

Linear Regression

UOS Session 1

Study plan

<환경을 분석할 줄 아는 사람이 되자!>



Session 2. Machine learning (9/23)

Session 3. Logistic regression (10/7)

Session 4. Artificial neural network & Deep learning (10/21)

Session 5. Recurrent neural network (11/4)

Session 6. Long short-term memory (11/18)

Session 7. Gated recurrent unit (12/2)

Session 8. Encoder decoder (12/16)



+ project based learning



Contents

1. Machine learning

Linear regression

Gradient decent optimization

Multi variable linear regression

2. Project description

Artificial Intelligence

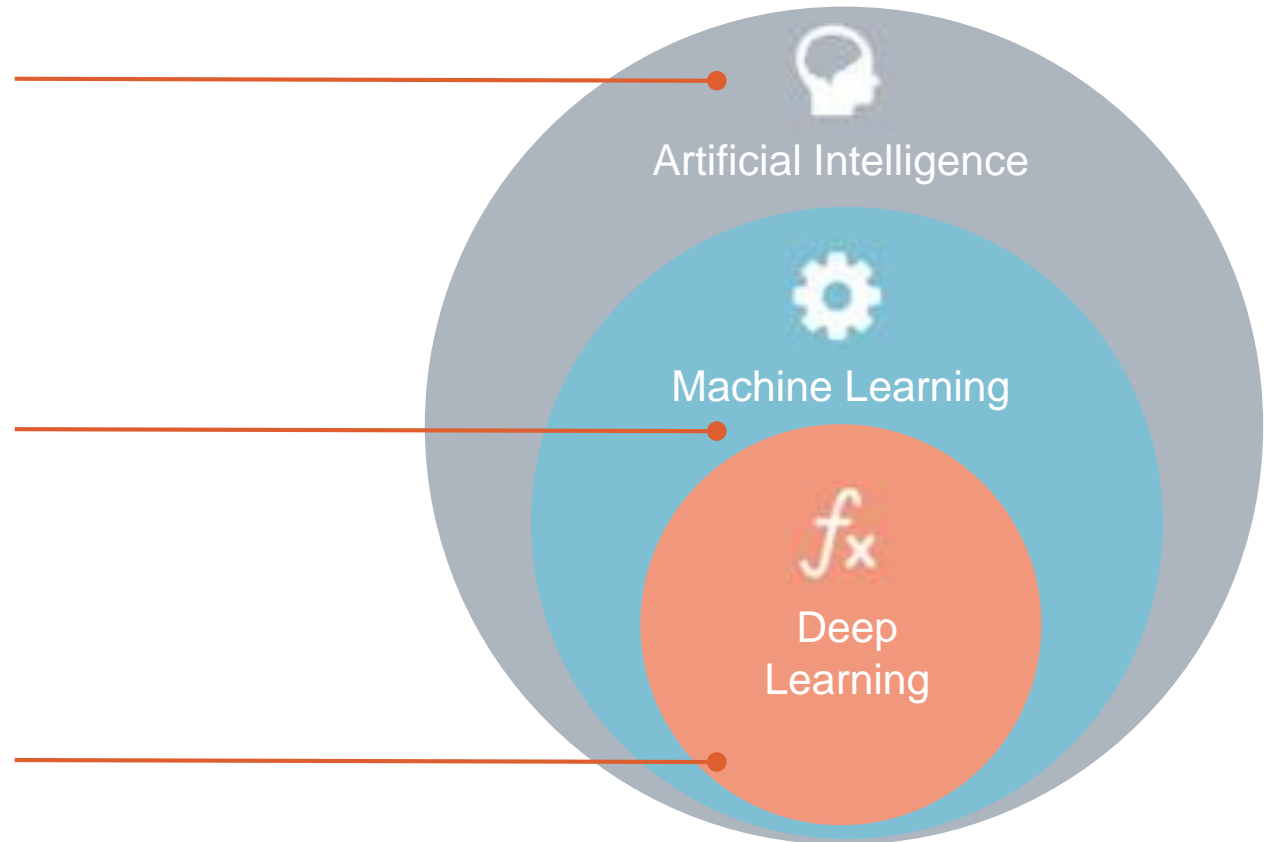
Any technique which enables computers to mimic human behavior.

Machine Learning

Subset of AI techniques which use statistical methods to enable machines to improve with experiences.

Deep Learning

Subset of ML which make the computation of multi-layer neural networks feasible.



1. Machine Learning

Linear regression

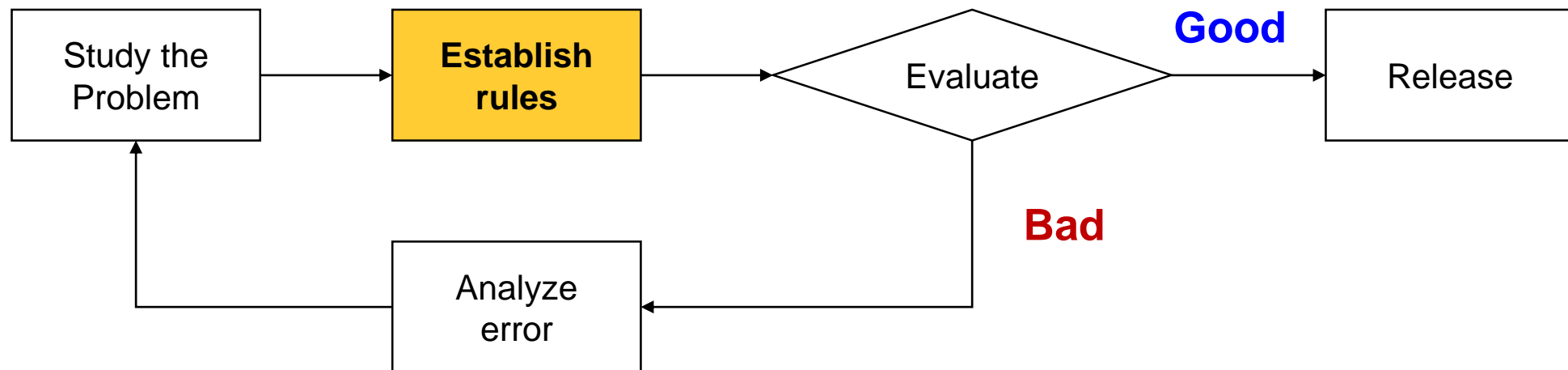
Gradient decent optimization

Multi variable linear regression

What is Machine Learning?

▪ Explicit programming

- 지식과 경험을 바탕으로 사람이 logic/rule을 설계
- 단편적인 현상을 통해 프로그래머가 인지한 몇 가지 rule을 가지고 문제를 해결



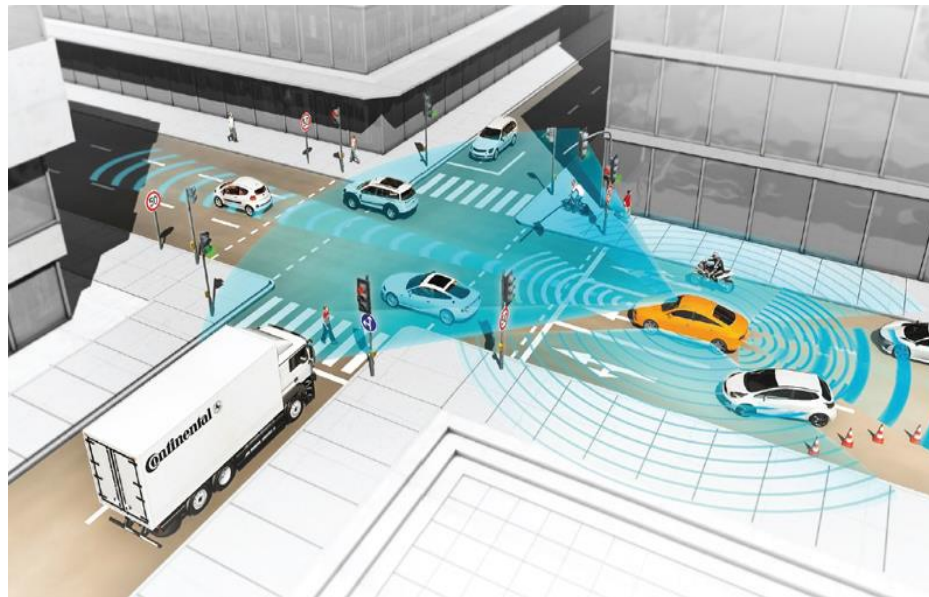
What is Machine Learning? (cont'd)

