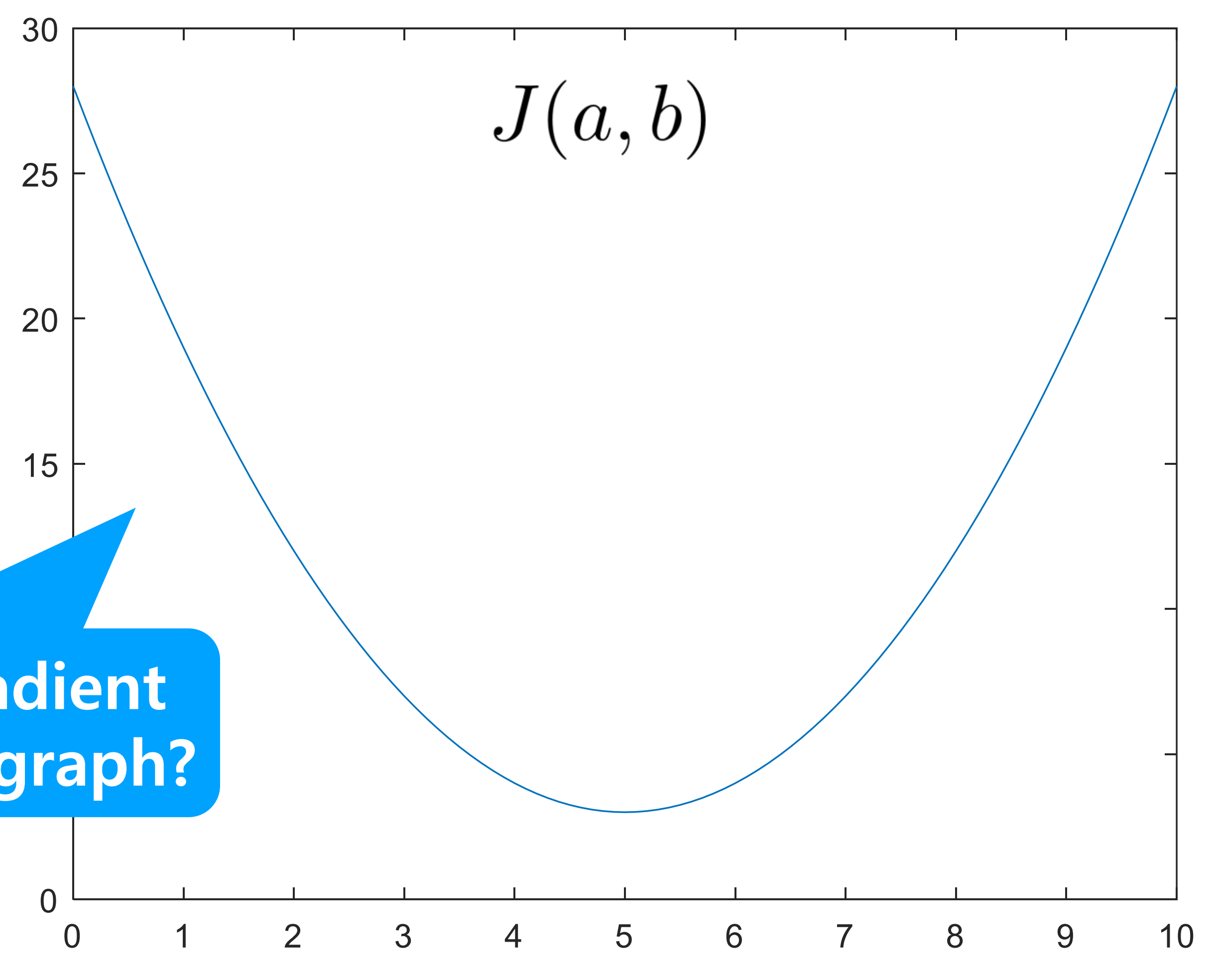
$h(x) = a^* x$ 가 됨.

Gradient Descent Approach

⇒ Cost 함수를 minimizing 하는 방향.

- The parameters can be found iteratively

$$a_{\text{new}} = a - \gamma \frac{\partial}{\partial a} J(a, b)$$



How we apply gradient descent with this graph?

Think

- Try to think:
  - What happen if the step size gamma is too small?

