Artificial intelligence

Linear Regression

Woohwan Jung (whjung@hanyang.ac.kr)
Fall 2022

HANYANG UNIVERSITY
Data Science Lab

Deadline for HW1 is extended

Features and Label

Features					Label
Position	**Experience**	**Skill**	**Country**	**City**	**Salary ($)**
Developer	0	1	USA	New York	103100
Developer	1	1	USA	New York	104900
Developer	2	1	USA	New York	106800
Developer	3	1	USA	New York	108700
Developer	4	1	USA	New York	110400
Developer	5	1	USA	New York	112300
Developer	6	1	USA	New York	114200
Developer	7	1	USA	New York	116100
Developer	8	1	USA	New York	117800
Developer	9	1	USA	New York	119700
Developer	10	1	USA	New York	121600

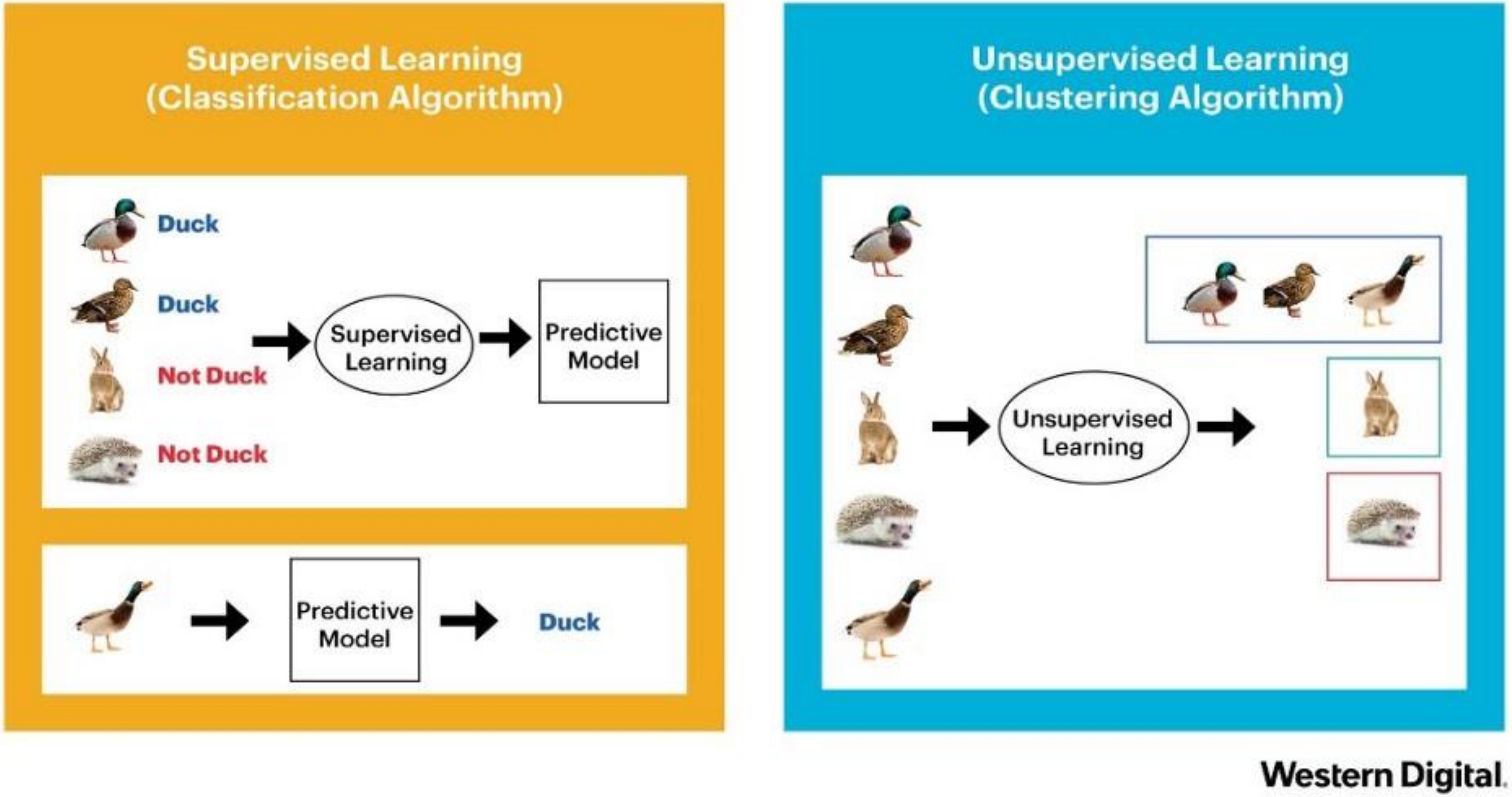
$x \rightarrow$ **Model** $\rightarrow \hat{y}$ $\qquad y$