▪ Explicit programming

- 보편적인 rule을 만들기 어렵고, 엔지니어의 경험에 매우 의존적
- 규칙이 너무 복잡하거나, 많거나, 예상하지 못한 상황이 발생하는 경우에 취약
예) 바둑, 음성 인식, 자율 주행 등



<http://www.itcle.com>



© 오토모티브리포트

What is Machine Learning? (cont'd)



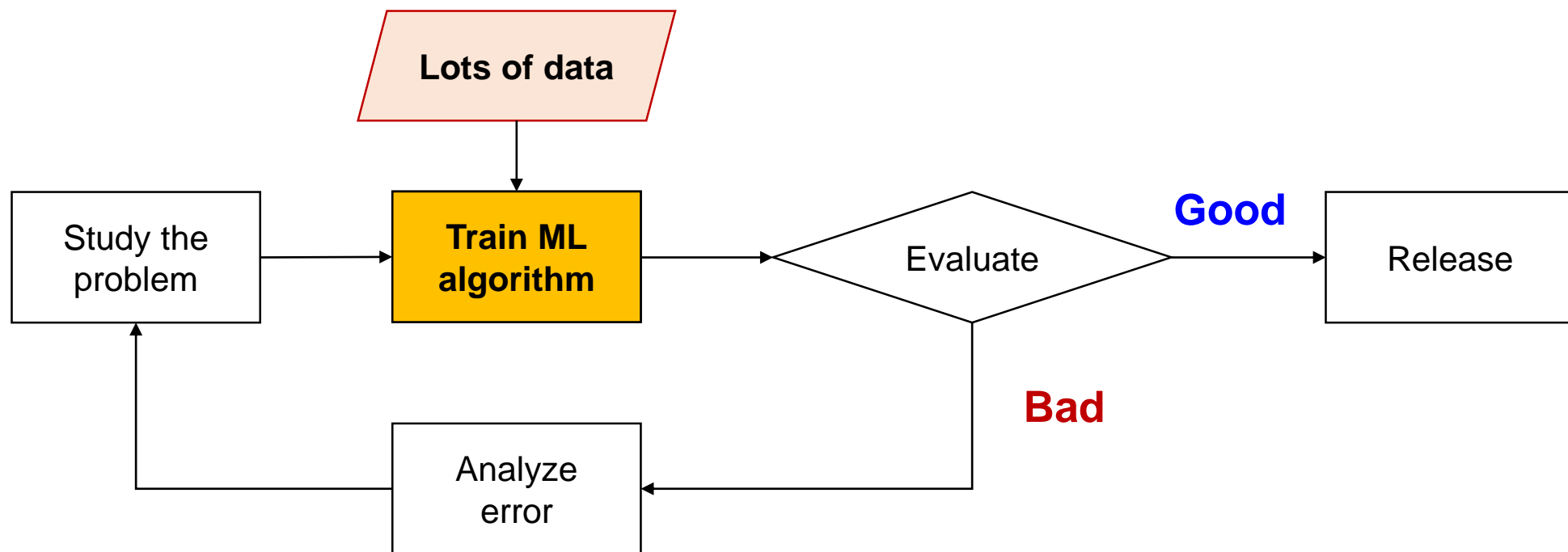
Machine learning is a field of study that gives computers the ability to learn without being explicitly programmed.

- Arthur Samuel, 1959

What is Machine Learning? (cont'd)

▪ Machine learning (ML)

- 주어진 data로부터 컴퓨터가 스스로 logic/rule을 학습
- Training dataset에 대한 일반화를 통해 문제 해결을 위한 보편적인 rule을 수립

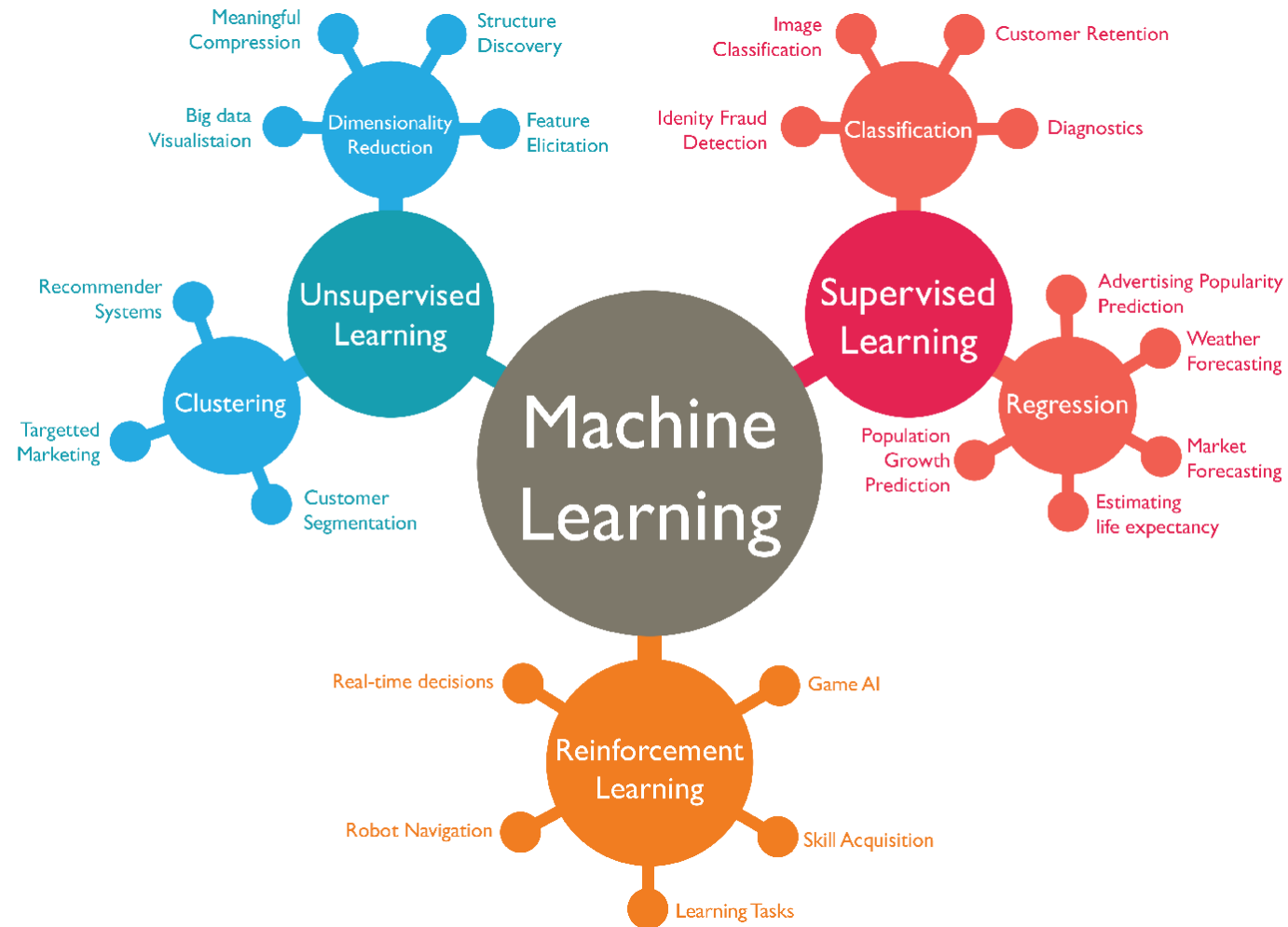


Characteristics of ML Algorithm

- All from data

- 1) Larger data → Better systems
- 2) Simple but universal rules
- 3) Easy update rule for given new data
- 4) Getting unrecognized insights about complex problems

