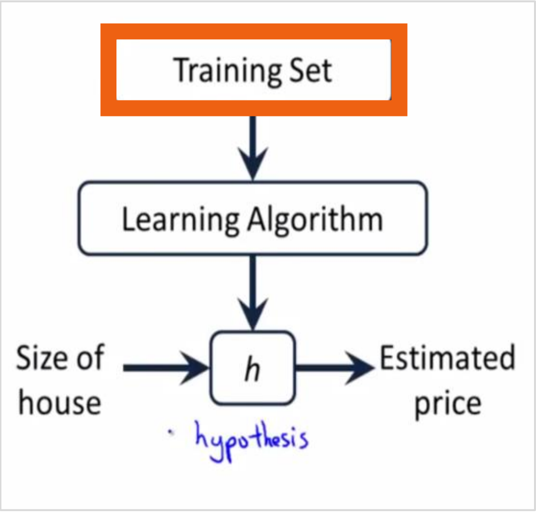
Machine Learning

ML(Machine Learning)? Why

The Limitation of explicit Programming

```
if  condition1
  else if  condition2
    else if   condition3
       else if  condition4
       .
   9990 Lines later …
       .
       .
                        else if condition 9994
                          else if  condition 9995
       .
```



Machine “Learning”

“Training” set

Size (feet)^2	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

Do it yourself!

Machine Learning(ML)

컴퓨터가 배워야 할 4가지 자료유형

- **Supervised** : Linear Regression, Logistic Regression- Classification,
  multi-level Classification : Neural Network (NN), Support Vector Machine(SVM)
- **Unsupervised** : Clustering, Recommendation System, NLP
- Reinforcement Learning : Q-Learning - Gaming, Automotive Mobility
- Semi-supervised : end-to-end self training

<u>Deep</u> Learning:

CNN, GAN, RNN, LSTM, Transformer

<u>challenging</u> : Overfitting, Dimensionality (**Data Compression**), Security & Privacy (**On-device**), Ethic

Supervised Learning = prediction

continuous -> linear regression

discrete -> logistic regression, multi-classification, NN

Unsupervised Learning = clustering

Cocktail party Algorithm

YouTube algorithm

Recommendation System

Schedule : ca. 3개월 (**Supervised** , **Unsupervised**)

3주 간격	진도	내용(syllabus 참조)
26.9.2020	**Week 1,2**	**Linear Regression**
17.10.2020	3,4,5	Logistic Regression, NN
7.11.2020	6,7	Supervised 내용 끝- 총 정리
28.11.2020	8, 9.5	Unsupervised 시작
19.12.2020	9.5, 10, 11	Recommendation system Unsupervised 총정리

Notation

training set

$$(x^{(i)}, y^{(i)})$$

n = Number of "x input variable" **(j = 1 ~ 4)**

m = Number of training examples (i = 1 ~ 4)