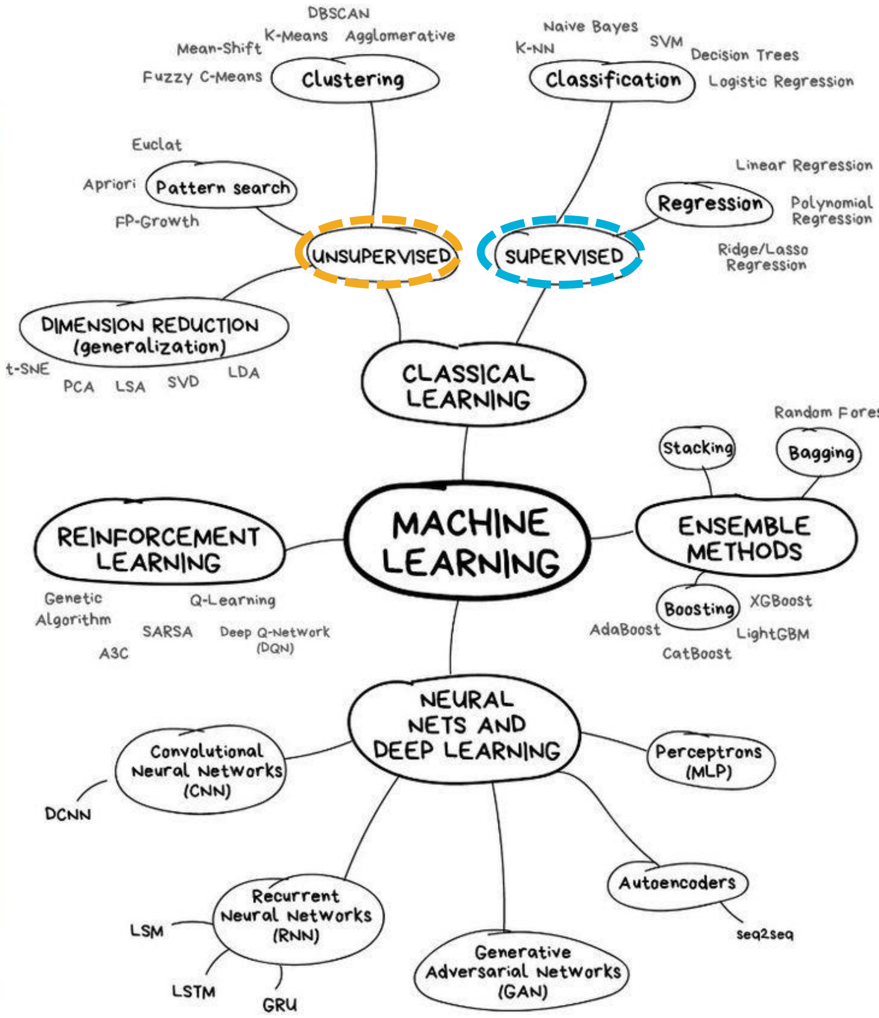
Input (features) — Output (prediction) — Label (Actual)

Training?

Building a model to make the model can **predict the labels** by using train data