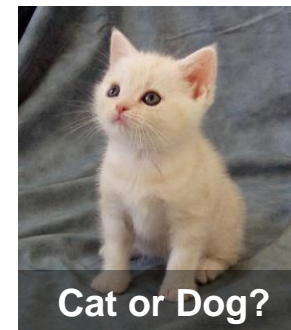
Types of ML Algorithms



Types of ML Algorithms (cont'd)

- **Supervised learning**

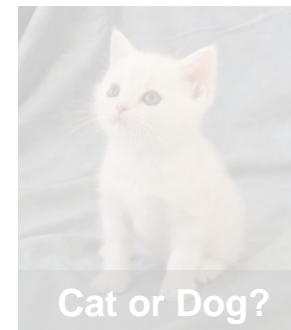
- **Labeled dataset**; 정답이 정해져 있는 데이터 쌍이 주어지는 경우
- 입력에 대한 모델의 예측 값과 '정답'의 차이가 줄도록 모델을 반복 수정
- 기존의 데이터들로부터 새로운 입력에 대한 결과를 예측하는데 사용



Types of ML Algorithms (cont'd)

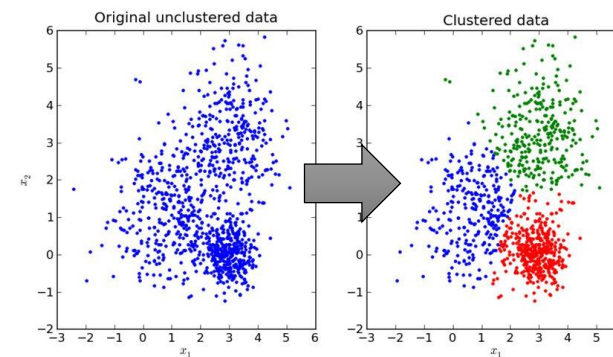
- **Supervised learning**

- **Labeled dataset**; 정답이 정해져 있는 데이터 쌍이 주어지는 경우
- 입력에 대한 모델의 예측 값과 '정답'의 차이가 줄도록 모델을 반복 수정
- 기존의 데이터들로부터 새로운 입력에 대한 결과를 예측하는데 사용



- **Unsupervised learning**

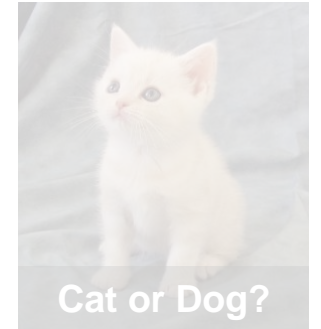
- **No labels**; 정답이 정해져 있지 않은 데이터가 주어지는 경우
- 데이터들 간의 관계에 기반해 숨겨진 패턴이나 형태, 구조를 도출
- 데이터의 특성을 분석, 분류하거나 데이터를 가공하는데 사용



Types of ML Algorithms (cont'd)

▪ Supervised learning

- **Labeled dataset**; 정답이 정해져 있는 데이터 쌍이 주어지는 경우
- 입력에 대한 모델의 예측 값과 '정답'의 차이가 줄도록 모델을 반복 수정
- 기존의 데이터들로부터 새로운 입력에 대한 결과를 예측하는데 사용



▪ Unsupervised learning

- **No labels**; 정답이 정해져 있지 않은 데이터가 주어지는 경우
- 데이터들 간의 관계에 기반해 숨겨진 패턴이나 형태, 구조를 도출
- 데이터의 특성을 분석, 분류하거나 데이터를 가공하는데 사용



▪ Reinforcement learning

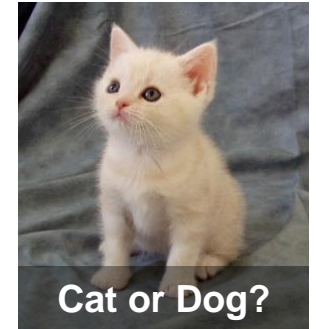
- **Reward**; 주어진 환경에서 설정된 보상이 최대화 되도록 하는 행동을 학습
- 동적인 상태(dynamic environment)에서 데이터를 수집하는 과정을 포함



Types of ML Algorithms (cont'd)

- **Supervised learning**

- **Labeled dataset**; 정답이 정해져 있는 데이터 쌍이 주어지는 경우
- 입력에 대한 모델의 예측 값과 '정답'의 차이가 줄도록 모델을 반복 수정
- 기존의 데이터들로부터 새로운 입력에 대한 결과를 예측하는데 사용



- **Unsupervised learning**

- **No labels**; 정답이 정해져 있지 않은 데이터가 주어지는 경우
- 데이터들 간의 관계에 기반해 숨겨진 패턴이나 형태, 구조를 도출
- 데이터의 특성을 분석, 분류하거나 데이터를 가공하는데 사용



- **Reinforcement learning**

- **Reward**; 주어진 환경에서 설정된 보상이 최대화 되도록 하는 행동을 학습
- 동적인 상태(dynamic environment)에서 데이터를 수집하는 과정을 포함



Types of Supervised Learning

- Regression (회귀)

- Input에 대응되는 **continuous output**을 찾는 문제
예) 집 값 예측(Housing price prediction), 시험 점수 예측

공부 시간	시험 점수
10	100
8	80
7	70
5	50
7.5	??

Types of Supervised Learning

▪ Regression (회귀)

- Input에 대응되는 continuous output을 찾는 문제
예) 집 값 예측(Housing price prediction), 시험 점수 예측

공부 시간	시험 점수
10	100
8	80
7	70
5	50
7.5	??

▪ Classification (분류)

- Input에 대응되는 **discrete output**을 찾는 문제
예) 손 글씨 판별(MNIST), 시험 성적 예측

공부 시간	성적
10	A
8	B
7	C
5	F
7.5	??

1. Machine Learning

Linear regression

Gradient decent optimization

Multi variable linear regression

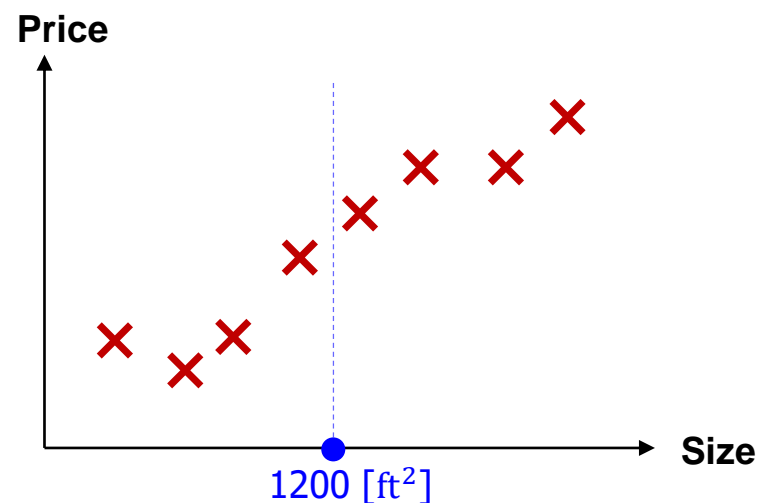
Linear Regression

▪ Linear regression 이란?