x's = "input" variable / features

y's = "output" variable / "target" variable / label

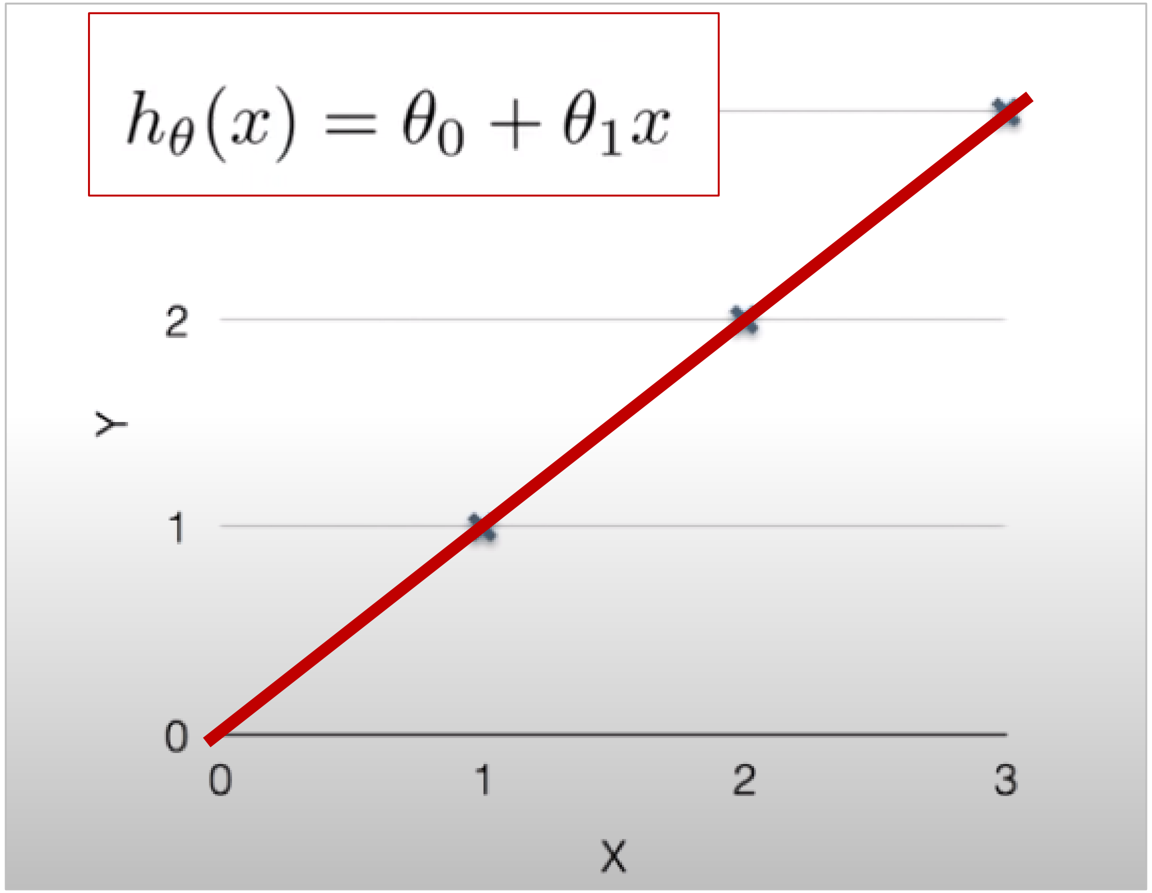
	Age of home (years)	Price ($1000)
$(x^{(1)}, y^{(1)})$	45	460
	40	232
	30	315
$(x^{(4)}, y^{(4)})$	36	178

$$(x_j^{(i)}, y^{(i)})$$

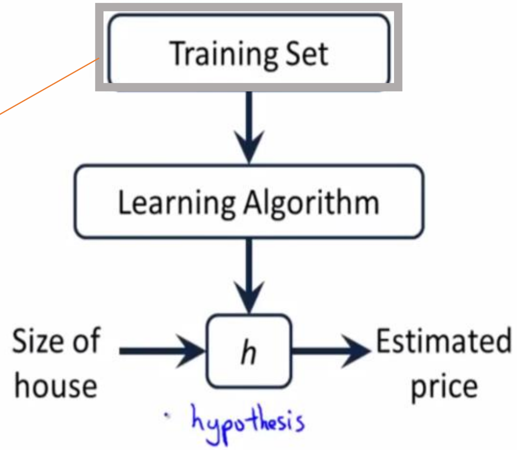
1	2	3	4	y
Size $(\text{feet})^2$	Number of bedrooms	Number of floors	Age of home (years)	Price ($1000)
2104	5	1	45 $\boxed{x_4^{(1)}}$	460
1416	3	2	40	232
1534	3 $\boxed{x_2^{(3)}}$	2	30	315
852 $\boxed{x_1^{(4)}}$	2	1	36	178

