ML in Python

Learn Machine Learning with Microsoft Azure ML Studio

Microsoft Student Partner
백진헌

CNTK, TensorFlow, Keras, ···

왜 프레임워크를 사용하지 않나요?

ML in Python

1. Linear Regression
(Boston 집값 예측, Boston housing price data set)

2. Logistic Regression
(필기체 숫자 학습 분류 모델, MNIST data set)

3. Neural-network
(필기체 숫자 학습 Deep-learning 모델, MNIST data set)

ML in Python

1. Linear Regression
(Boston 집값 예측, Boston housing price data set)

2. Logistic Regression
(필기체 숫자 학습 분류 모델, MNIST data set)

3. Neural-network
(필기체 숫자 학습 Deep-learning 모델, MNIST data set)

Linear Regression

$$Y = (w1*x1) + (w2*x2) + \cdots + (wn*xn)$$

Linear Regression

$$Y = (w1*x1) + (w2*x2) + \cdots + (wn*xn) + b$$

x1, x2, ···, xn : Y를 예측할 때 필요한 유의미한 피처
w1, w2, ···, wn, b : Y를 예측할 때 사용되는 각 피처에 대한 무게 값
Y : 학습된 모델을 통해 예측한 값

Linear Regression

$$Y = (w1*x1) + (w2*x2) + \cdots + (wn*xn) + b$$

그럼 우리가 학습을 통해 계산해야 하는 것은?