- 주어진 dataset을 기반으로 새로운 입력에 대한 결과값을 예측
- 정확한 결과 값을 예측하기 위해 주어진 dataset에서 관계를 유추
- 입력과 대응되는 정답 사이의 관계를 linear model로 가정
- 입력(feature)과 결과(target)의 관계를 대표하는 직선을 찾는 문제

Table. House price prediction

Size[feet ²]	Price (1000\$)
2104	460
1416	232
1534	315
852	178
⋮	⋮



Linear Regression (cont'd)

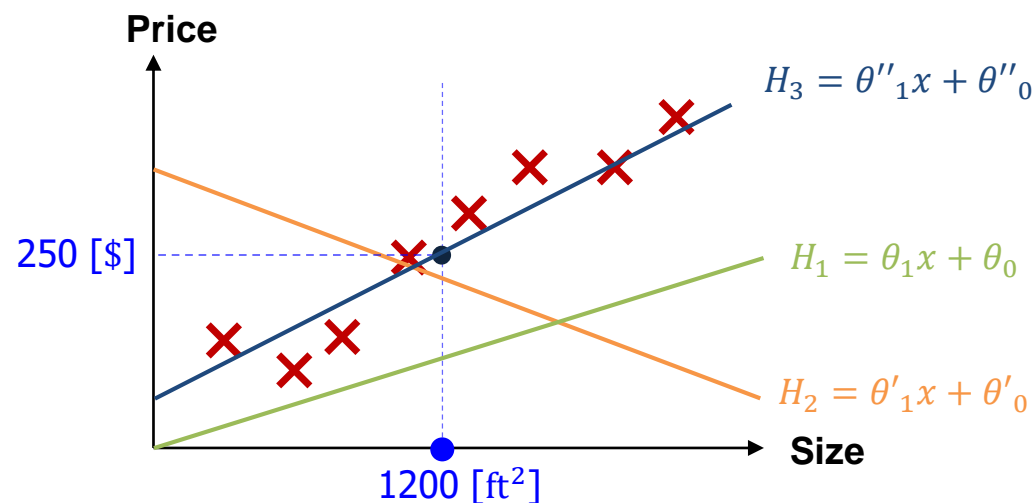
▪ Hypothesis: Linear model

$$H(x) = \theta_1 x + \theta_0$$

- 입력 x 와 결과 y 사이의 관계를 추정하기 위한 가설(hypothesis)을 수립
- $\theta = [\theta_0, \theta_1]$ 에 따라 여러 다른 모델 $H(x)$ 와 예측 결과를 생성
- 주어진 dataset에 알맞은 최적의 θ 는 어떻게 알 수 있을까?

Table. House price prediction

Size[feet ²]	Price (1000\$)
2104	460
1416	232
1534	315
852	178
⋮	⋮

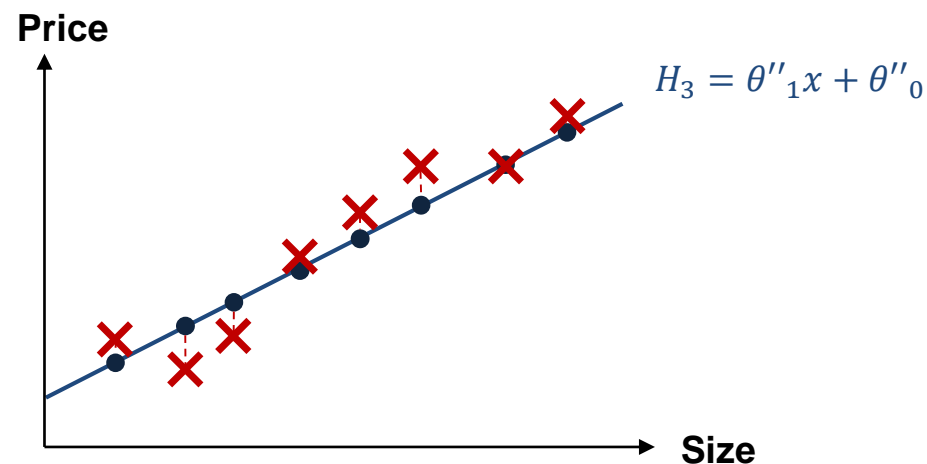
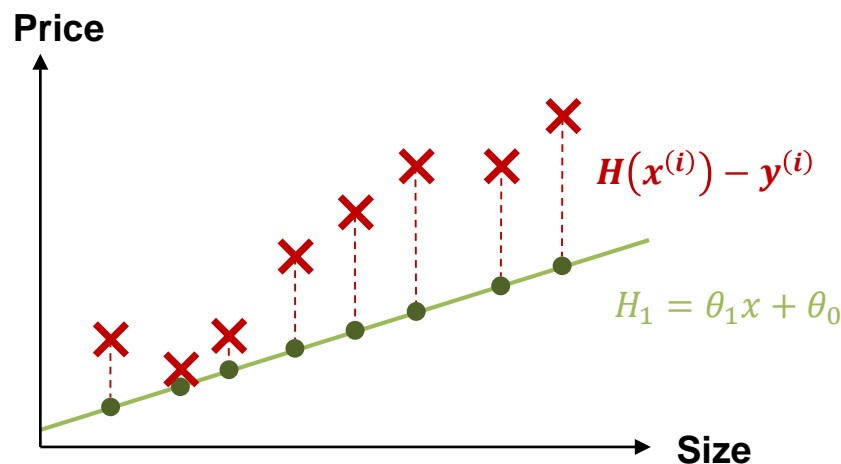


Linear Regression (cont'd)

▪ Cost function

$$\text{cost}(\theta) = \frac{1}{m} \sum_{i=1}^m \left(\underbrace{H(x^{(i)})}_{\text{Prediction}} - \underbrace{y^{(i)}}_{\text{True value}} \right)^2$$

- Hypothesis를 통한 prediction이 dataset에 얼마나 잘 맞는지를 정량적으로 나타내는 함수
- 정의한 hypothesis에 대해 cost function을 최소화하는 parameter θ 를 구하는 것이 목적



1. Machine Learning

Linear regression

Gradient decent optimization

Multi variable linear regression

Gradient Descent Method

▪ Gradient descent algorithm

- 주어진 cost function을 최소화하는 parameters θ 를 찾는 방법
- 현재 위치에서 기울기가 가장 급격한 방향으로 조금씩 이동하며 최소값을 탐색