Hypothesis

$$h_\theta(x) = \theta_0 + \theta_1 x$$



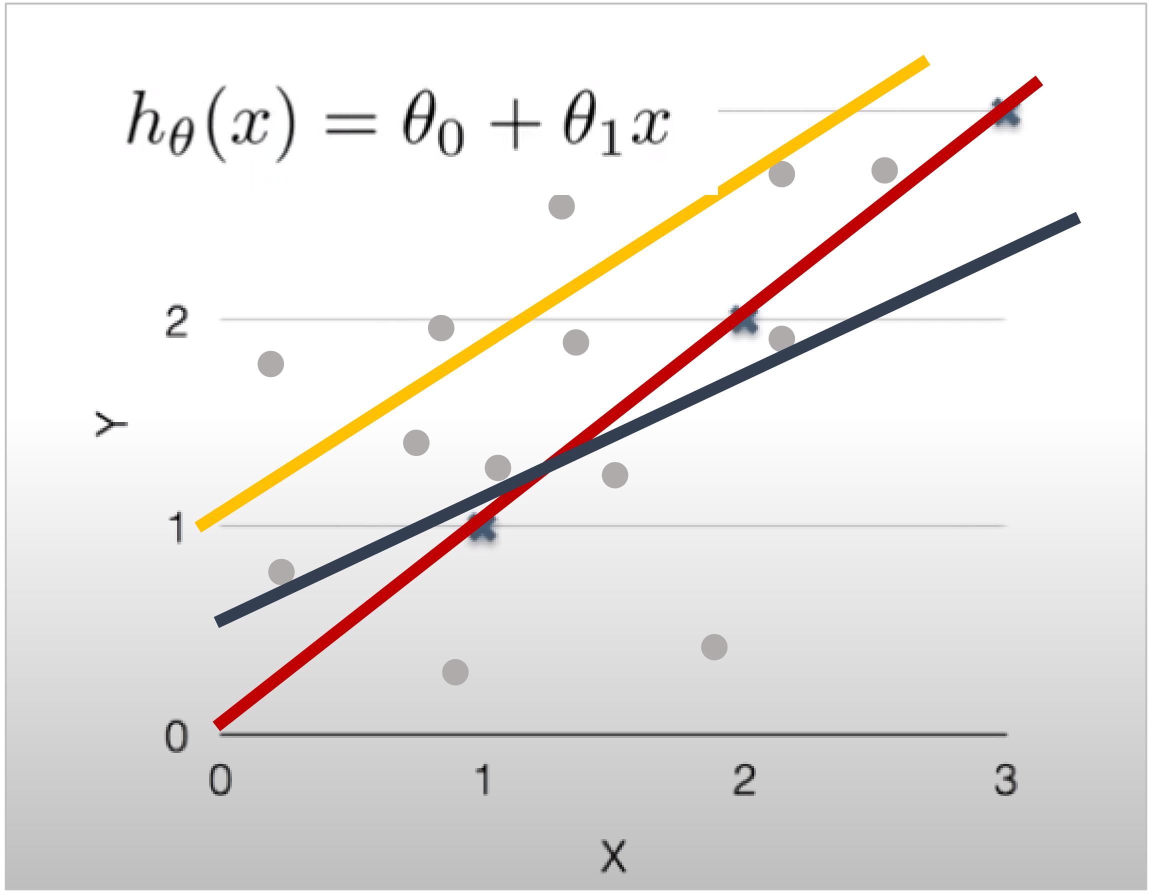
x	Y
1	1
2	2
3	3

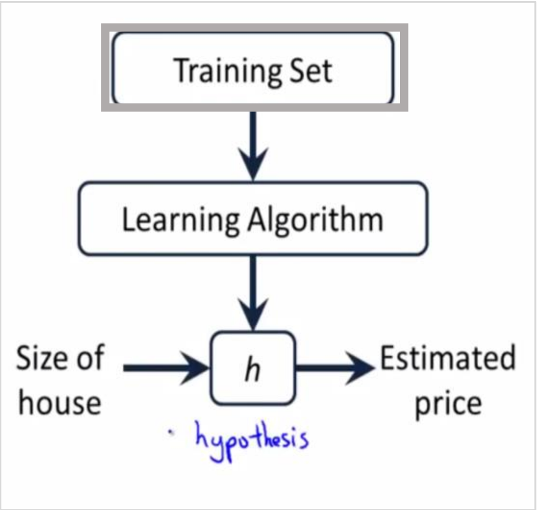


$$h_\theta(x) = \theta^T x = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

Which is the best …?