Linear Regression

$$Y_{(predict)} = (w1*x1) + (w2*x2) + \cdots + (wn*xn) + b$$

Linear Regression

$$Y_{(predict)} = (w1*x1) + (w2*x2) + \cdots + (wn*xn) + b$$

$$(Y - Y_{(predict)}) \wedge 2$$

Linear Regression

$$Y_{(predict)} = (w1*x1) + (w2*x2) + \cdots + (wn*xn) + b$$

$$\text{Cost Function} : (Y - Y_{(predict)}) \text{ ^ } 2$$

Linear Regression

$$Y_{(predict)} = (w1*x1) + (w2*x2) + \cdots + (wn*xn) + b$$

Cost Function – Low : 더 적게 학습

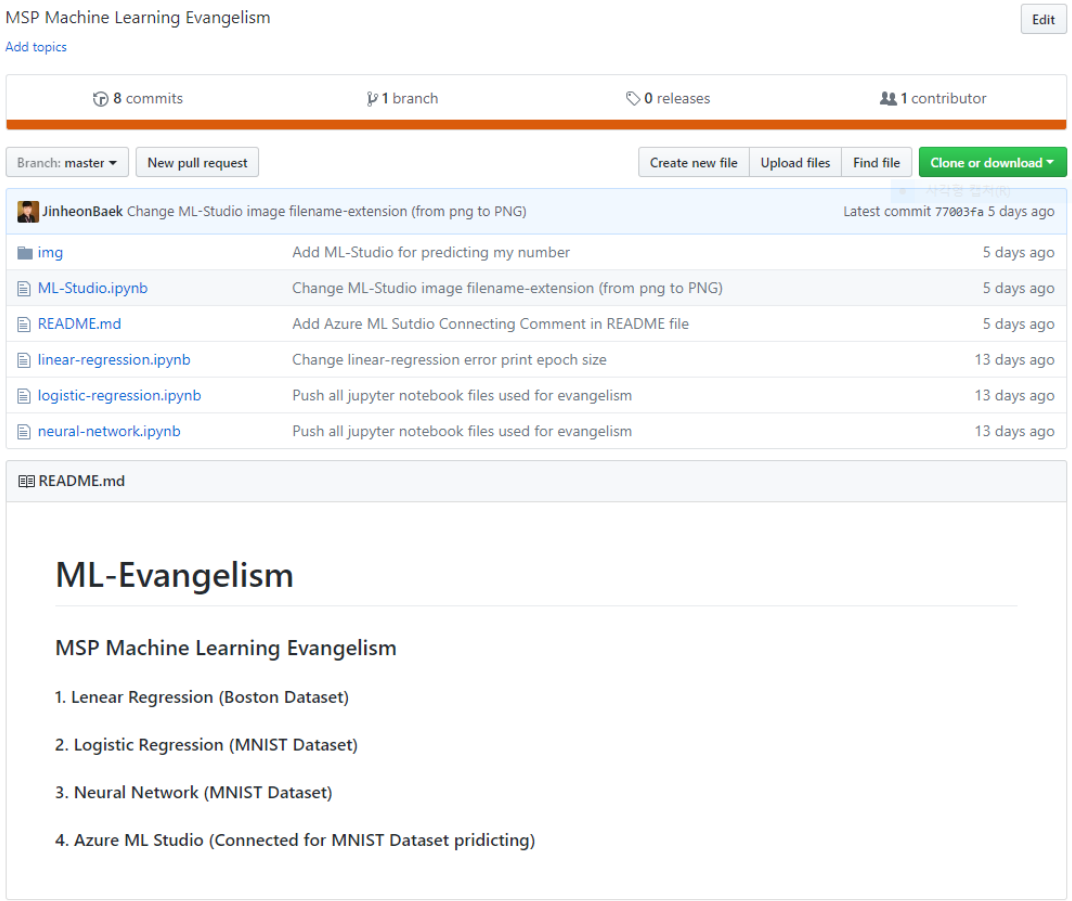
Cost Function – High : 더 많이 학습

Linear Regression

우리가 도움말을 보지 않고 스마트폰을 사용하듯이,
Machine Leaning 잘 모르겠지만 일단 해보자.