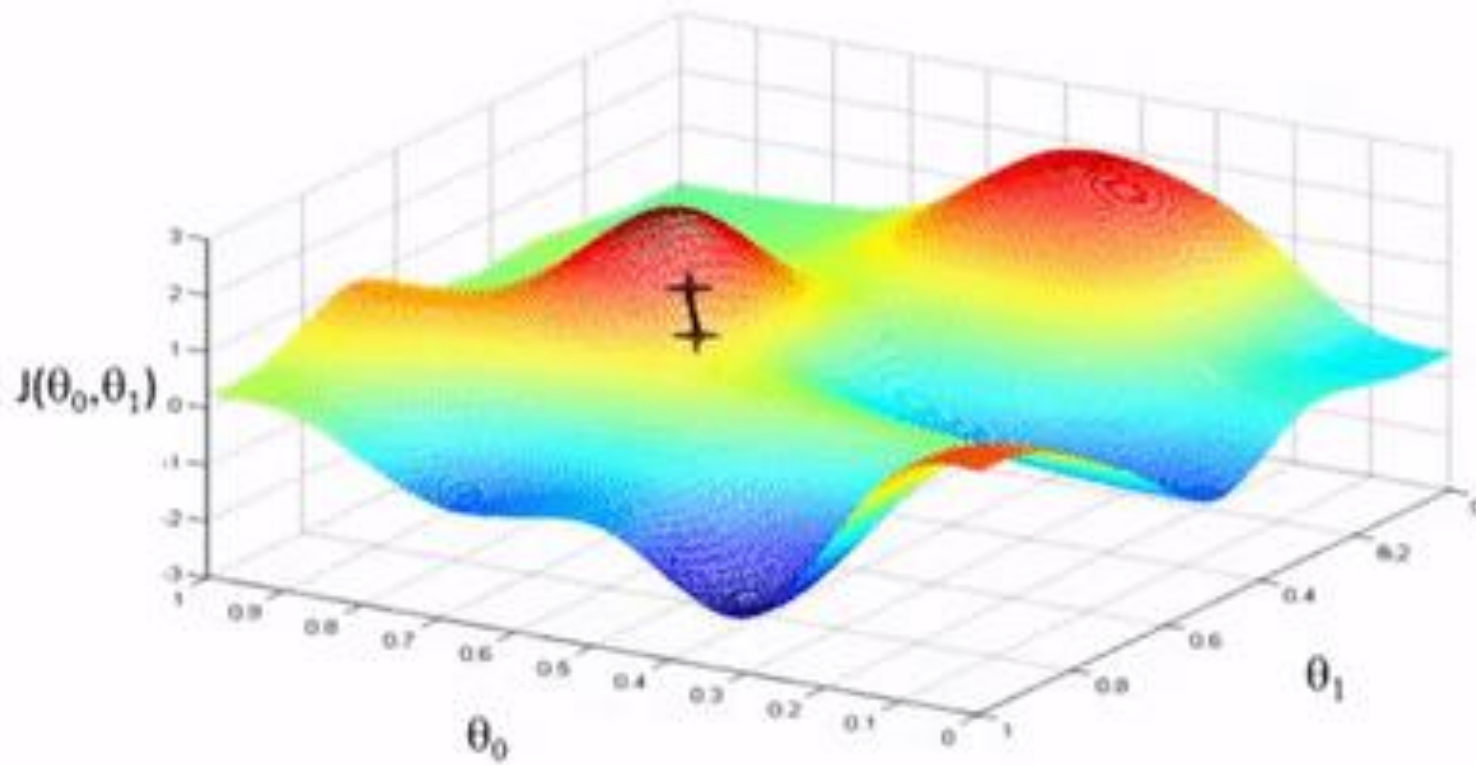
- 1) Start with some initial θ
- 2) Update θ to reduce $cost(\theta)$ using a gradient of θ
- 3) Repeat until we hopefully end up at (local) minimum

$$\theta_j := \theta_j - \underbrace{\alpha}_{\text{learning rate}} \underbrace{\frac{\partial}{\partial \theta_j} cost(\theta)}_{\text{gradient w.r.t } \theta_j}$$

* **learning rate** $\alpha > 0$: 한 번에 움직이는 크기

* **gradient** $\frac{\partial}{\partial \theta_j} cost(\theta)$: 이동할 방향과 크기

Gradient Descent Method (cont'd)



Exercise

- Simple example for intuition

x (hour)	y (score)
1	1
2	2
3	3

- Hypothesis: $H(x) = \theta_1 x$
- Cost function: $J(\theta_1) = \frac{1}{m} \sum_{i=1}^m (H(x^{(i)}) - y^{(i)})^2$
- Gradient descent: $\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_1)$

$$J(0) = \frac{(0*1-1)^2 + (0*2-2)^2 + (0*3-3)^2}{3} = 4.66$$

$$J(1) = \frac{(1*1-1)^2 + (1*2-2)^2 + (1*3-3)^2}{3} = 0$$

← Minimum cost @ $\theta_1 = 1$

$$J(2) = \frac{(2*1-1)^2 + (2*2-2)^2 + (2*3-3)^2}{3} = 4.66$$

$$J(3) = \frac{(3*1-1)^2 + (3*2-2)^2 + (3*3-3)^2}{3} = 18.66$$

Exercise (cont'd)

- How cost function looks like ?

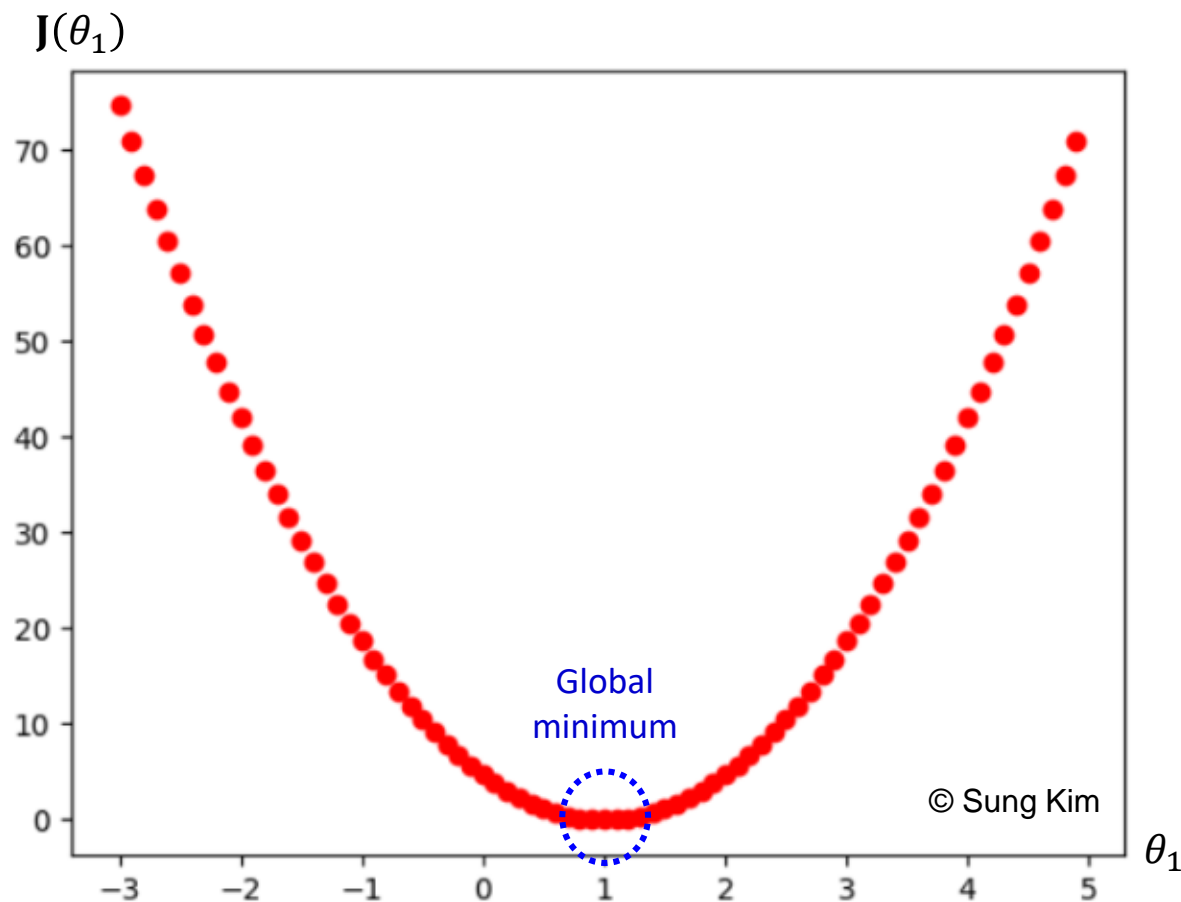
x (hour)	y (score)
1	1
2	2
3	3

$$J(0) = \frac{(0*1-1)^2 + (0*2-2)^2 + (0*3-3)^2}{3} = 4.66$$

$$J(1) = \frac{(1*1-1)^2 + (1*2-2)^2 + (1*3-3)^2}{3} = 0$$

$$J(2) = \frac{(2*1-1)^2 + (2*2-2)^2 + (2*3-3)^2}{3} = 4.66$$

$$J(3) = \frac{(3*1-1)^2 + (3*2-2)^2 + (3*3-3)^2}{3} = 18.66$$



Exercise (cont'd)

