

3/23/2019

머신러닝과 빅데이터

장서영

CONTENTS

1. 빅데이터의 개념
2. 머신러닝의 개념
3. 머신러닝 모델
4. 모델 평가
5. 모델 최적화
6. Quest

빅데이터란?

- **빅데이터**는 디지털 환경에서 만들어지고 저장되는 데이터가 엄청난 속도로 증대됨으로써 생성된 대규모 데이터 세트

정형데이터/비정형데이터

- 정형데이터

구조화된 형태가 있고 연산 가능한 데이터

Ex) 관계형 데이터베이스, 스프레드시트

- 비정형데이터

구조화되지 않고 연산 불가능한 데이터

Ex) 텍스트, 그림, 음성, 영상

- Cf) 반정형데이터 : 연산이 불가능하지만 형태(schema)를 가진 데이터

Ex) xml, html, json

3/22/2019

머신러닝의 개념

5

머신러닝이란?

- 머신러닝(Machine Learning)

개념: 인공 지능을 구현하는 구체적 접근 방식 및 수단
(인공지능을 달성하기 위한 수단)

특징 : 1. AI로부터 파생
2. 기존 컴퓨터와 다른 새로운 능력을 포함

Cf) **딥러닝** : 머신러닝의 일종으로 인공신경망에서 발전, 뇌의 뉴런과 유사한 정보 입출력 계층을 활용해 데이터를 학습

머신러닝 활용 예시

1. 데이터베이스 수집

Ex) web click data, medical records

2. 수동적으로 프로그래밍을 할 수 없을 때 사용하는 **자동화 학습 프로그래밍 분야**

Ex) 자동헬리콥터, 손글씨 자동인식, NLP

3. 개인추천시스템

Ex) 아마존, 넷플릭스의 추천시스템

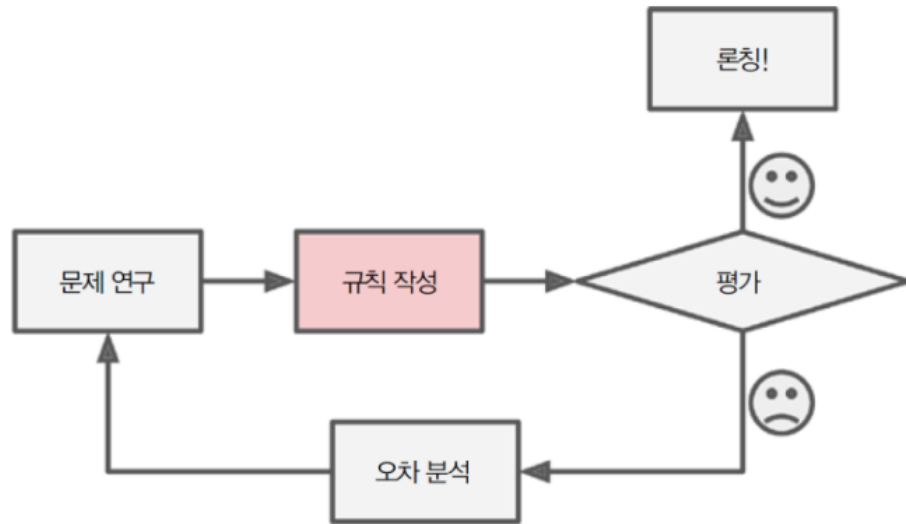
4. 인간의 학습 이해

Ex) real AI

전통적 컴퓨터 VS 머신러닝

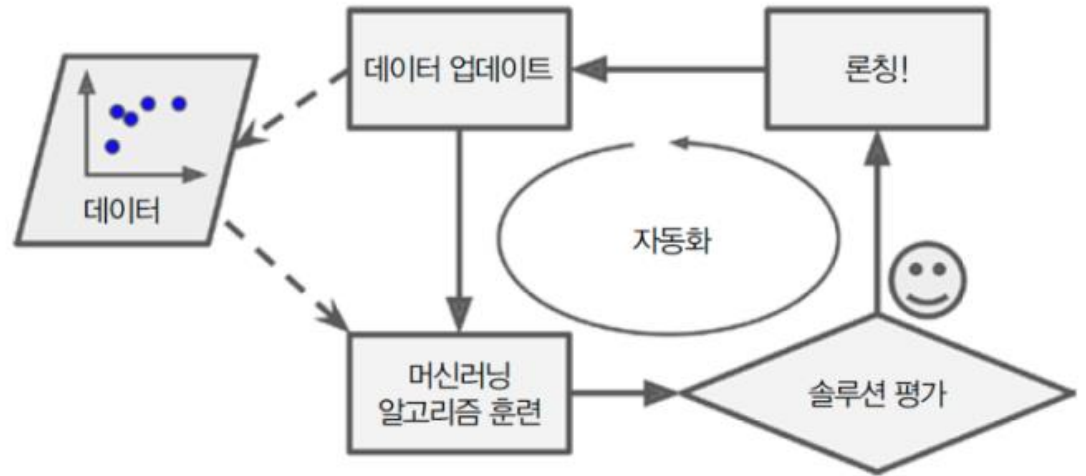
전통적 컴퓨터

규칙기반(rule based)의 접근법



머신러닝

사람이 직접 생각하는 방법을 알려주는 것이 아니라 기계가 스스로 배우도록 하는 것



머신러닝의 정의

“명시적인 프로그래밍 없이 컴퓨터가 학습하는 능력을 갖추게 하는 연구 분야다.”