hypothesis의 parameter를 결정해야 하는 상황 발생

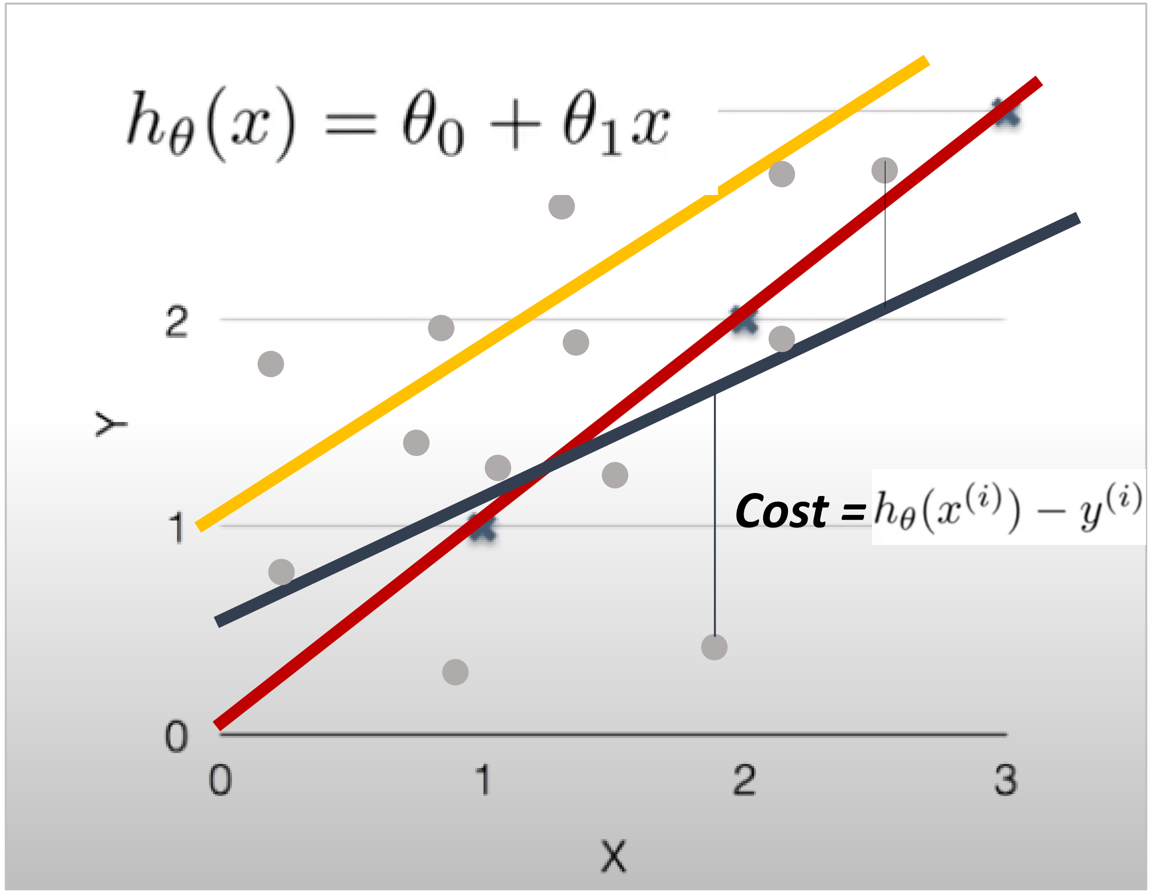




Parameters: θ_0, θ_1

y절편, 기울기

Solution: ***Cost function*** *minimization*



$$\underset{\theta_0,\theta_1}{\text{minimize}}\ J(\theta_0, \theta_1)$$

$$\boxed{J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2}$$

One variable(one input x variable)

Hypothesis: $h_\theta(x) = \theta_0 + \theta_1 x$

Parameters: θ_0, θ_1

Cost Function: $J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$

Goal: $\underset{\theta_0, \theta_1}{\text{minimize}}\ J(\theta_0, \theta_1)$

Multiple variable(multiple input x variable)

Hypothesis: $h_\theta(x) = \theta^T x = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$

Parameters: $\theta_0, \theta_1, \ldots, \theta_n$

Cost function:

$$J(\theta_0, \theta_1, \ldots, \theta_n) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

parameter decision: Gradient Descent

one variable(one input x variable)

(n=1):

Repeat {

$$\theta_0 := \theta_0 - \alpha \underbrace{\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})}_{\frac{\partial}{\partial\theta_0}J(\theta)}$$

$$\theta_1 := \theta_1 - \alpha \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x^{(i)}$$

(simultaneously update θ_0, θ_1)

}

Multiple variable(multiple input x variable)

$(n \geq 1)$:

Repeat {

$$\theta_j := \theta_j - \alpha \boxed{\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}}$$

$\frac{\partial}{\partial\theta_j}J(\theta)$

(simultaneously update θ_j for $j = 0, \ldots, n$)

}

$$\theta_0 := \theta_0 - \alpha \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_0^{(i)}$$

$$\theta_1 := \theta_1 - \alpha \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_1^{(i)}$$

$$\theta_2 := \theta_2 - \alpha \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_2^{(i)}$$

...