https://bioinformaticsandme.tistory.com/118



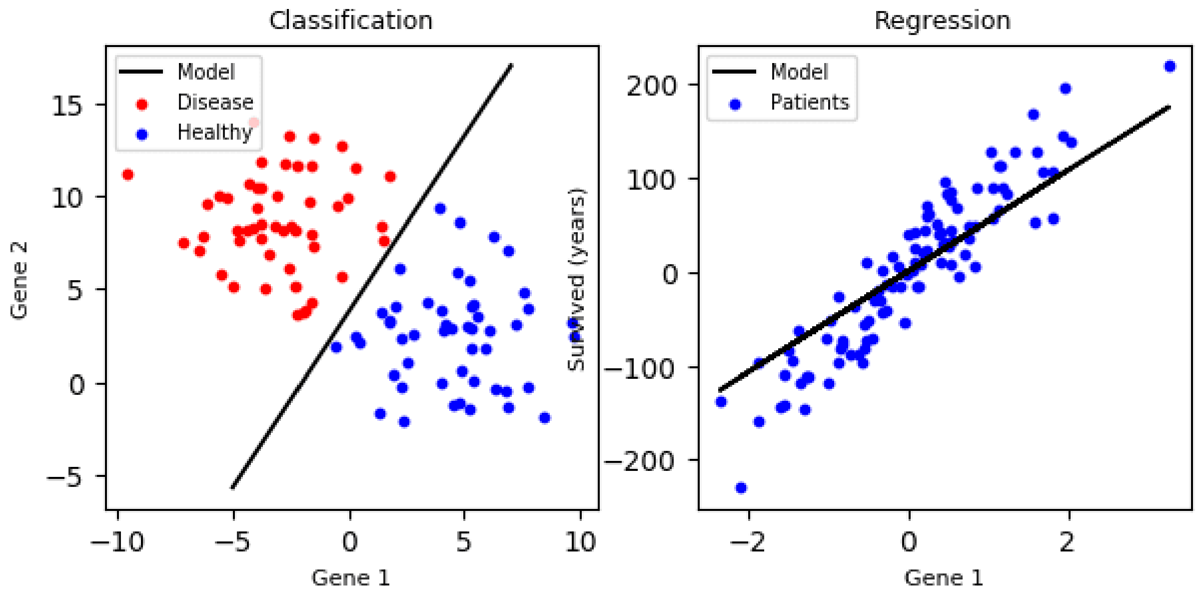


Supervised/Unsupervised 여부는 label (정답) 존재 여부로 구분!

Supervised learning : label 있음

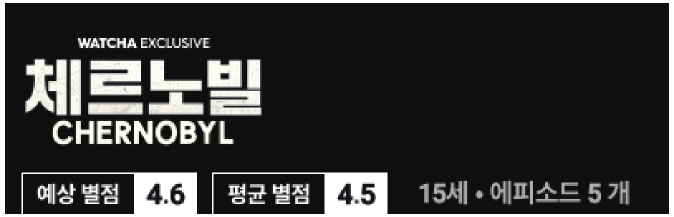
Unsupervised learning: label 없음

Classification vs Regression



Output Categorical value (Class) Numeric value

Q1. Classification? Regression?



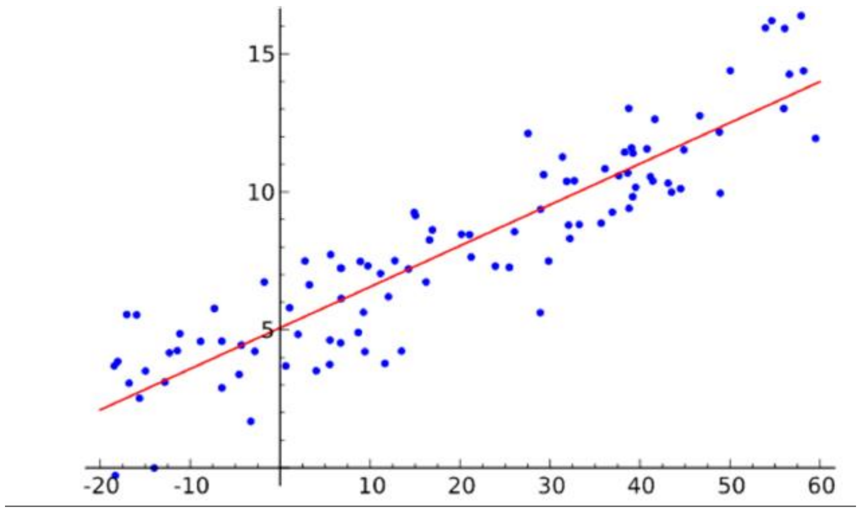
Q2. Classification? Regression?