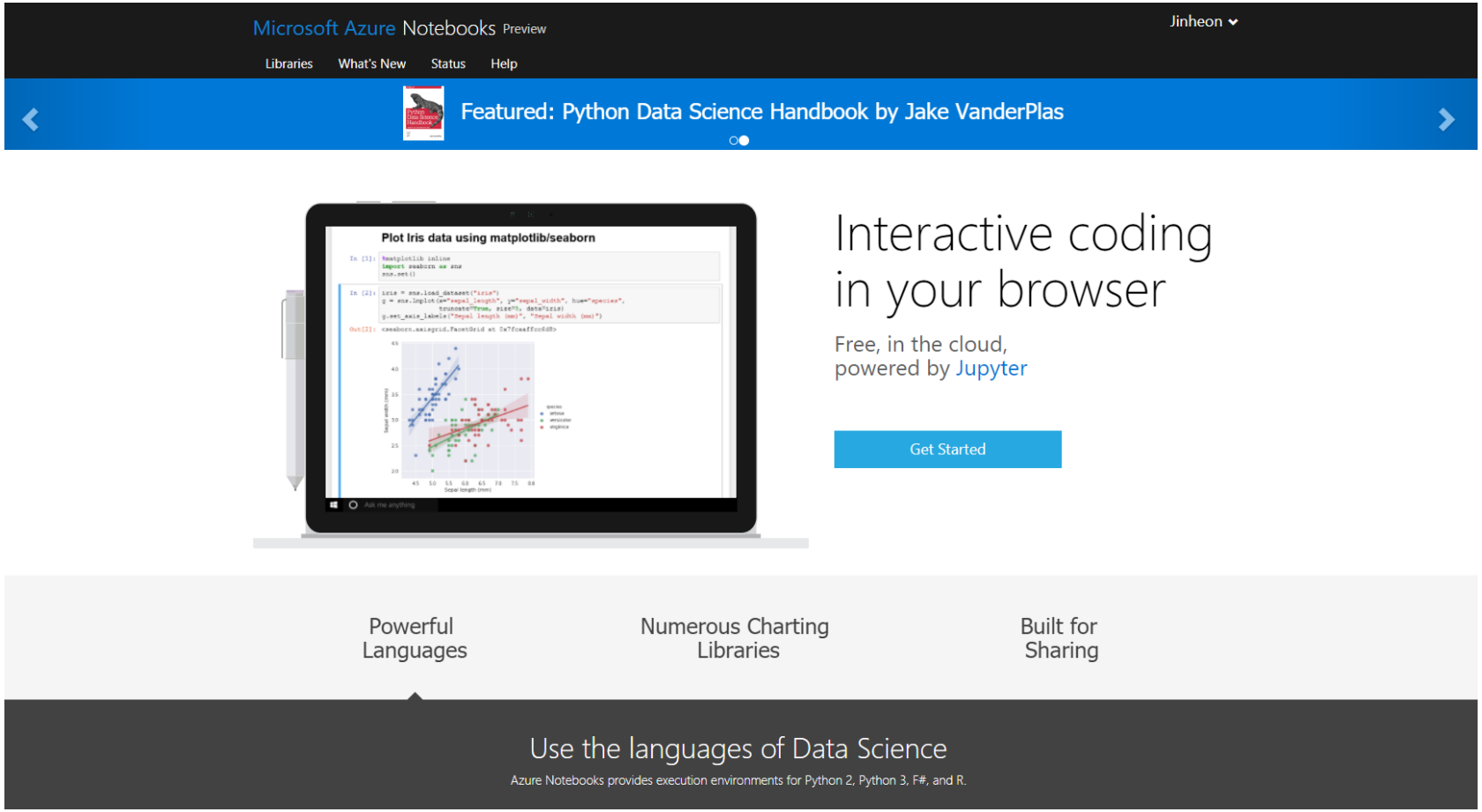
Linear Regression

https://github.com/JinheonBaek/ML-Evangelism



Linear Regression

https://notebooks.azure.com/



Linear Regression

1. Login
2. Libraries 메뉴
3. New Library
4. From Github
5. JinheonBaek/ML-Evangelism (GitHub repository)
6. Import
7. Linear-regression.ipynb

다 함께 해봅시다.

Linear Regression 설명

ML in Python

1. Linear Regression
(Boston 집값 예측, Boston housing price data set)

2. Logistic Regression
(필기체 숫자 학습 분류 모델, MNIST data set)

3. Neural-network
(필기체 숫자 학습 Deep-learning 모델, MNIST data set)

Logistic Regression

$$Y = (w1*x1) + (w2*x2) + \cdots + (wn*xn)$$

Logistic Regression

개 : 0 / 고양이 : 1

$$Y = (w1*x1) + (w2*x2) + \cdots + (wn*xn)$$

Logistic Regression

개 : 0 / 고양이 : 1

$$Y = (w1*x1) + (w2*x2) + \cdots + (wn*xn)$$

Y 〉 0 : True

Y 〈 0 : False

Logistic Regression

Cost Function ???

Logistic Regression

개 : 0 / 고양이 : 1

$$Y = (w1*x1) + (w2*x2) + \cdots + (wn*xn)$$

Sigmoid(Y) 〉 0.5 : True

Sigmoid(Y) 〈 0.5 : False

Logistic Regression

개 : 0 / 고양이 : 1

$$Y = (w1*x1) + (w2*x2) + \cdots + (wn*xn)$$

Sigmoid(Y) 〉 0.5 : True
Sigmoid(Y) 〈 0.5 : False

Cost Function : 1(고양이) – Sigmoid(Y) (이런느낌)

Logistic Regression

$$Y1 = (w11*x1) + (w12*x2) + \cdots + (w1n*xn)$$
$$Y2 = (w21*x1) + (w22*x2) + \cdots + (w2n*xn)$$
$$\cdots$$
$$Yn = (wn1*x1) + (wn2*x2) + \cdots + (wnn*xn)$$

Logistic Regression

$$Y1 = (w11*x1) + (w12*x2) + \cdots + (w1n*xn)$$
$$Y2 = (w21*x1) + (w22*x2) + \cdots + (w2n*xn)$$
$$\cdots$$
$$Yn = (wn1*x1) + (wn2*x2) + \cdots + (wnn*xn)$$

$$Max\ (\ y_{(1 \sim n)}\ ) = y3\ ?$$

Logistic Regression

$$Y1 = (w11*x1) + (w12*x2) + \cdots + (w1n*xn)$$
$$Y2 = (w21*x1) + (w22*x2) + \cdots + (w2n*xn)$$
$$\cdots$$
$$Yn = (wn1*x1) + (wn2*x2) + \cdots + (wnn*xn)$$

$$Max\ (\ Sigmoid(y_{(1 \sim n)})\ ) = y3\ ?$$

MNIST data set ?