Andrew Ng

Gradient Descent

Feature scaling *(*feature = input x variable*)*

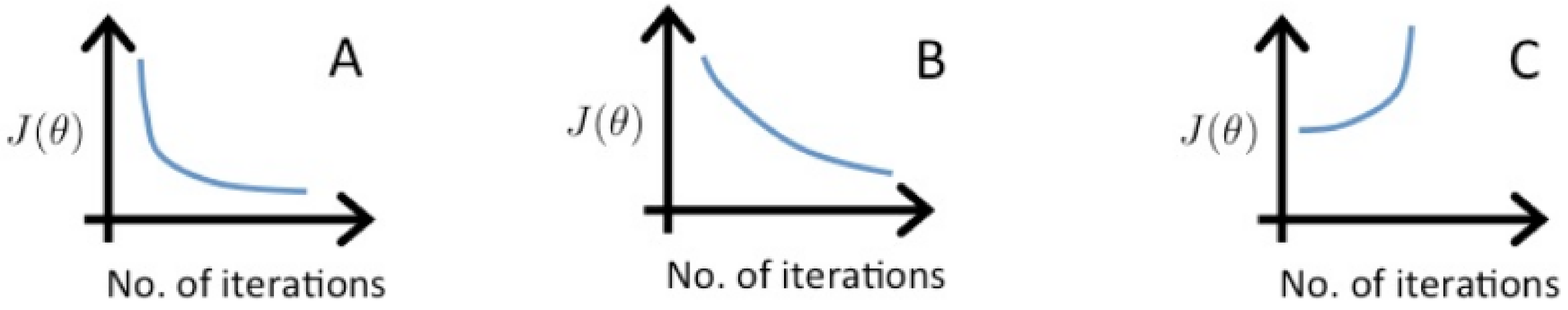
To speed up gradient descent

Mean normalization (u = avg)

$$x_i := \frac{x_i - \mu_i}{s_i}$$

*Gradient Descent- **Learning Rate***

Suppose a friend ran gradient descent three times, with $\alpha = 0.01$, $\alpha = 0.1$, and $\alpha = 1$, and got the following three plots (labeled A, B, and C):



Which plots corresponds to which values of α?

- ○ A is $\alpha = 0.01$, B is $\alpha = 0.1$, C is $\alpha = 1$.
- ○ A is $\alpha = 0.1$, B is $\alpha = 0.01$, C is $\alpha = 1$.
- ○ A is $\alpha = 1$, B is $\alpha = 0.01$, C is $\alpha = 0.1$.
- ○ A is $\alpha = 1$, B is $\alpha = 0.1$, C is $\alpha = 0.01$.

If α is too small: slow convergence.
If α is too large: $J(\theta)$ may not decrease on every iteration; may not converge.