Cat

Dog

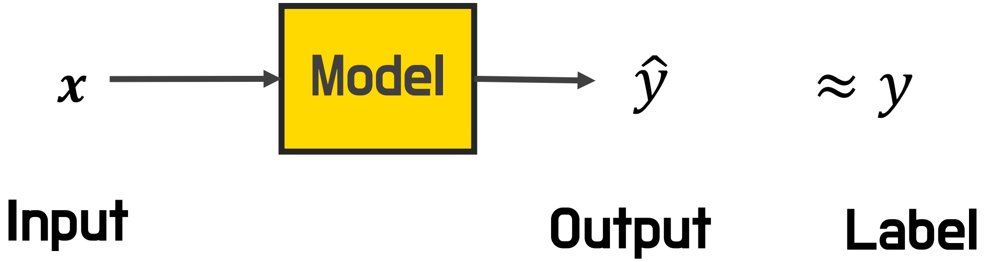
Linear Regression

- Modelling the linear relationship between a scalar response (label) and one or more explanatory variables (features)



Examples)
Area of house -> house price
# of iPhones sold -> Apple's sales

Linear Regression



- Input: $\boldsymbol{x} \in \mathbb{R}^n$
- Output: $\hat{y} = \boldsymbol{w}^\top \boldsymbol{x} + b$
  - Parameters: $\boldsymbol{w} \in \mathbb{R}^n$, $b \in \mathbb{R}$

Linear regression model의 학습이란?

데이터를 이용해 추정값 $\hat{y}$가 정답 y 와 유사해지도록 하는 모델 파라미터 $\boldsymbol{w}$와 b를 찾는 과정

Q1. 추정값과 정답의 유사도 측정?

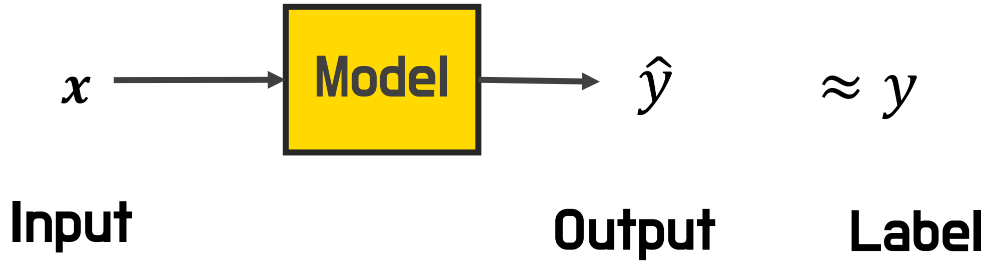
Loss function 으로 squared error 사용 $L(y, \hat{y}) = (y - \hat{y})^2$

Q2. 추정값과 정답의 유사도 측정?

Gradient descent!

Linear Regression

- Data $D = \{(\boldsymbol{x}_1, y_1), \dots (\boldsymbol{x}_n, y_n)\}$
- Model
  - Input: $\boldsymbol{x}_i \in \mathbb{R}^d$
  - Output $\hat{y} = \boldsymbol{w}^\top \boldsymbol{x} + b$
    - Parameters: $\boldsymbol{w} \in \mathbb{R}^d$, $b \in \mathbb{R}$



Linear regression model의 학습이란?

데이터를 이용해 추정값 $\hat{y}$가 정답 y 와 유사해지도록 하는 모델 파라미터 $\boldsymbol{w}$와 b를 찾는 과정

Training a linear regression model

Linear regression model의 학습이란?

데이터를 이용해 추정값 $\hat{y}$가 정답 y 와 유사해지도록 하는 모델 파라미터 $\boldsymbol{w}$와 b를 찾는 과정

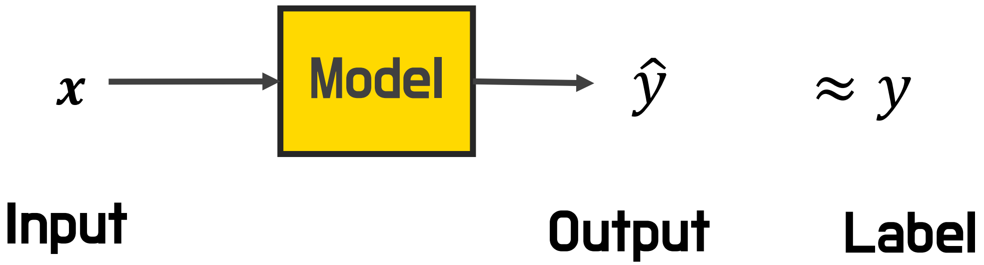
- Given
  - Training data $D = \{(\boldsymbol{x}_1, y_1), \dots (\boldsymbol{x}_n, y_n)\}$
- Our goal
  - Find $\boldsymbol{w}, b$ that minimizes $J(\boldsymbol{w}, b) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$