아서 사무엘 Arthur Samuel, 1959

“어떤 과제 T 에 대한 성능이 P 라고 측정되고 경험 E 를 통해 향상된다면 프로그램은 과제 T 에 대해 경험 E 로부터 성능 기준 P 에 따라 학습한다고 할 수 있다”

톰 미첼 Tom Mitchell, 1997

예시

체스게임

E = 프로그램이 수만번의 게임을 스스로 수행했던 경험

T = 체스 게임을 하는 작업

P = 다음 체스 게임에서 새로운 적에게 이길 확률

3/22/2019

머신러닝 모델

11

모델이란?

라벨(label)

예측하는 항목 (y), target

특성(feature)

입력 변수 (x)

모델

- 특성과 라벨의 관계
- 다양한 변수 간의 수학적(or 확률적) 관계를 표현한 것

모델의 분류

지도학습(supervised learning)

데이터에 대한 레이블(명시적인 정답)이 주어진 상태에서 컴퓨터를 학습시키는 방법

비지도학습(unsupervised learning)

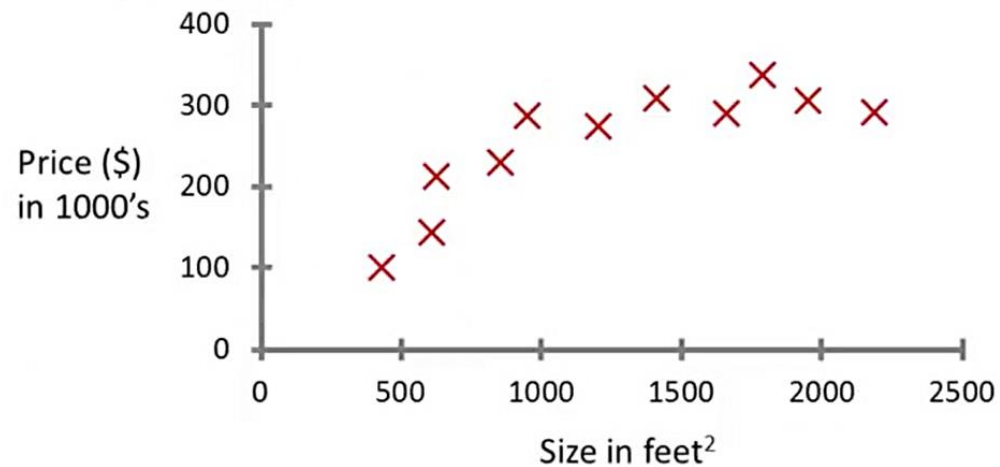
데이터에 대한 레이블(명시적인 정답)이 주어지지 않은 상태에서 컴퓨터를 학습시키는 방법

Cf) 준지도학습/강화학습/추천시스템

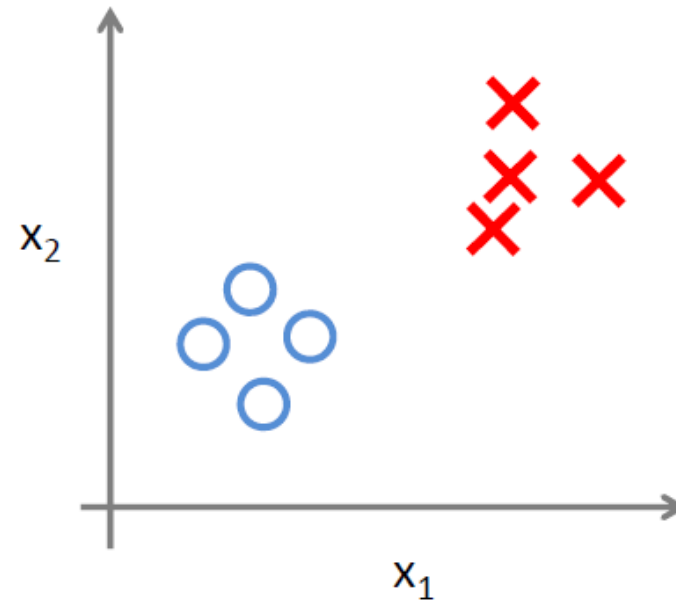
지도학습

Regression

Housing price prediction.