    ⇒ 너무 조금 줄어서 오랜시간이 걸린다.
  - Or too big?

    ⇒ 핑퐁문제가 발생한다.
  - When do we stop?

    ⇒ 더이상 줄어들지 않을때

Multi Dimensional Training Data (변수가 여러개 일때)

Size (feet²)	Number of bedrooms	Number of floors	Age of home (years)	Price ($1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178

How to model?

$$\mathbf{X} = \begin{bmatrix} 2104 & 5 & 1 & 45 \\ 1416 & 3 & 2 & 40 \\ 1534 & 3 & 2 & 30 \\ 852 & 2 & 1 & 36 \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} 460 \\ 232 \\ 315 \\ 178 \end{bmatrix}$$

Multivariate Linear Regression

- Our function is now

$$h(\mathbf{x}) = a_0 + a_1 x_1 + a_2 x_2 + a_3 x_3$$

Three-variable regression

$$h(\mathbf{x}) = \mathbf{a}^{\mathsf{T}}\mathbf{x}$$

$$\mathbf{x} = [1, x_1, x_2, x_3]$$

$$\mathbf{a} = [a_0, a_1, a_2, a_3]$$

$h(x) = a + bx$ 를 matrix로 변경시키면 가능

$$h(x) = [a\ b]\begin{bmatrix}1 \\ x\end{bmatrix}$$

이므로

$$h(x) = [a\ b\ c\ d]\begin{bmatrix}1 \\ x_1 \\ x_2 \\ x_3\end{bmatrix}$$

➡ Remaining question is now:

곱해진 값 편미분 시키면 됨.

How to apply Gradient descent?

Matrix-based Reformulation

- Vector-form cost function

$$J(\bar{\theta}) = \sum_{i=1}^{m} (h_{\theta}(\mathbf{x}) - \mathbf{y})^2$$

- Gradient descent

$$\theta_j \leftarrow \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\bar{\theta})$$

- Find:
  - $\min \sum_{i=1}^{N} (h(x_i) - t_i)^2$
- We define the cost function $J(a, b) = \sum_{i=1}^{N} (h(x_i) - t_i)^2$

Detail

- Assume we have one variable as before

$$\theta_0 \leftarrow \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\bar{\theta})$$

$$\Longleftrightarrow \quad \theta_0 \leftarrow \theta_0 - \alpha \frac{\partial}{\partial \theta_0} \sum_{i=1}^{m} (h_\theta(\mathbf{x}_i) - y_i)^2$$

$$\Longleftrightarrow \quad \theta_0 \leftarrow \theta_0 - 2\alpha \sum_{i=1}^{m} (h_\theta(\mathbf{x}_i) - y_i)$$