Logistic Regression

28 x 28, 필기체 숫자 (0 ~ 9)

Logistic Regression

$$Y0 = (w11*x1) + (w12*x2) + \cdots + (w1n*xn)$$
$$Y1 = (w21*x1) + (w22*x2) + \cdots + (w2n*xn)$$
$$\cdots$$
$$Y9 = (wn1*x1) + (wn2*x2) + \cdots + (wnn*xn)$$

다 함께 해봅시다.

Logistic Regression 설명

ML in Python

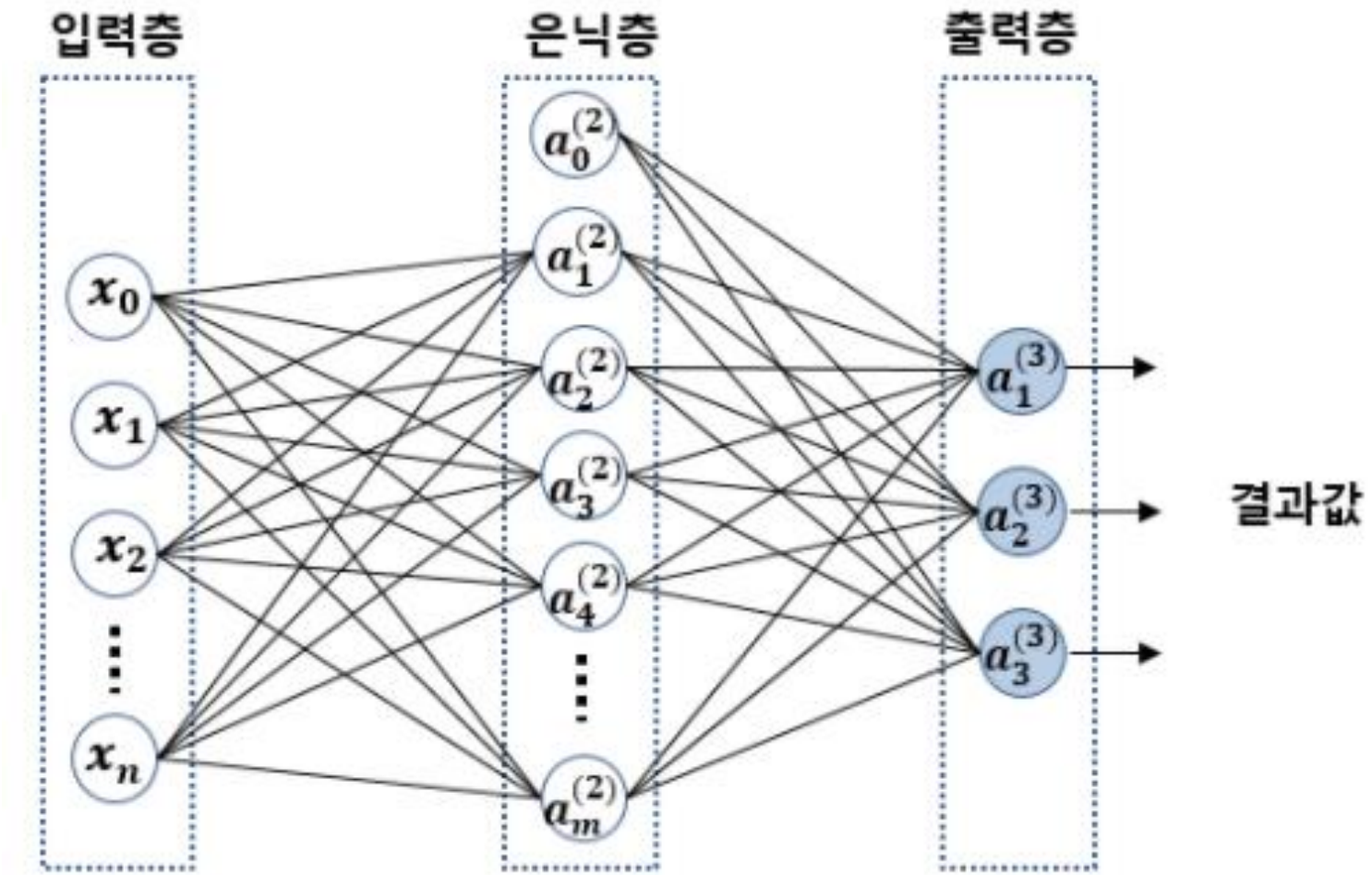
1. Linear Regression
(Boston 집값 예측, Boston housing price data set)