Q. 어떻게?

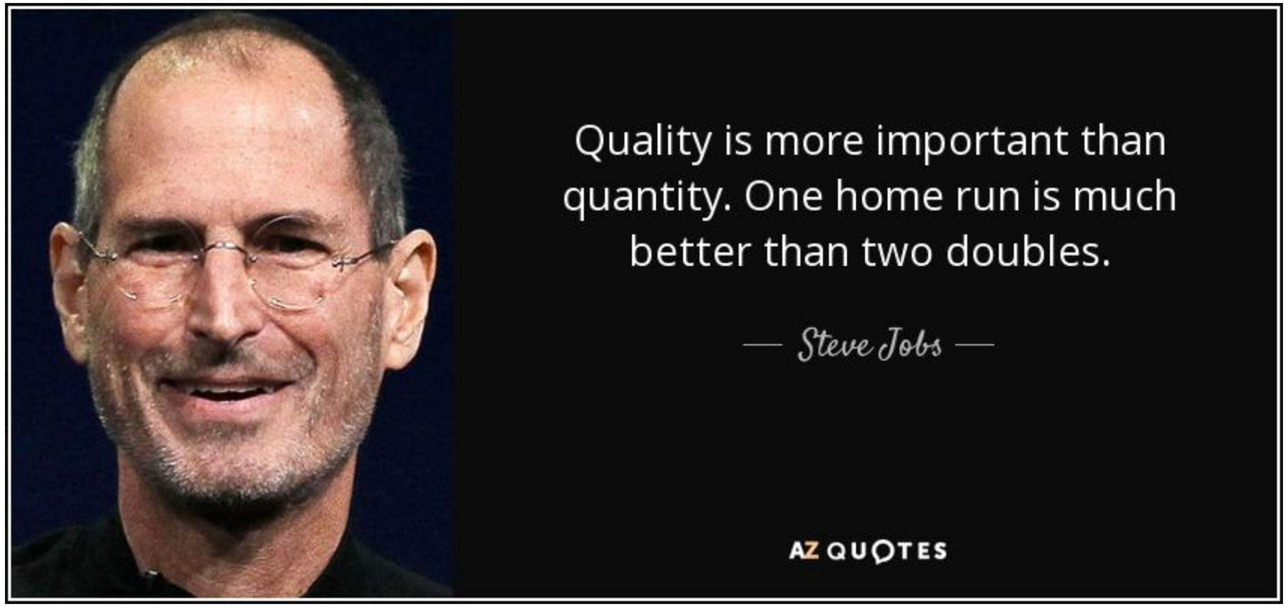
Gradient descent!

Interpretation of **w**

- Input: $\boldsymbol{x} \in \mathbb{R}^d$
- Output $\hat{y} = \boldsymbol{w}^\top \boldsymbol{x} + b$
  - Parameters: $\boldsymbol{w} \in \mathbb{R}^d$, $b \in \mathbb{R}$



Example1) 홈런의 가치



	안타 H	볼넷 BB	아웃 O	삼진아웃 K	홈런 HR	도루 SB		점수 R	
$\boldsymbol{x}_1$	12	7	27	7	3	1		11	y_1
$\boldsymbol{x}_2$	10	4	27	9	2	0		5	y_2

$$\hat{y} = \boldsymbol{w}^{\top}\boldsymbol{x} = w_H \cdot n_{\mathrm{H}} + w_{BB} \cdot n_{BB} + \ldots + w_{SB} \cdot n_{\mathrm{SB}} = \text{(Expected runs)}$$

w_e: expected runs of an event e

Run Value

w

Run Values : KBO 2005-2014

(완료되지 않은 이닝 제외)