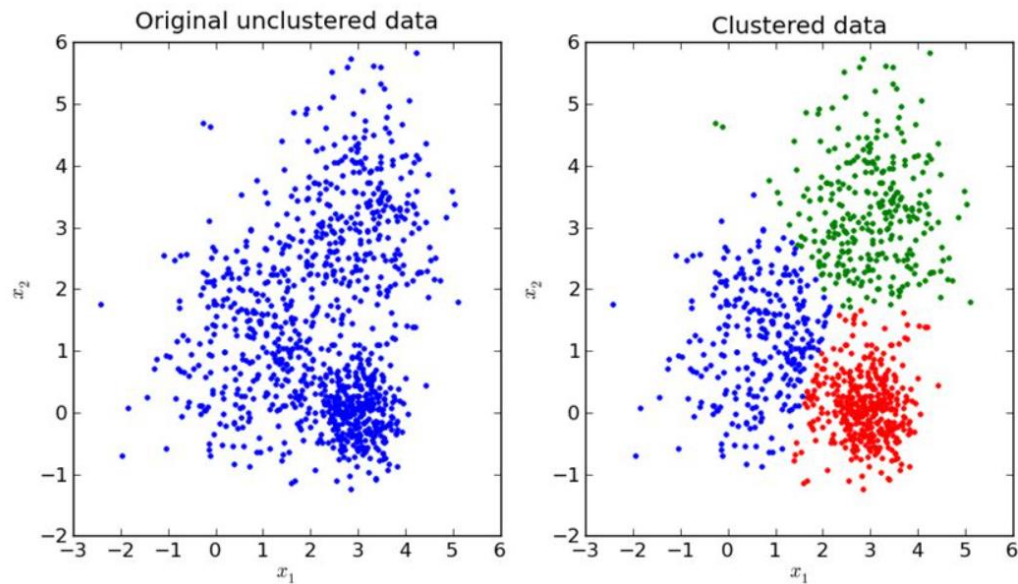


Classification

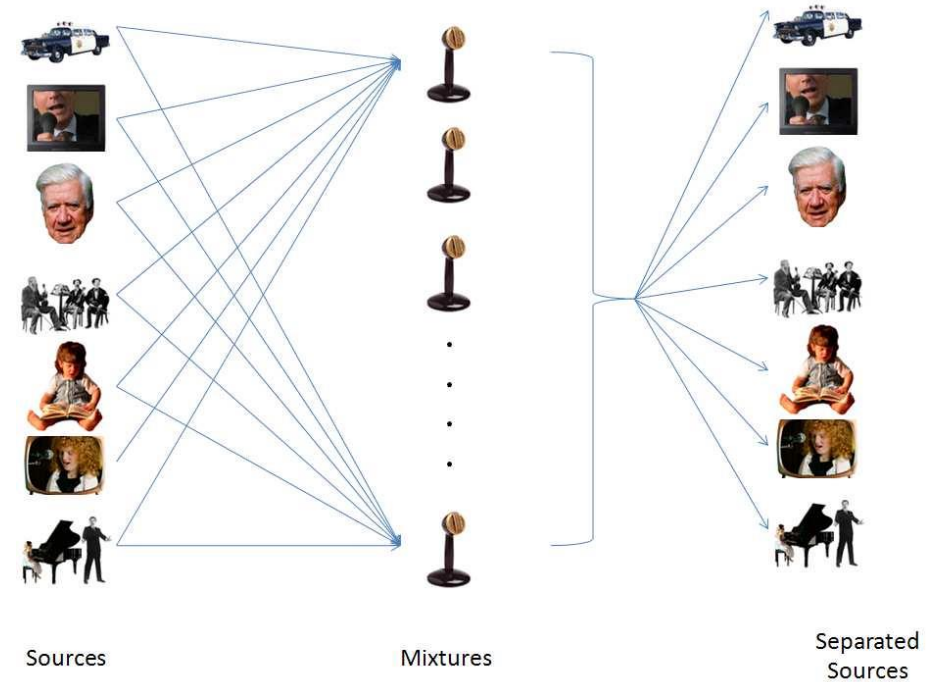


비지도학습

Clustering



Non-clustering



머신러닝 모델

지도학습	Classification	kNN
		Naïve Bayes
		Support Vector machine
		Decision Tree
	Regression	Linear regression
		Locally weighted linear regression
		Ridge
		Lasso
비지도학습		Clustering
		K means
		Density estimation
		Expectation maximization
		Pazen window
		DBSCAN