- Update parameter iteratively

Gradient descent: $\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_1)$

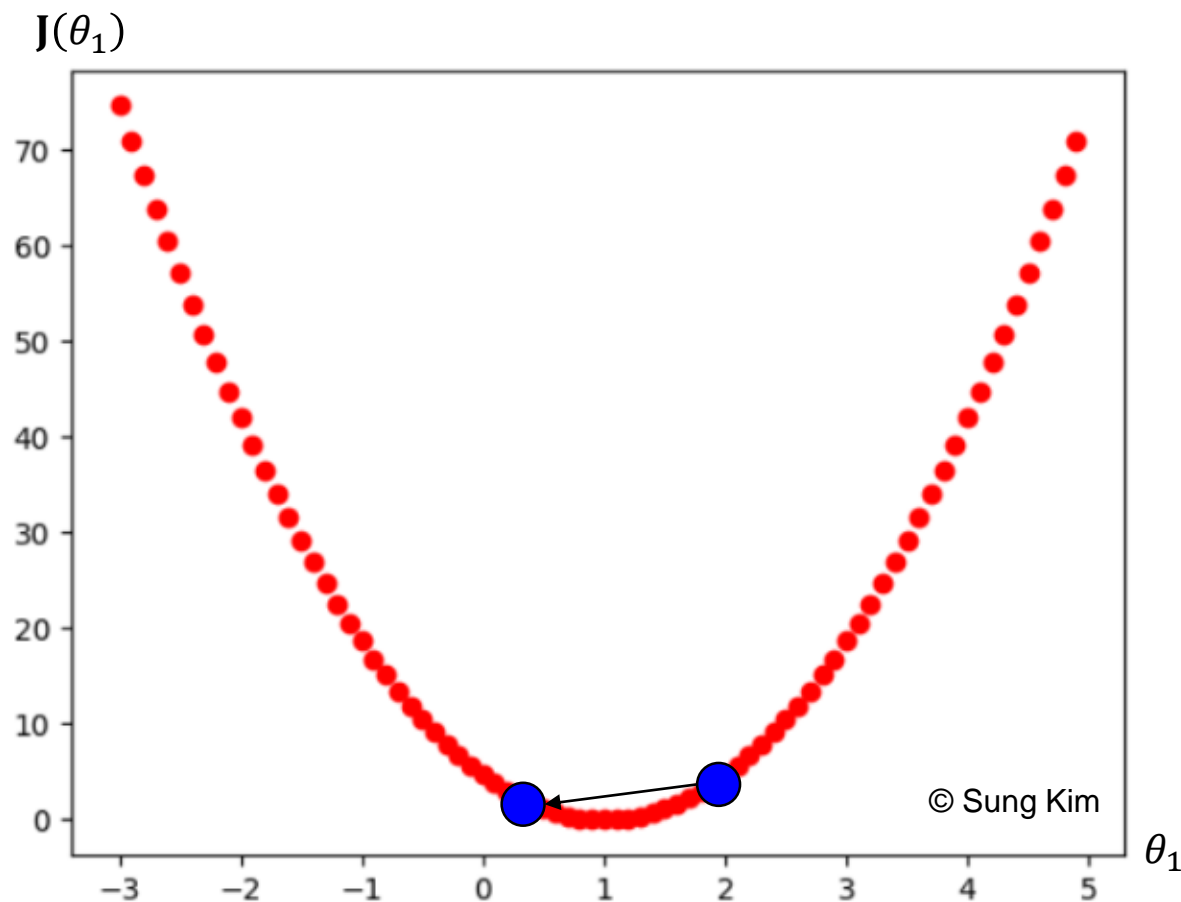
$$\frac{\partial}{\partial \theta_1} J(\theta_1) = \frac{2}{m} \sum_{i=1}^m x^{(i)} (\theta_1 x^{(i)} - y^{(i)})$$

Initialize: $\theta_1 = 2$, $\alpha = 0.1$

Repeat:

$$\frac{\partial}{\partial \theta_1} J(2) = \frac{2 * (1 * (2 * 1 - 1)) + 2 * (2 * 2 - 2) + 3 * (3 * 3 - 3))}{3}$$

$$\theta_1 := 2 - 0.1 * 15.3 = 0.47$$



Exercise (cont'd)

- Update parameter iteratively

Gradient descent: $\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_1)$

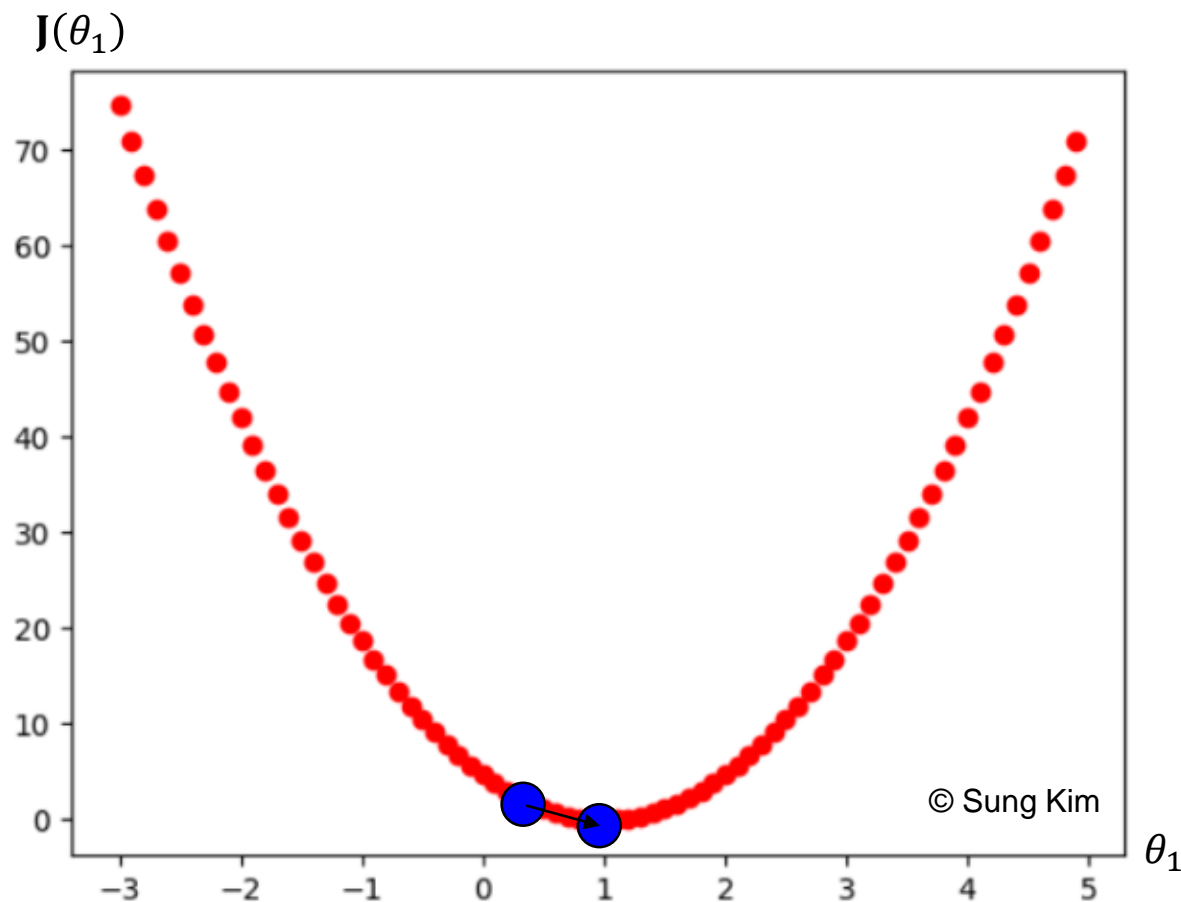
$$\frac{\partial}{\partial \theta_1} J(\theta_1) = \frac{2}{m} \sum_{i=1}^m x^{(i)} (\theta_1 x^{(i)} - y^{(i)})$$