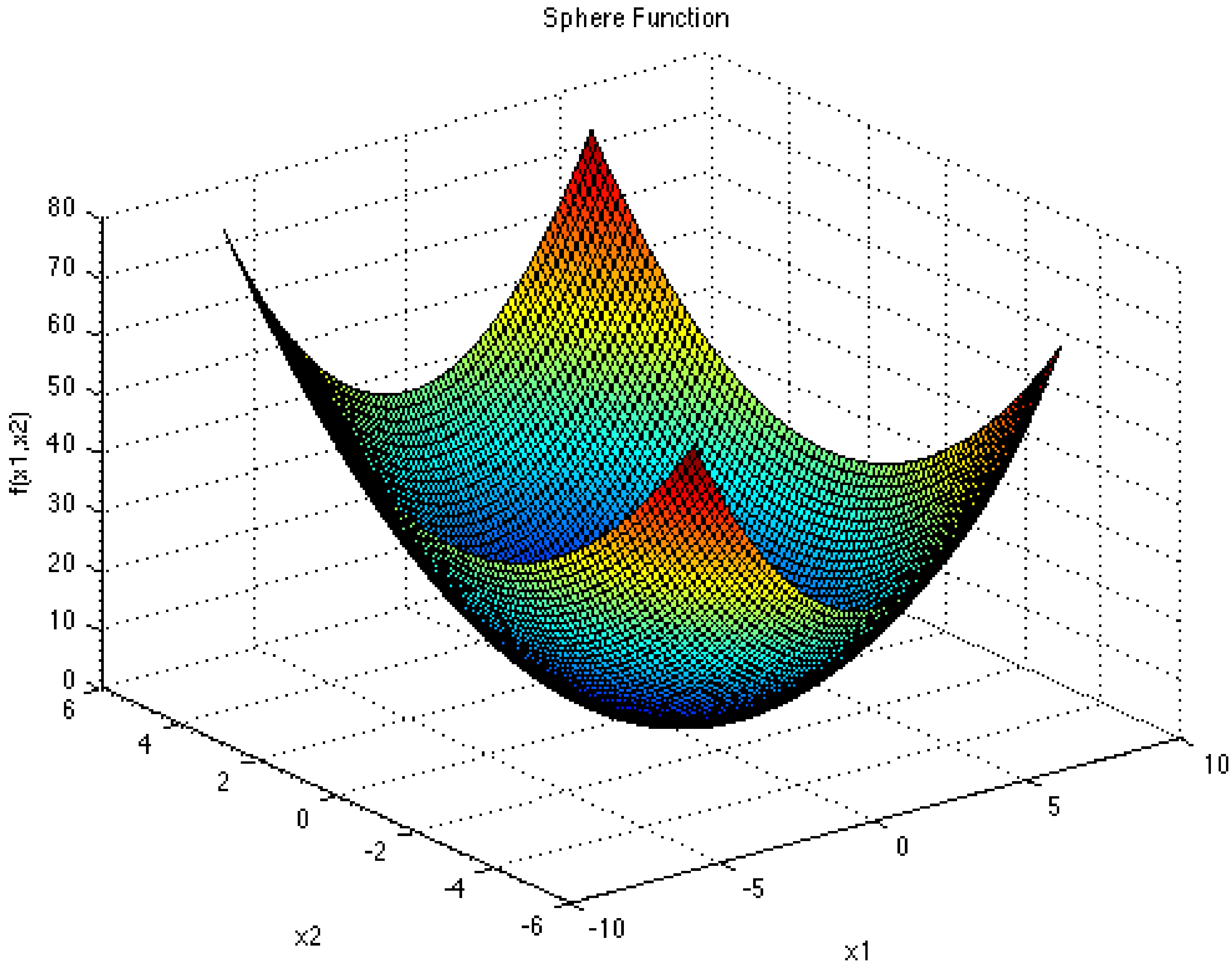
$$\theta_1 \leftarrow \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\bar{\theta})$$

$$\Longleftrightarrow \quad \theta_0 \leftarrow \theta_0 - 2\alpha \sum_{i=1}^{m} (h_\theta(\mathbf{x}_i) - y_i)\mathbf{x}_i(1)$$

Multivariate Case

$$\theta_j \leftarrow \theta_j - 2\alpha \sum_{i=1}^{m} (h_\theta(\mathbf{x}_i - y_i)\mathbf{x}_i(j))$$