out/base	05_14	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
uBB	0.334	0.330	0.307	0.299	0.347	0.336	0.372	0.323	0.349	0.319	0.350
HBP	0.366	0.362	0.325	0.391	0.399	0.406	0.267	0.394	0.336	0.350	0.461
IBB	0.035	0.246	0.081	0.271	-0.126	0.248	-0.054	-0.119	0.015	0.015	-0.082
1H	0.480	0.502	0.451	0.475	0.486	0.445	0.497	0.501	0.460	0.498	0.513
2H	0.820	0.785	0.808	0.838	0.816	0.882	0.829	0.864	0.757	0.822	0.825
3H	1.165	0.994	1.152	1.229	1.247	1.180	1.198	1.153	1.114	1.157	1.331
HR	1.464	1.463	1.423	1.442	1.477	1.450	1.473	1.459	1.451	1.456	1.465
all_out	-0.290	-0.281	-0.242	-0.267	-0.279	-0.304	-0.299	-0.282	-0.247	-0.279	-0.327

baseball-in-play.com

2루타 2개의 가치: (0.820+0.290)*2 = 2.220

홈런 1개의 가치: 1.464 + 0.290 = 1.754

Training a linear regression model

- Given
  - Training data $D = \{(\boldsymbol{x}^{(\mathbf{1})}, y^{(1)}), (\boldsymbol{x}^{(2)}, y^{(2)}), \ldots (\boldsymbol{x}^{(m)}, y^{(m)})\}$
- Our goal
  - Find $\boldsymbol{w}, b$ that minimizes $J(\boldsymbol{w}, b) = \frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)} - \hat{y}^{(i)}\right)^2$

Q. How?

Applicable methods: gradient descent, linear least squares, …

We are going to use gradient descent!

Gradient Descent
:Linear Regression

Optimization

$$\boldsymbol{x} \longrightarrow \boxed{\text{Model}} \longrightarrow \hat{y} \quad \approx y$$

Input Output Label

- Linear regression model
  - $\hat{y} = \boldsymbol{w}^\top \boldsymbol{x} + b$
- Loss function $L\left(\hat{y}^{(i)}, y^{(i)}\right) = \left(y^{(i)} - \hat{y}^{(i)}\right)^2$
- Cost function

$$J(\boldsymbol{w}, b) = \frac{1}{m}\sum_{i=1}^{m} L\left(\hat{y}^{(i)}, y^{(i)}\right) = \frac{1}{m}\sum_{i=1}^{m} \left(y^{(i)} - \hat{y}^{(i)}\right)^2$$

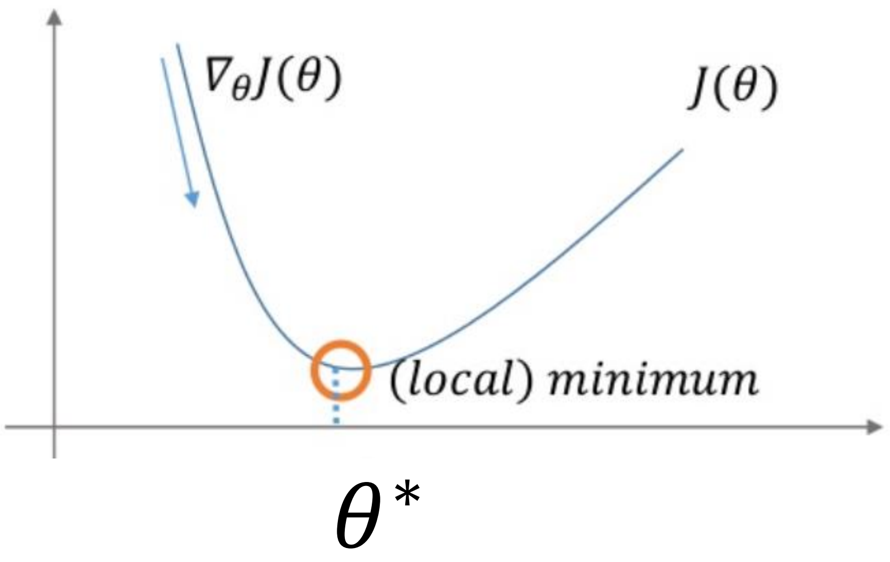
- Our goal
  - Find parameters $\boldsymbol{w} \in \mathbb{R}^n$, $b \in \mathbb{R}$ that minimize $J(\boldsymbol{w}, b)$
- Gradient Descent!