2. Logistic Regression
(필기체 숫자 학습 분류 모델, MNIST data set)

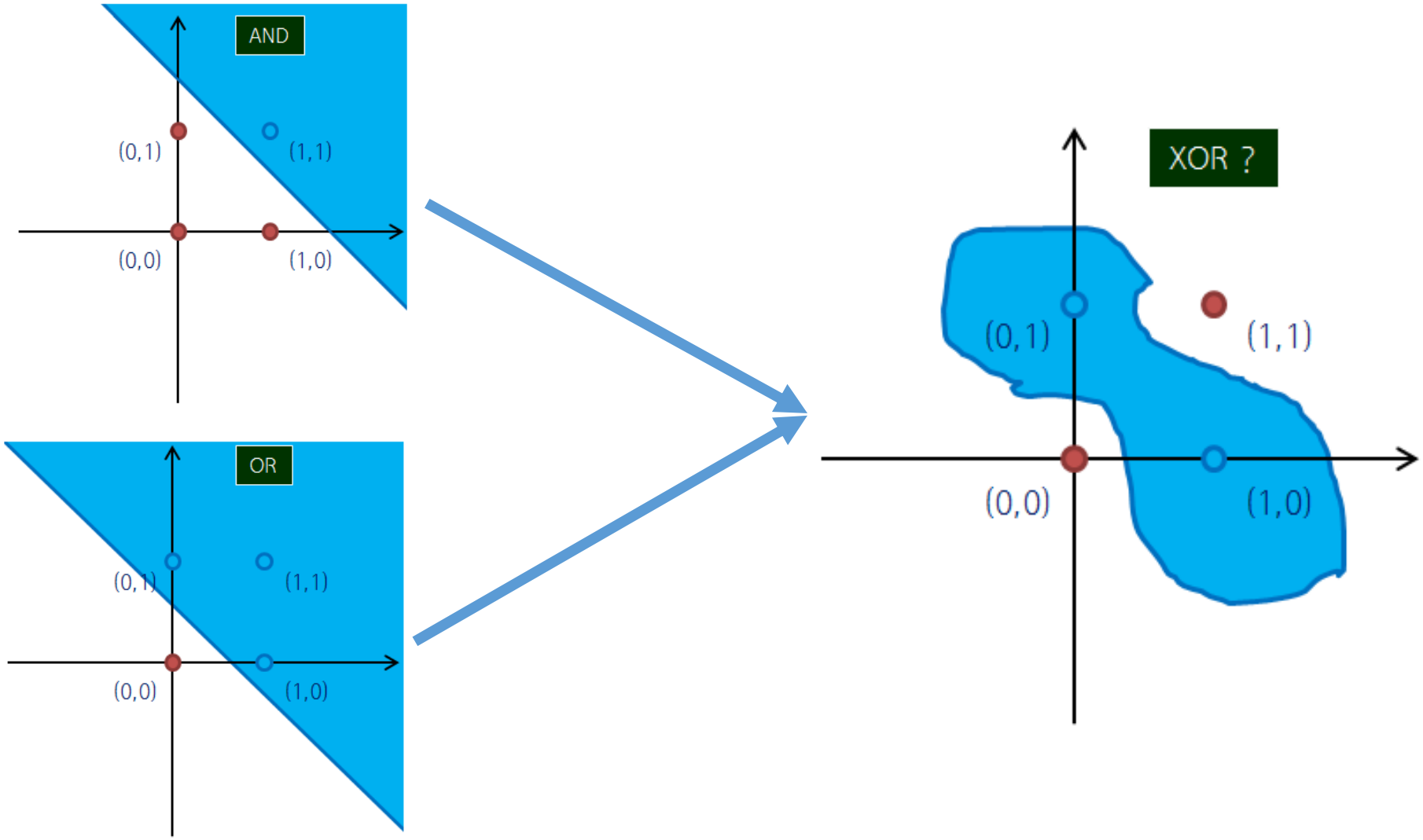
3. Neural-network
(필기체 숫자 학습 Deep-learning 모델, MNIST data set)