Gradient Descent



Feature Scaling

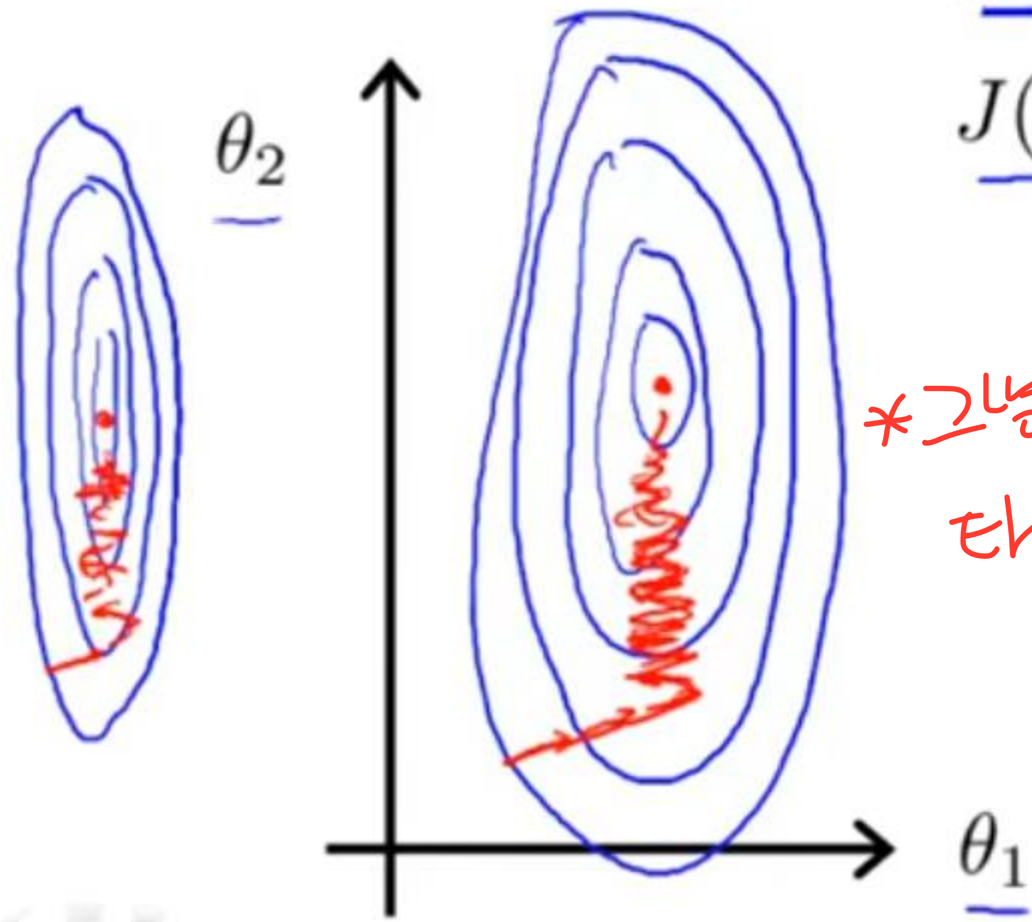
데이터를 다시 Normalize 해줌

Feature Scaling

Idea: Make sure features are on a similar scale.

E.g. x_1 = size (0-2000 feet²)

x_2 = number of bedrooms (1-5)

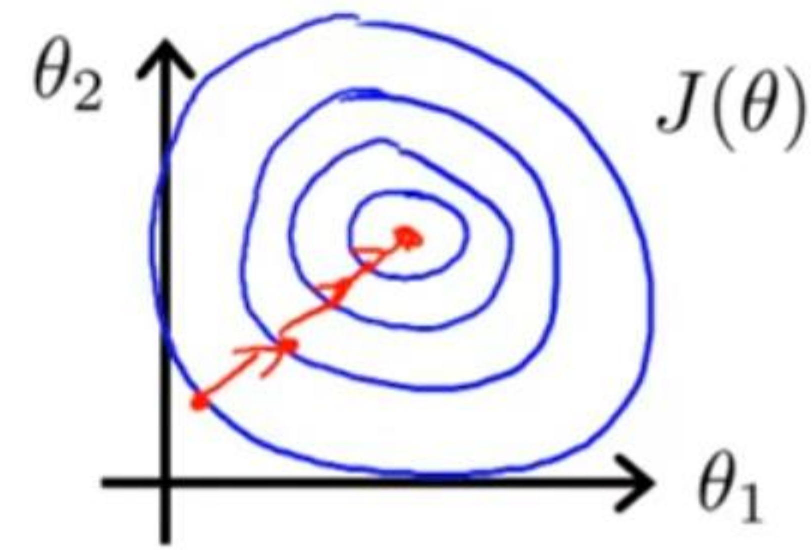


*그냥 진행시 타원

$$x_1 = \frac{\text{size (feet}^2)}{2000}$$

$$x_2 = \frac{\text{number of bedrooms}}{5}$$

*리스케일링후 원형



Andrew Ng

$$x_1 = \frac{\text{size} - 1000}{2000} \qquad x_2 = \frac{\text{방수} - 2}{5}$$

Feature Normalization

평균

$$x_1^{\text{new}} \leftarrow \frac{x_1 - \mu_1}{\underbrace{S_1}_{\text{Range (max - min)}}}$$