Gradient Descent

- Algorithm to minimize a cost function $J(\theta)$
  - $J(\boldsymbol{\theta})$: cost function
  - $\boldsymbol{\theta}$: model parameters
- Repeatedly update

$$\boxed{\boldsymbol{\theta} = \boldsymbol{\theta} - \eta \cdot \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})}$$

  - η: Learning rate



$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \left[\frac{\partial J(\boldsymbol{\theta})}{\partial \theta_1} \quad \frac{\partial J(\boldsymbol{\theta})}{\partial \theta_2} \quad \ldots \quad \frac{\partial J(\boldsymbol{\theta})}{\partial \theta_k}\right]^T \text{where } \boldsymbol{\theta} = [\theta_1 \quad \theta_2 \ldots \quad \theta_k]^T$$

By Hakky St

Gradient of L

- Output $\hat{y} = \boldsymbol{w}^\top \boldsymbol{x} + b$
- Loss function $L(\hat{y}, y) = (y - \hat{y})^2$

$$\frac{\partial L}{\partial w_k} = -2(y - \hat{y})x_k \qquad \frac{\partial L}{\partial b} = -2(y - \hat{y})$$

$$L(a, y) = (y - a)^2$$

Linear Regression Recap

y

w_1

w_2

x_1

x_2

b

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

$$L(a, y)$$

$$\frac{\partial a}{\partial w_i}$$

$$\frac{dL}{da}$$

$$\frac{\partial L}{\partial w_i} = \frac{\partial L}{\partial a} \frac{\partial a}{\partial w_i} = -2(y - a) x_i$$

$$L(a, y) = (y - a)^2$$

Linear Regression Recap

y

w_1

w_2

x_1

x_2

b

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

$$L(a, y)$$

$$\frac{da}{dw_i}$$

$$\frac{dL}{da}$$

$$\frac{\partial L}{\partial w_i} = \frac{\partial L}{\partial a}\frac{\partial a}{\partial w_i} = -2(y - a)x_i$$

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial a}\frac{\partial a}{\partial b} = -2(y - a)$$

Gradient descent on m examples

$$J(\boldsymbol{w}, b) = \frac{1}{m}\sum_{i=1}^{m} L\left(\hat{y}^{(i)}, y^{(i)}\right)$$

$$\frac{\partial}{\partial w_k} J(\boldsymbol{w}, b) = \frac{1}{m}\sum_{i=1}^{m} \frac{\partial}{\partial w_k} L\left(\hat{y}^{(i)}, y^{(i)}\right) = \frac{2}{m}\sum_{i=1}^{m}\left(\hat{y}^{(i)} - y\right) x_k$$

$$\frac{\partial}{\partial b} J(\boldsymbol{w}, b) = \frac{1}{m}\sum_{i=1}^{m} \frac{\partial}{\partial b} L\left(\hat{y}^{(i)}, y^{(i)}\right) = \frac{2}{m}\sum_{i=1}^{m}\left(\hat{y}^{(i)} - y\right)$$

Gradient descent

$$\boldsymbol{\theta} = \boldsymbol{\theta} \text{-}\ \eta \cdot \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$$

Gradient descent for training a Linear Regression model (n = 2)

- Randomly Initialize $\boldsymbol{w}, b$
- lr = 0.1
- For e = 1 to n_{epoch}:
  - d_w1 = 0; d_w2 = 0; d_b=0
  - For i= 1 to m:
    - $a = w_1 x_1^{(i)} + w_2 x_2^{(i)} + b$
    - d_w1 $+= 2(a-y)x_1^{(i)}$
    - d_w2 $+= 2(a-y)x_2^{(i)}$
    - d_b $+= 2(a-y)$
  - $w_1 -= lr \ * d\_w1$/m
  - $w_2 -= lr \ * d\_w2$/m
  - $b -= lr \ * d\_b$/m

Gradient descent

$$\boldsymbol{\theta} = \boldsymbol{\theta} - \eta \cdot \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$$

$$\frac{\partial}{\partial w_k} J(\boldsymbol{w}, b) = \frac{2}{m} \sum_{i=1}^{m} (a-y) x_k$$

$$\frac{\partial}{\partial b} J(\boldsymbol{w}, b) = \frac{2}{m} \sum_{i=1}^{m} (a-y)$$