Neural-network 장점

Neural-network

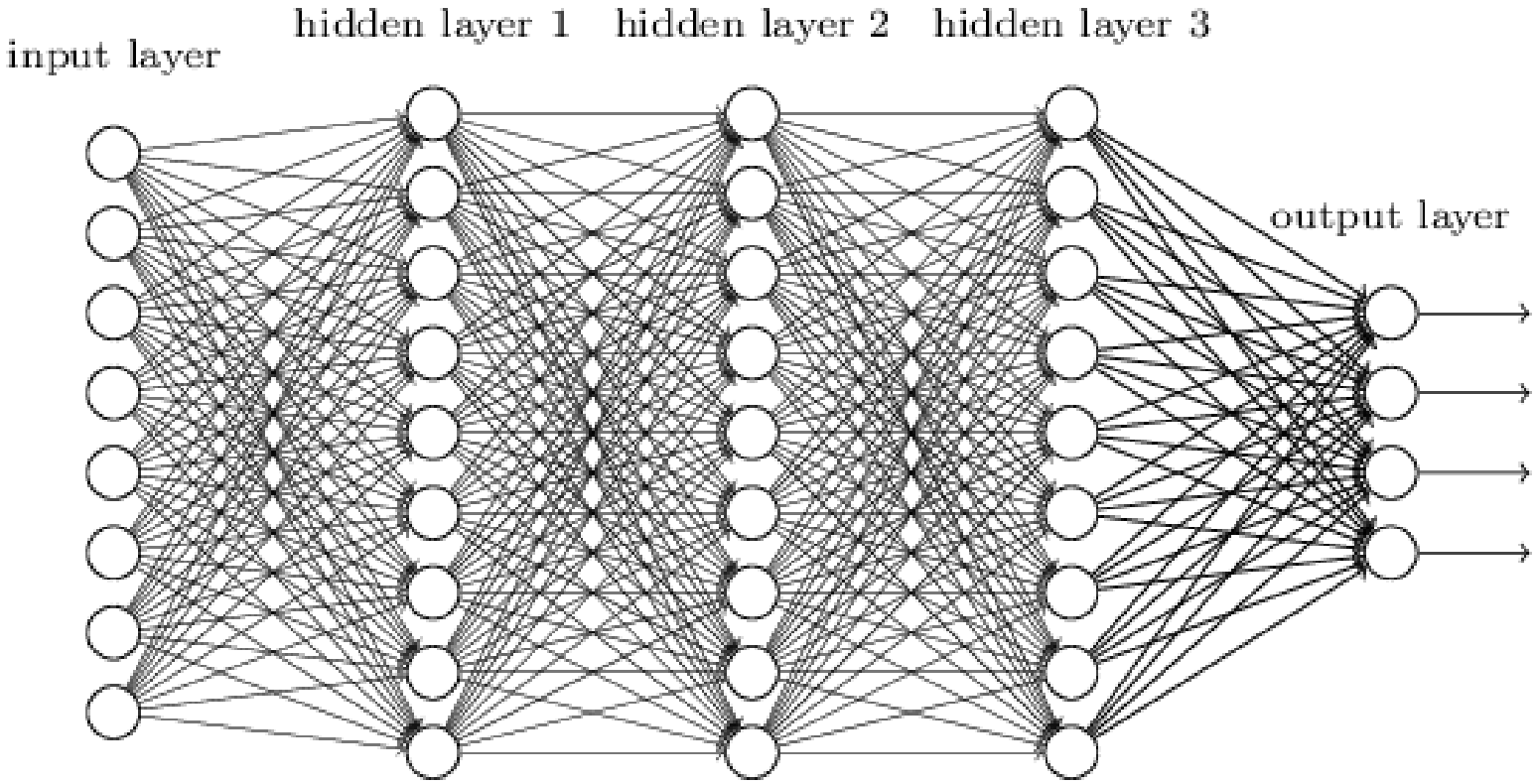