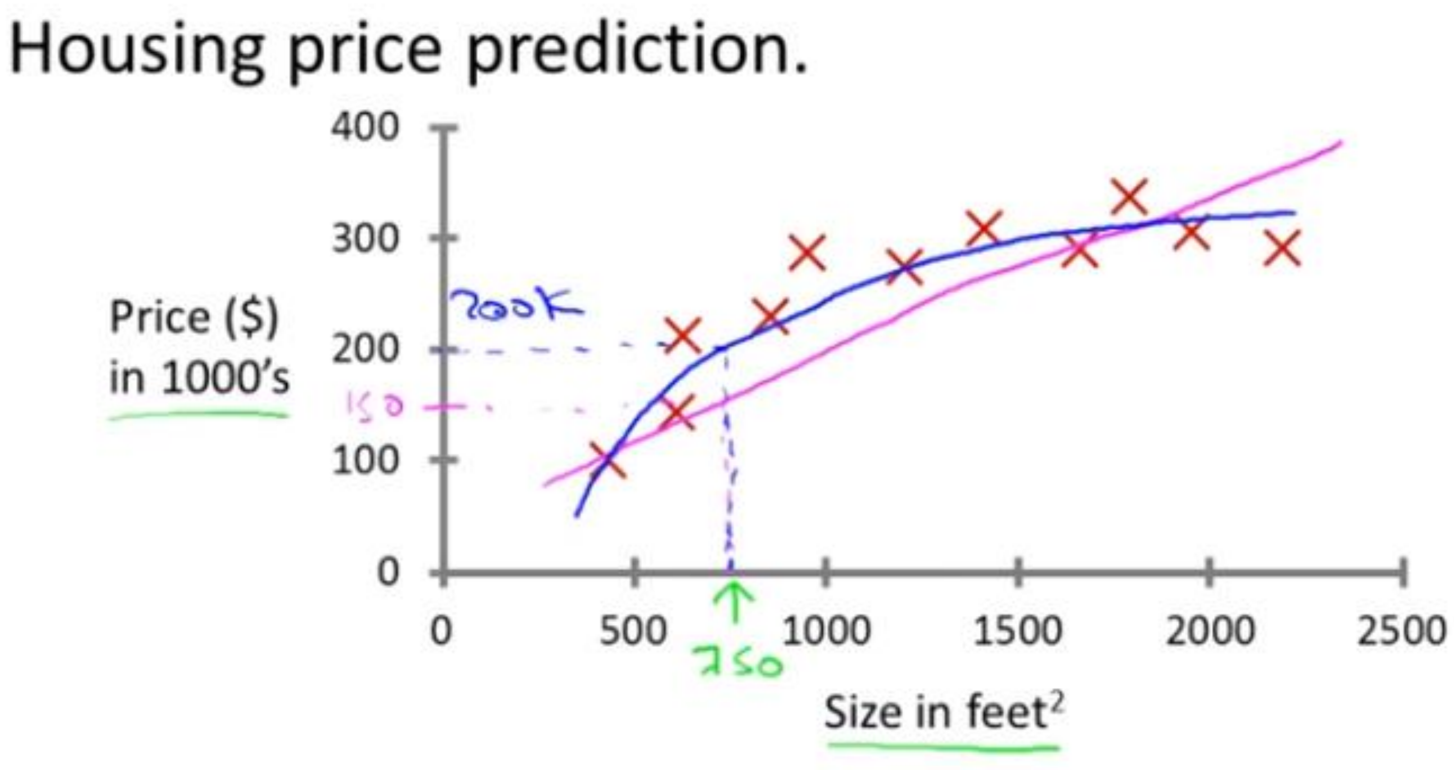
jeonghun.park@knu.ac.kr

Polynomial Regression

- We use a polynomial function for regression

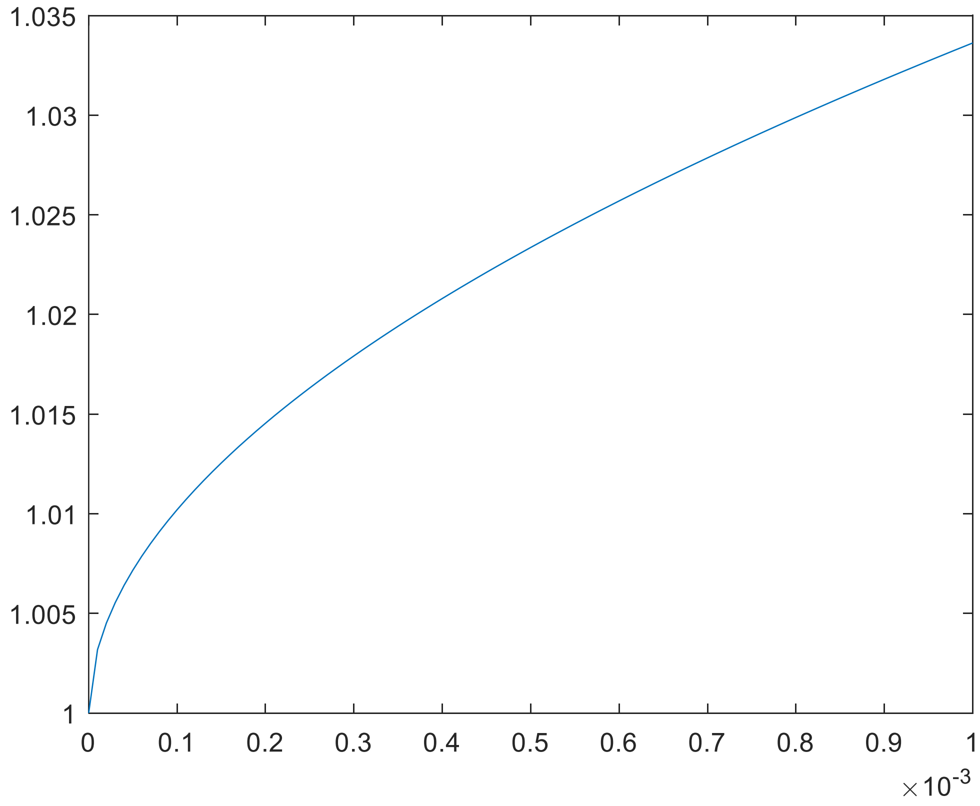