Initialize: $\theta_1 = 2$, $\alpha = 0.1$

Repeat: $\theta_1 = 0.47$

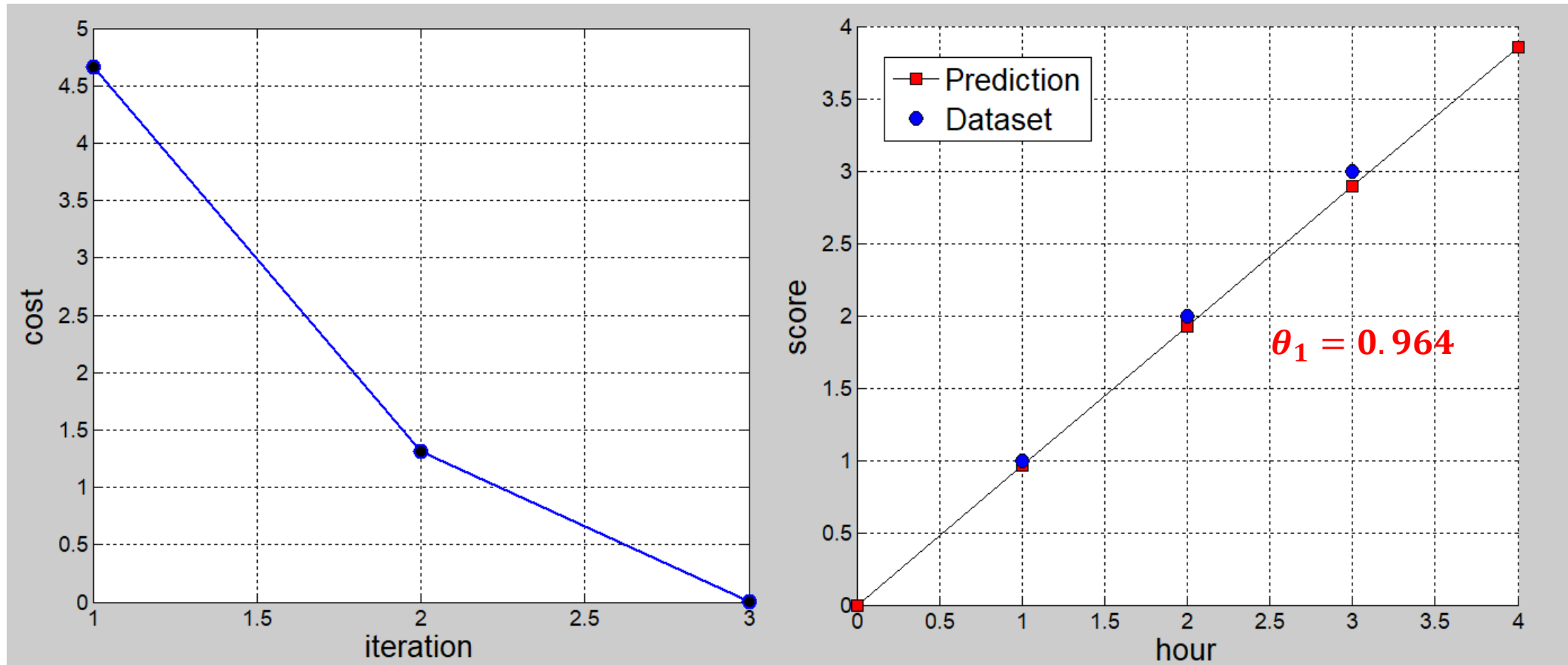
$$\frac{\partial}{\partial \theta_1} J(0.47) = -4.94$$

$$\theta_1 := 0.47 - 0.1 * (-4.94) = 0.964$$



Exercise (cont'd)

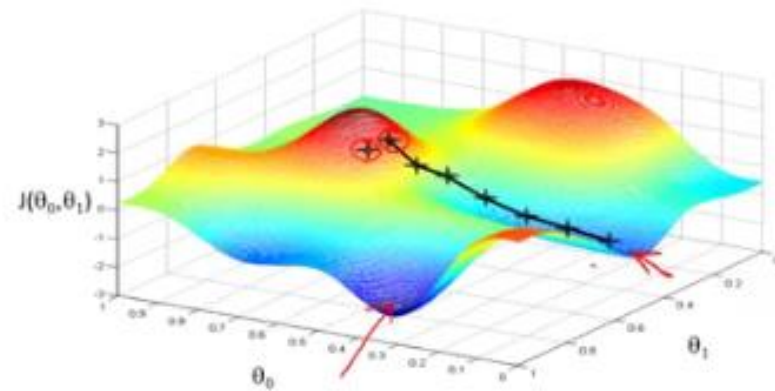
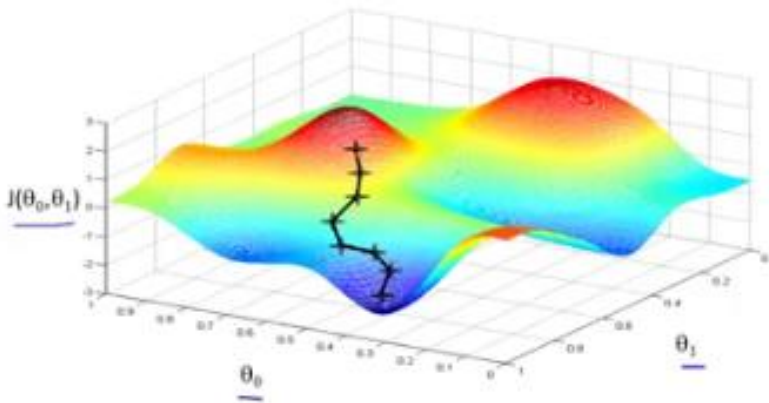
- Experimental results



Characteristics of GD

1. Initial value

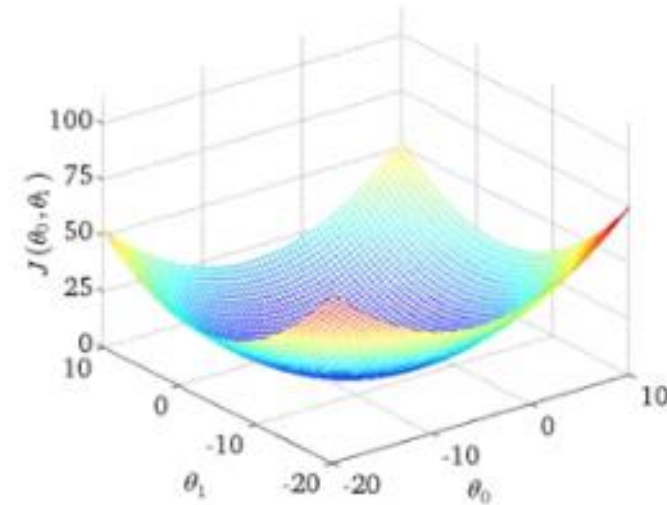
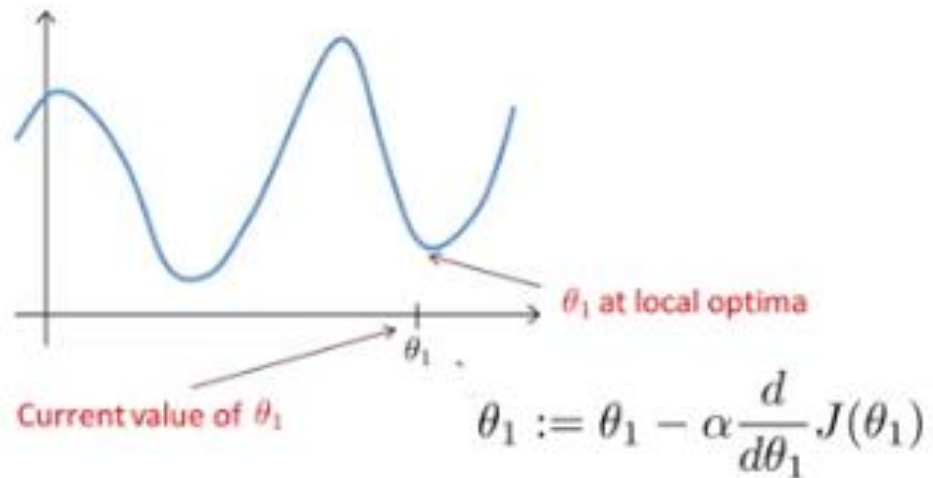
- 초기 값에 따라 학습 속도나 결과가 달라짐
- 가중치 초기 값의 설정이 매우 중요
- 일반적으로는 정규 분포에 따른 랜덤 값을 사용