3/22/2019

모델 평가

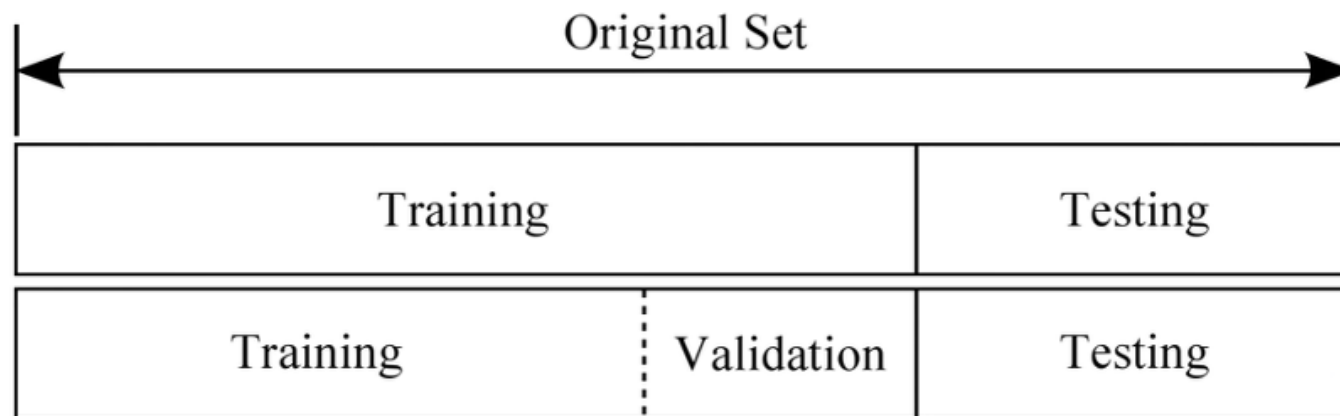
17

모델 평가

- **Train/Test split**

비율은 임의대로지만 대부분 7:3 or 6:2:2

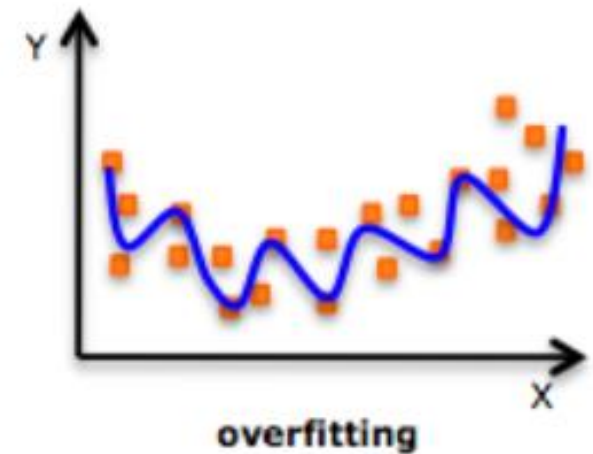
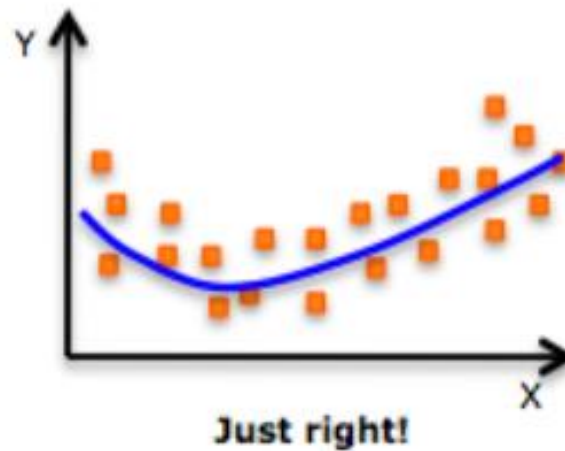
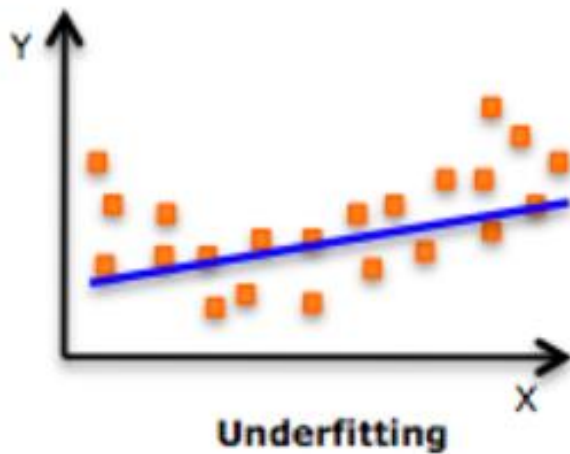
Train/validation/test set으로 나누어 validation set으로 모델이 여러 개일 때 최종 모델을 선정하기 위한 성능 평가를 하기도 한다



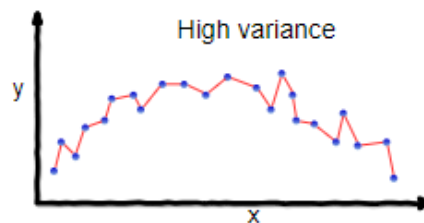
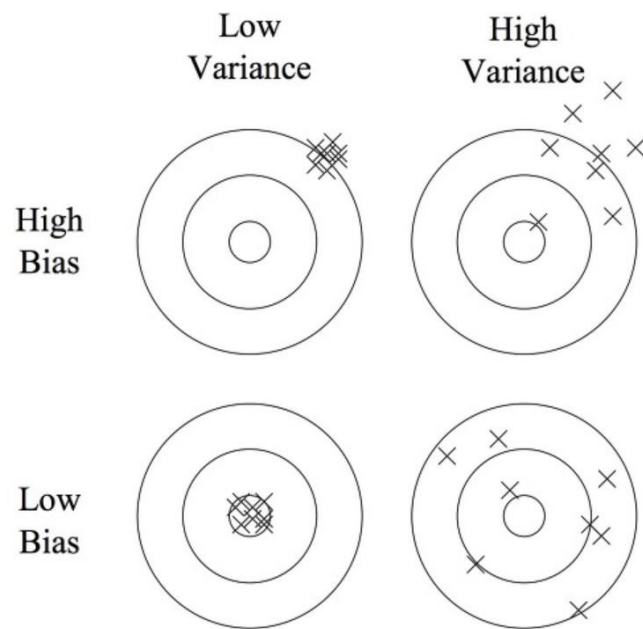
정확도

		Predicted Data		Total
		Predicted Condition POSITIVE	Predicted Condition NEGATIVE	
Actual Data	Condition TRUE	TP True Positive	FN False Negative	P
	Condition FALSE	FP False Positive	TN True Negative	N
Total		P⁺	N⁺	P+N