$$h_{\theta,2}(x) = \theta_0 + \theta_1 x + \theta_2 x^2$$

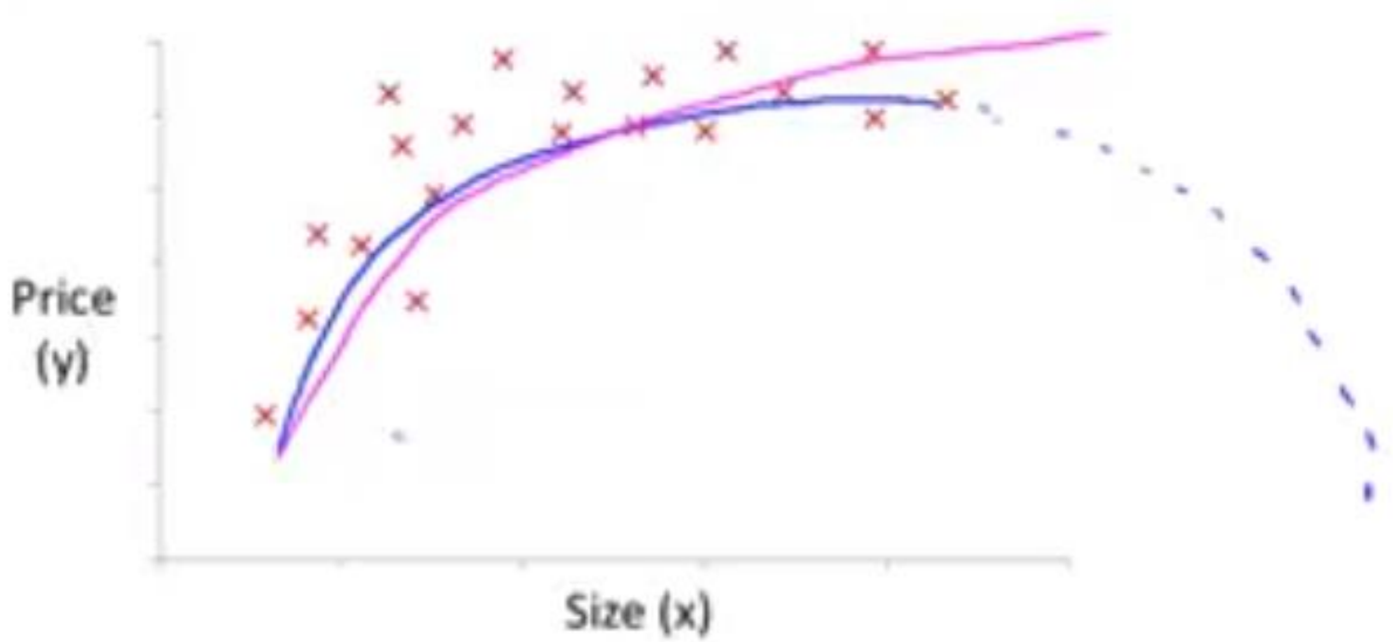
$$h_{\theta,3}(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3$$