Neural-network

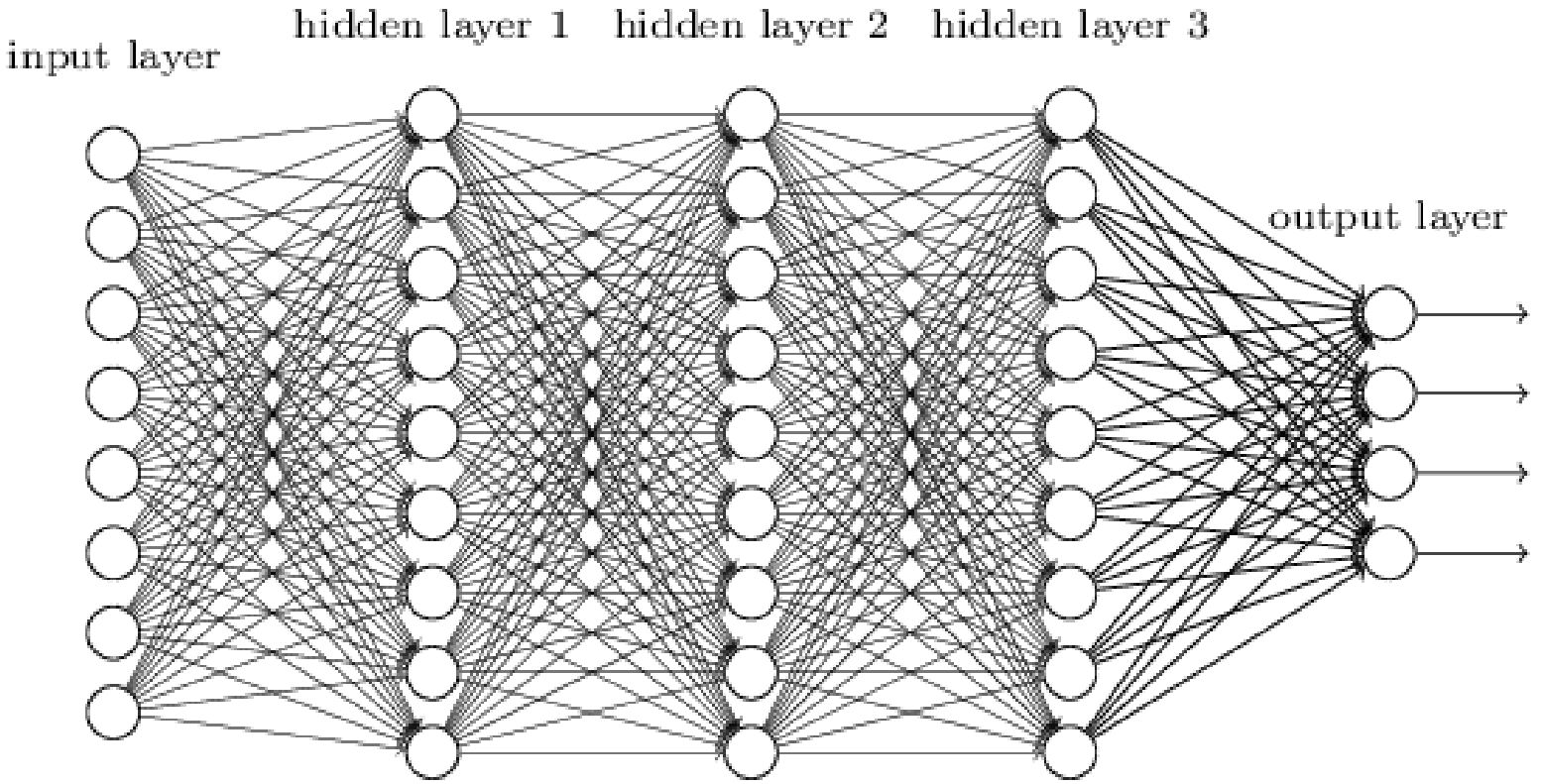


우리가 해야할 일은?