Wrap up : Summary

Why do we need to study ML? -> The Limitation of explicit Programming

Supervised Learning = prediction

Unsupervised Learning = clustering

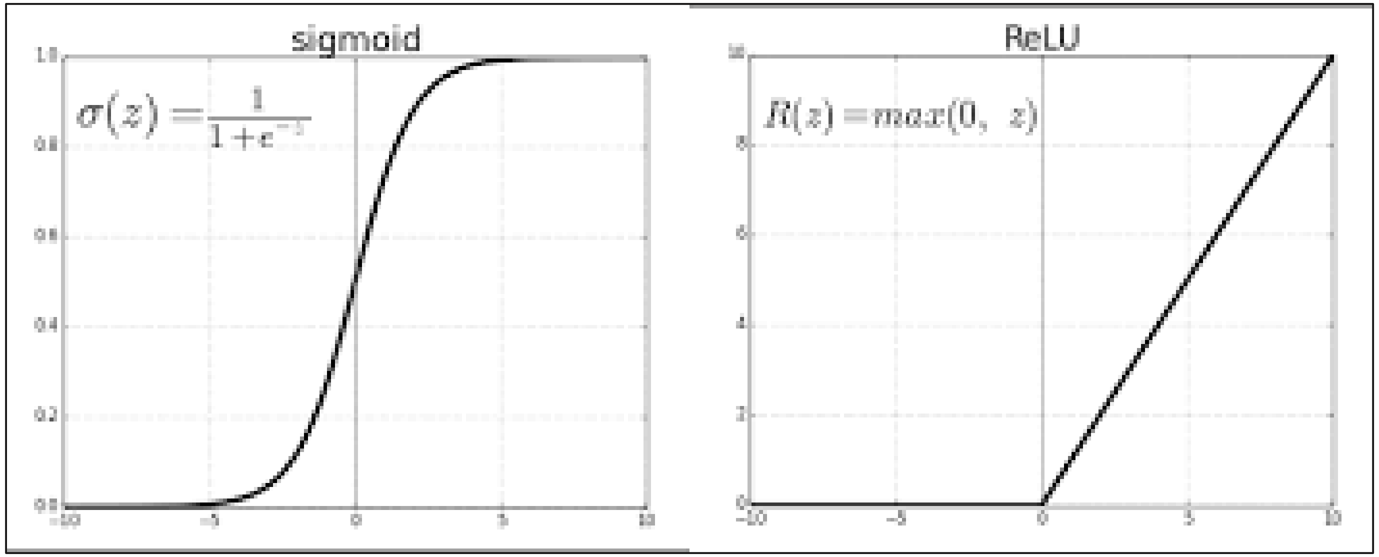
Linear Regression

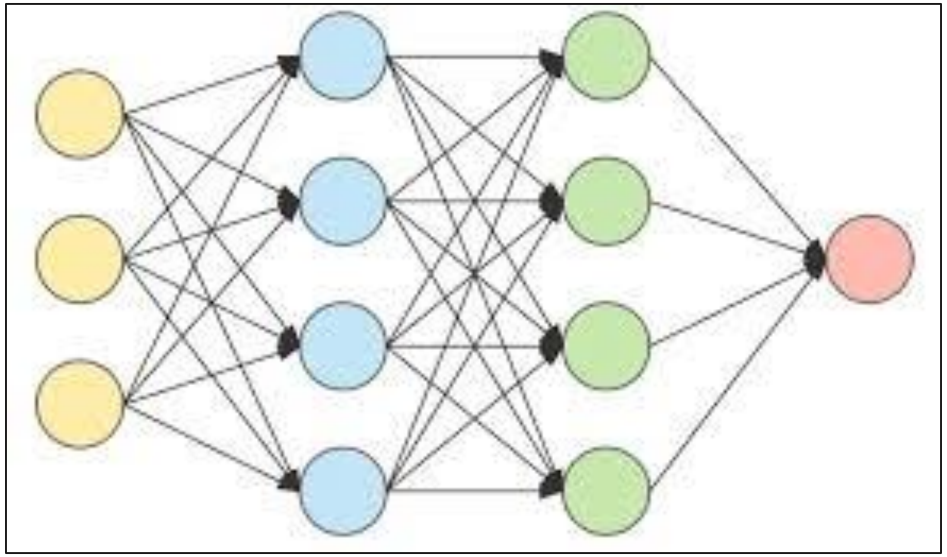
$\boxed{(x_j^{(i)}, y^{(i)})}$ *training set notation*

- *Training of many dataset (Experience accumulation) ->* ***hypothesis***
- *Which Hypothesis is better? ->* ***cost function***
- *How can I get minimum cost function? ->* ***parameter*** *decision->* ***Gradient Descent***
- *Feature scaling : to speedup -> mean(averrage) normalization*
- *Learning rate* If α is too small: slow convergence.
  If α is too large: $J(\theta)$ may not decrease on every iteration; may not converge.

Wrap up : Spoiler 😎

Logistic Regression, Neural Network(NN)





@ https://hay.dk/en/hay/accessories-8b1762fc/decoration/desk-accessories-b2b/time

감사합니다

Danke

ありがとう

Thank you