Different Choice is Also Possible

$$h_{\theta,1/2}(x) = \theta_0 + \theta_1 \sqrt{x} + \theta_2 x$$



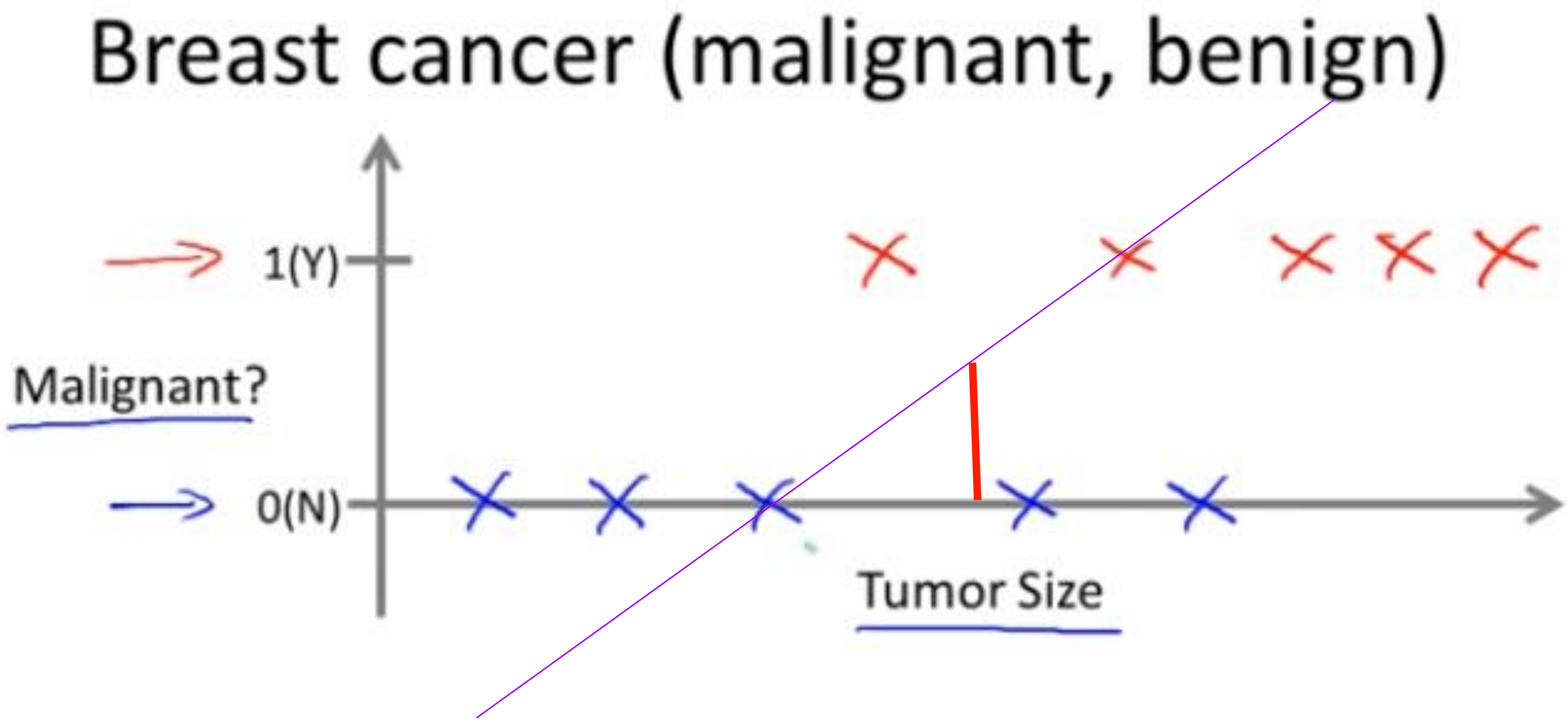


Normal Equation

- Any other [non-iterative] method?