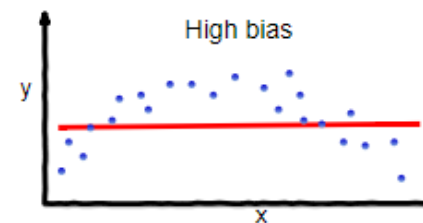
Overfitting vs Underfitting



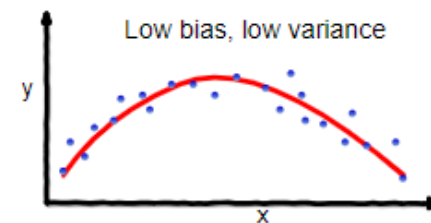
Bias vs Variance



overfitting



underfitting



Good balance

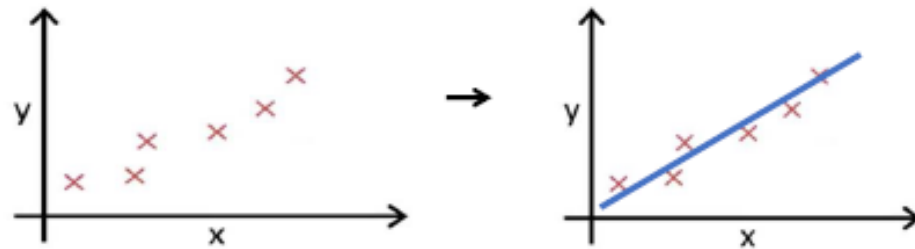
Trade-off 관계!

모델 평가 - 가설

가설(Hypothesis)

Input(feature)과 output(target)의 관계를 나타내는 함수

$$\text{예 : } h_{\theta}(x) = \underbrace{\theta_0}_{\text{parameters}} + \underbrace{\theta_1 x}_{\text{parameters}}$$



모델 평가 – Cost function


Cost function

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m ((\underbrace{h_{\theta}(x^{(i)})}_{\text{가설}}) - \underbrace{y^{(i)}}_{\text{실제}}))^2 = \frac{1}{2m} \sum_{i=1}^m ((\hat{y}^{(i)}) - y^{(i)})^2$$

최적화 - Gradient Descent

Gradient Descent(경사하강법)