Characteristics of GD

2. Local minima

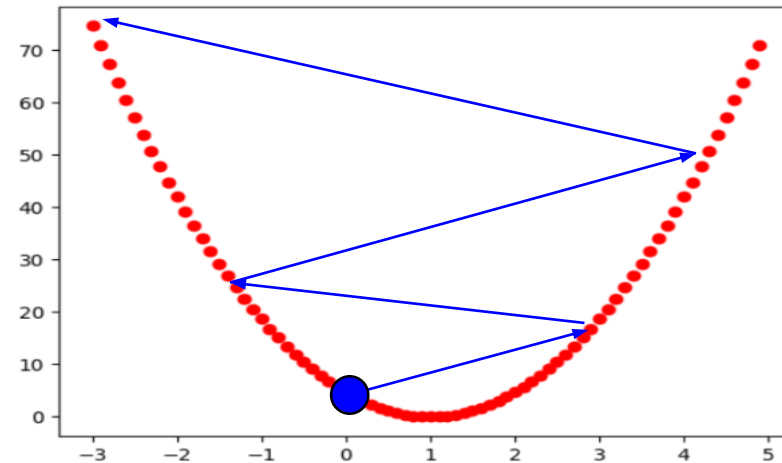
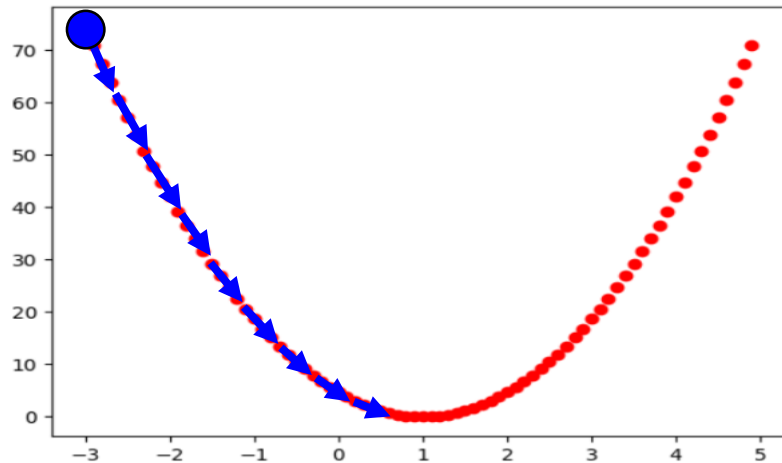
- Local optima에서 gradient가 '0'이 됨
- 더 이상 parameter 값이 업데이트 되지 않음 (학습 중지)
- Cost function이 convex function이 되도록 설계 / 적절한 learning rate 사용



Characteristics of GD

3. Learning rate

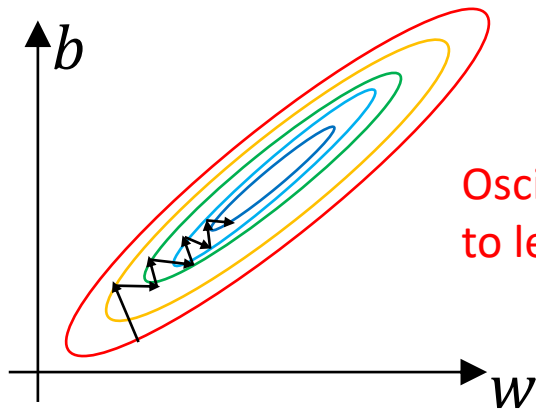
- Learning rate α 가 너무 작으면 수렴하는데 긴 학습 시간이 소요
- 최적 값에 수렴할수록 gradient의 크기가 작아져서 조금씩 변화함
- 반대로 너무 크면 최소값에 수렴하지 못하거나 발산



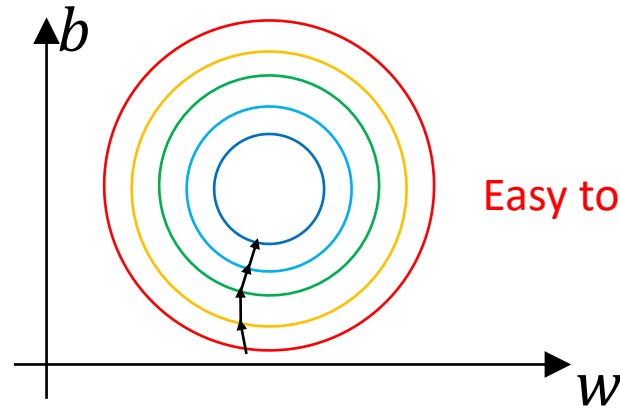
Characteristics of GD

4. Normalization

- Multi-variable regression의 경우 각각의 input의 scale이 다름
- Normalization을 통해 scale을 유사한 범위로 변환해야 학습 효율 향상



Oscillate and difficult
to learn



Easy to descent

1. Machine Learning

Linear regression

Gradient decent optimization

Multi variable linear regression

Multiple Features

▪ Multi-variable linear regression

- 여러 개의 inputs(features)를 고려하여 결과 값을 예측하는 방법
- Feature의 수가 많아짐에 따라, 학습해야 하는 parameter의 개수도 증가

Table. House price prediction

i	x_1	x_2	x_3	x_4	y
No.	Size[feet ²]	# bedrooms	# floors	Age [year]	Price (1000\$)
1	2104	5	1	45	460
2	1416	3	2	40	232
3	1534	3	2	30	315
4	852	2	1	36	178
5	⋮	⋮	⋮	⋮	⋮

$$H(\mathbf{x}) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

Multiple Features (cont'd)

- Matrix representation

- Hypothesis: $H(\mathbf{X}) = (\theta_0 + \theta_1 x_1 + \cdots \theta_n x_n) = (1 \quad x_1 \quad \cdots \quad x_n) \begin{pmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{pmatrix} = \mathbf{X}_{1 \times n} \boldsymbol{\theta}_{n \times 1}$
- Cost function: $J(\boldsymbol{\theta}_{n \times 1}) = \frac{1}{m} \sum_{i=1}^m \left(H(\mathbf{X}_{1 \times n}^{(i)}) - y^{(i)} \right)^2 = \frac{1}{m} (\mathbf{X}_{m \times n} \boldsymbol{\theta}_{n \times 1} - \mathbf{y}_{m \times 1})^T (\mathbf{X} \boldsymbol{\theta} - \mathbf{y})$
- Gradient descent: $\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\boldsymbol{\theta}) \rightarrow \boldsymbol{\theta} := \boldsymbol{\theta}_{n \times 1} - \frac{\alpha}{m} \mathbf{X}^T (\mathbf{X}_{m \times n} \boldsymbol{\theta}_{n \times 1} - \mathbf{y}_{m \times 1})$

Summary

▪ Machine learning (ML)

- Explicit programming vs. Machine learning
- ML: Supervised / Unsupervised / Reinforcement learning
- Supervised: Regression vs. Classification

▪ Linear regression

- Linear hypothesis
- Cost function (MSE; mean square error)
- Gradient descent algorithm

2. Project : 외래어종의 개체밀도 예측



- 물환경정보시스템
 - 생물측정망
 - 수질환경측정망
 - 퇴적물측정망