Logistic Regression

- Regression for the classification problem



Logistic Regression

$$0 \leq h_\theta(x) \leq 1$$

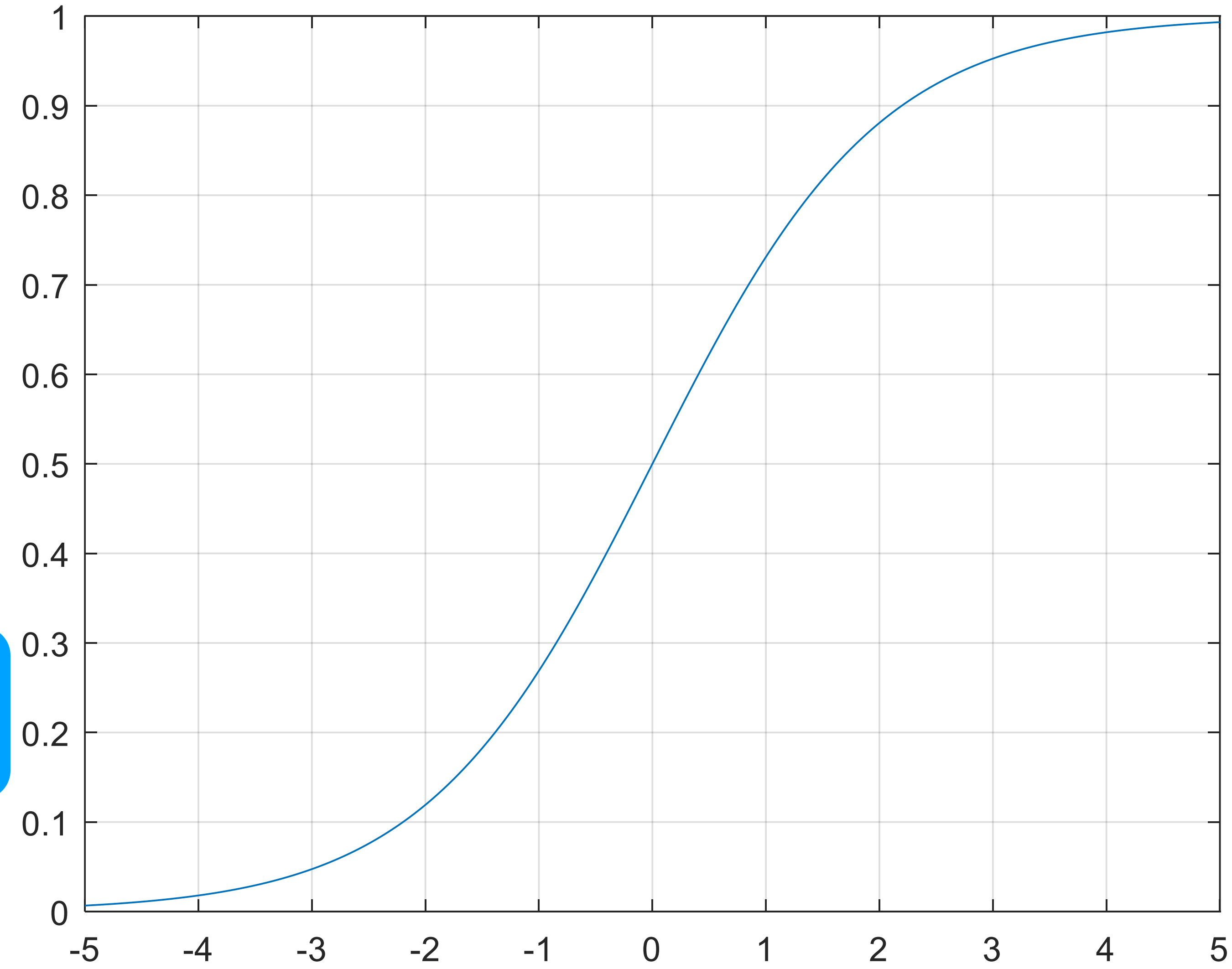
Logistic regression function

$$h_\theta(x) = g(\theta^T x), g(z) = \frac{1}{1+e^{-z}}$$

Sigmoid Function

- Or Logistic function

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$



Could be interpreted as probability