Hypothesis function의 최적의 parameter를 찾는 방법

$$h_{\theta}(x) = \boxed{\theta_0} + \boxed{\theta_1}x$$


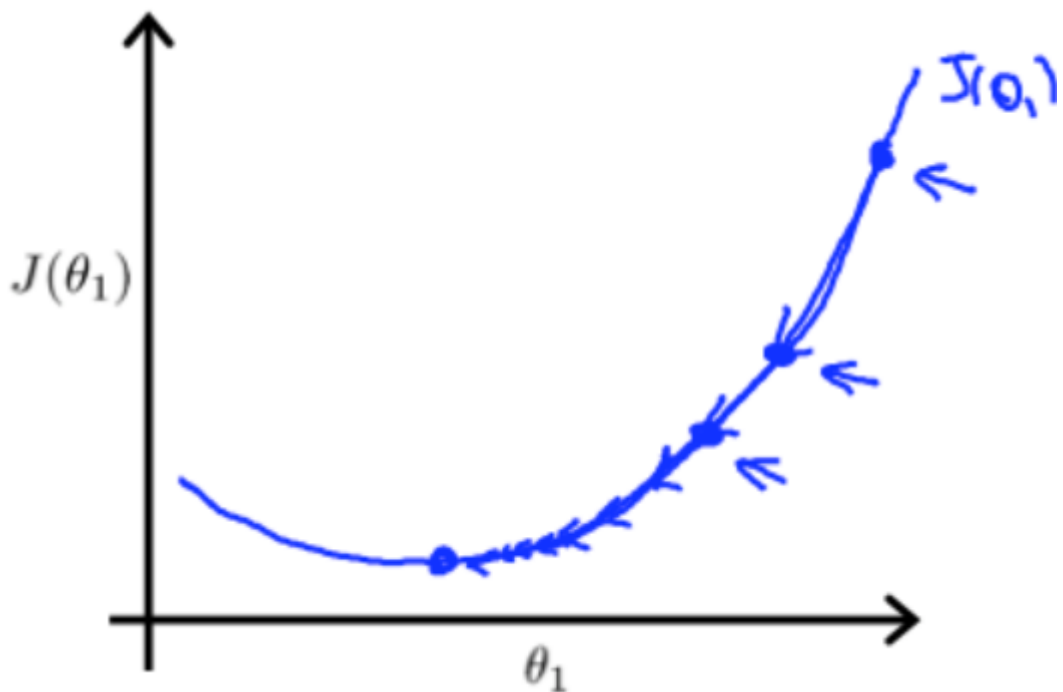
parameters

목표: Cost function에서 $J(\theta_0, \theta_1)$ 을 최소화

1. 어떤 parameter에서든 시작 가능
2. 계속 parameter를 변화해가면서 최소의 J를 찾는 것

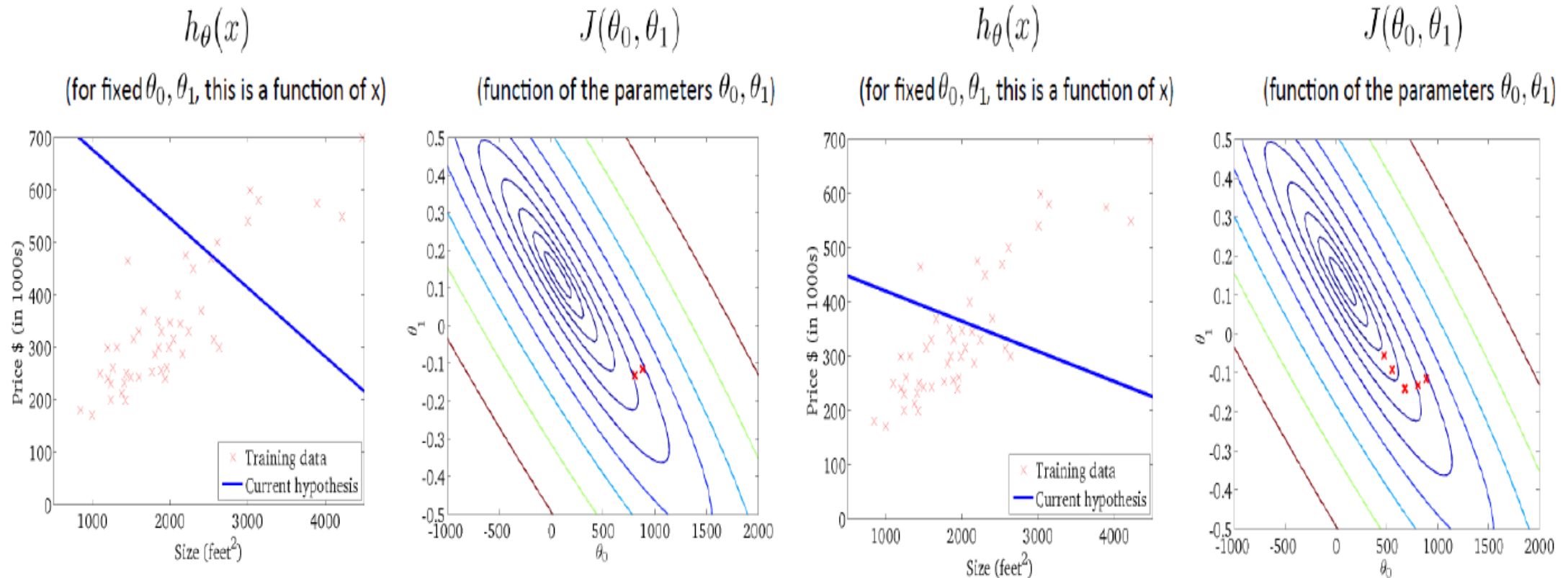
Gradient Descent

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$$

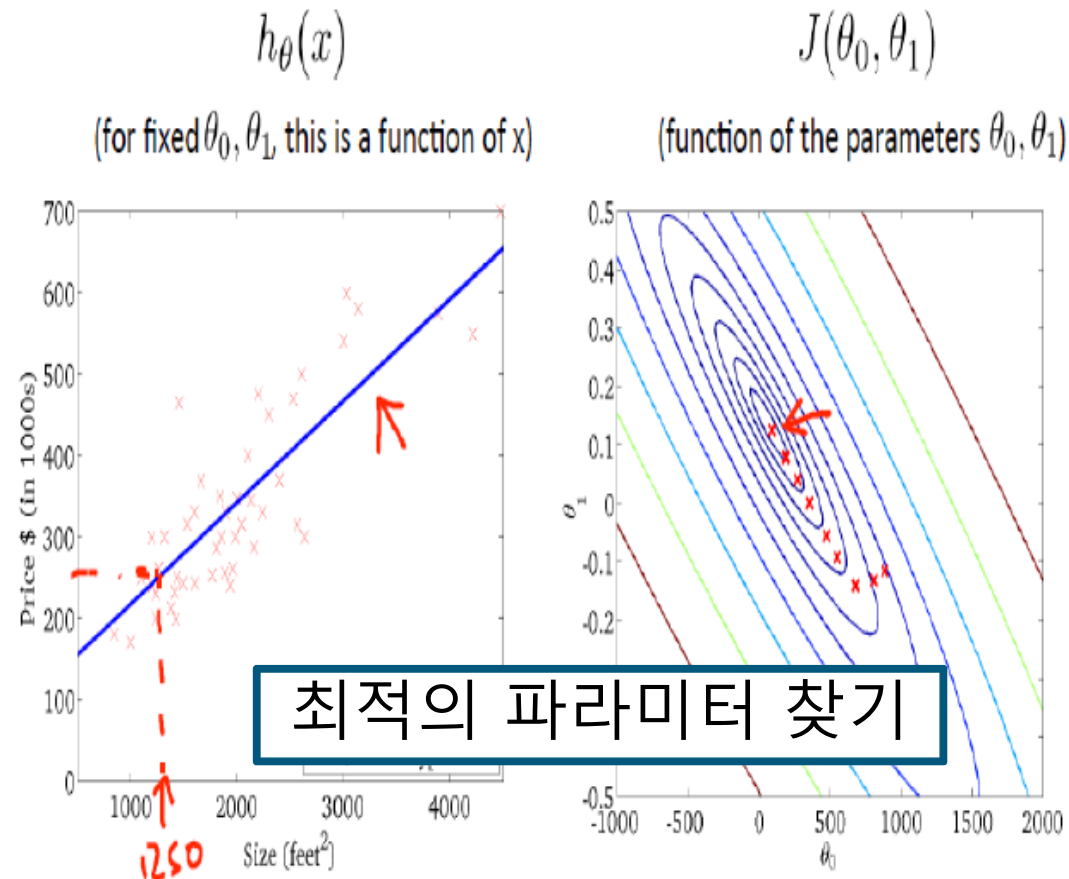
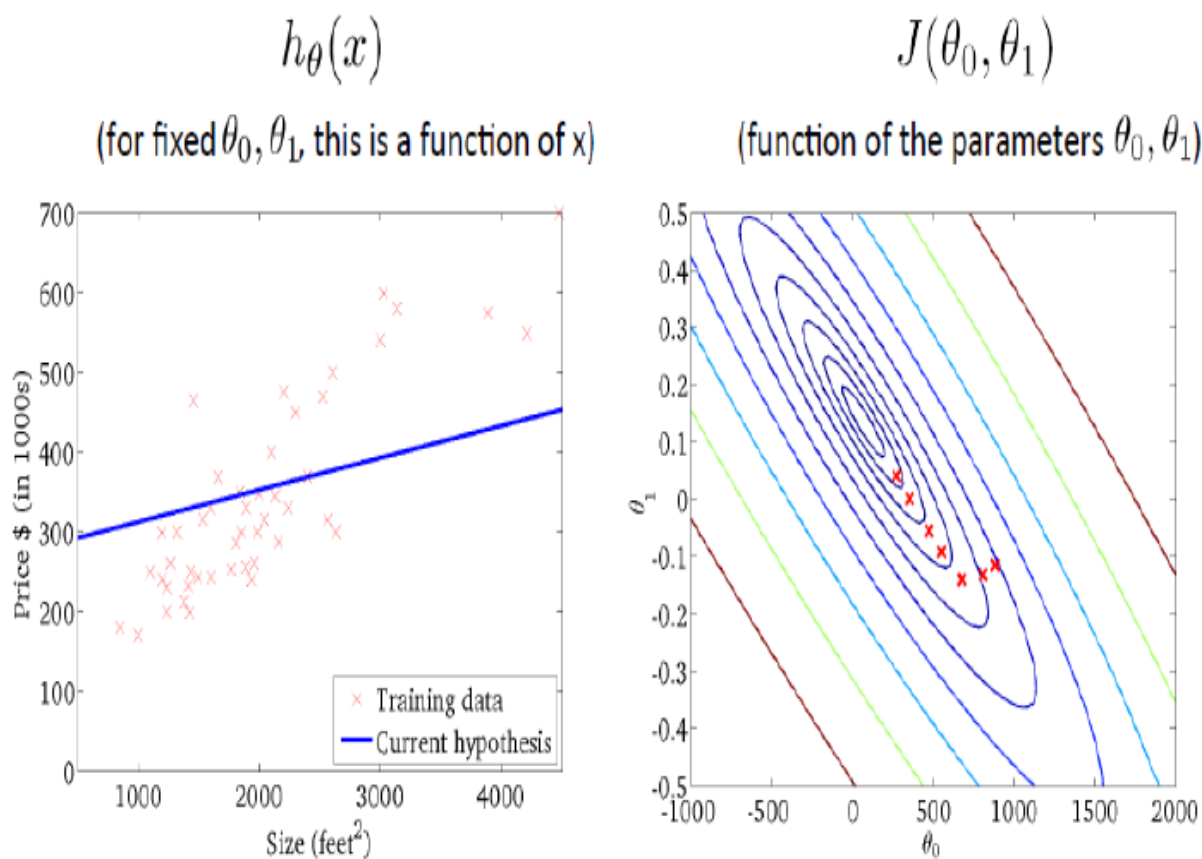


학습률(learning rate) = α 은 항상 양수

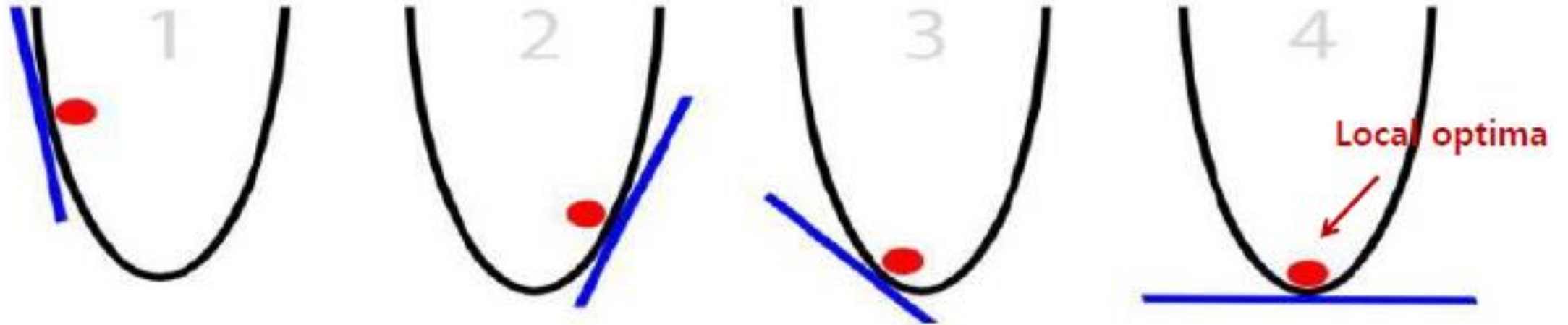
Gradient Descent



Gradient Descent

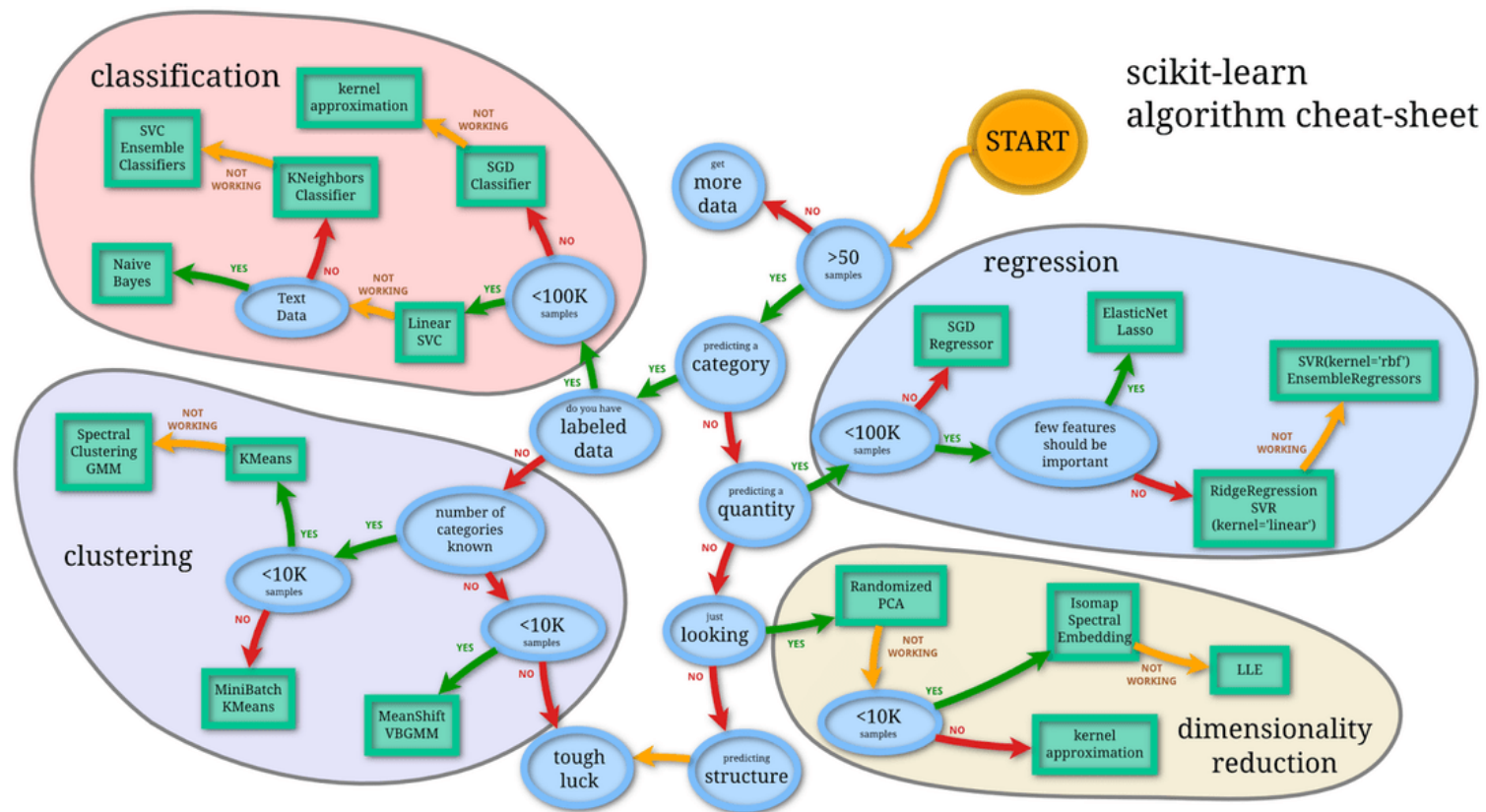


Gradient Descent



머신러닝 라이브러리

- Scikit-learn



Quest

- Scipy 라이브러리를 활용해 함수 최적화해보기!
job.csv 데이터로 모델 $Y=a*X1+b*X2+c$ 의 cost function을 minimize 해보세요! (파라미터 a, b, c의 최적값 찾기)

참고자료

- Coursera Machine Learning 강의
<https://www.coursera.org/learn/machine-learning/home/welcome>
- 오렐리앙 제롱, 핸즈온 머신러닝, 한빛 미디어
- <https://lukelab.tistory.com/10>
- <https://developers.google.com/machine-learning/crash-course/ml-intro?hl=ko>