Neural-network

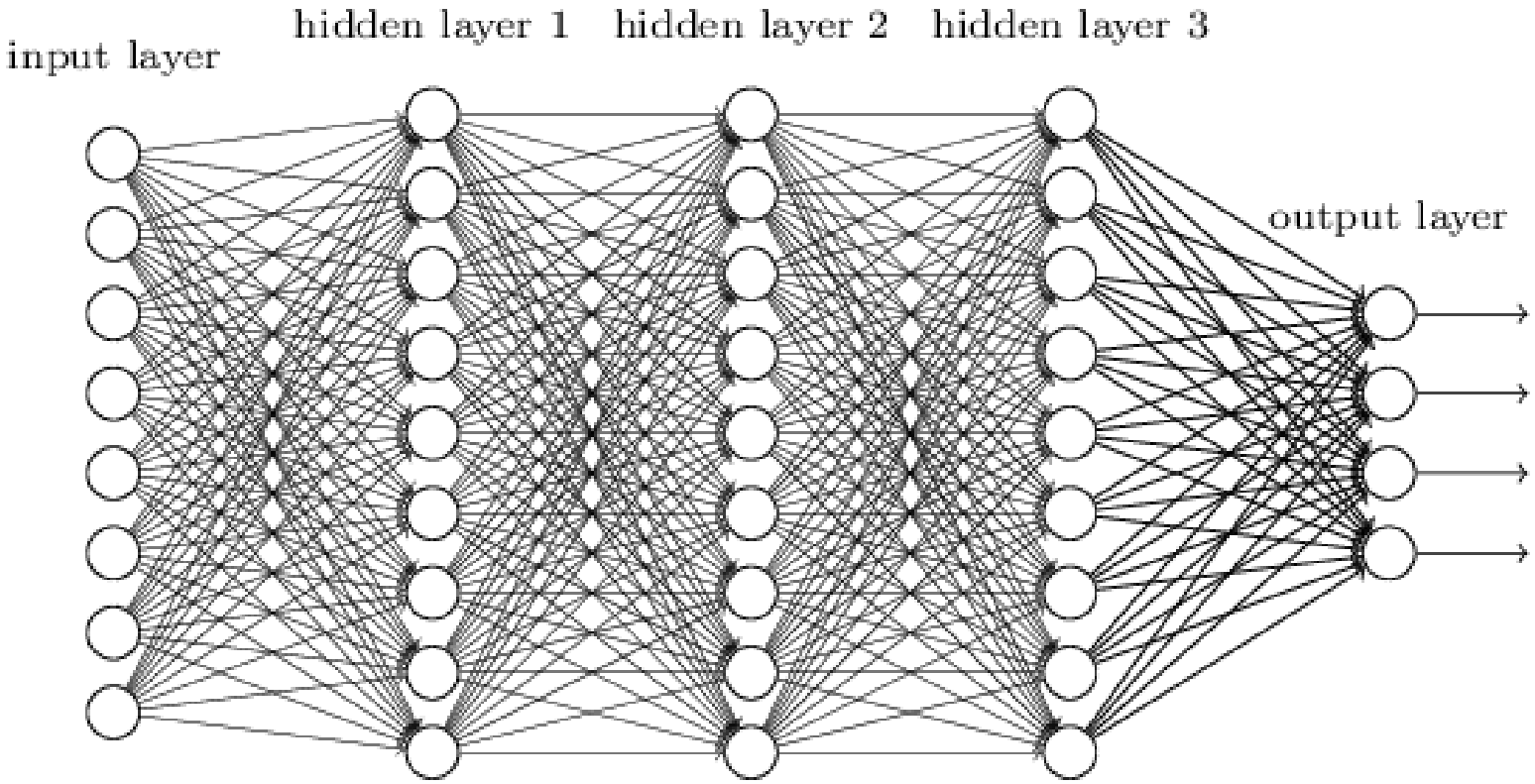


Neural-network



결과 예측

Neural-network



Layer 값 조정

다 함께 해봅시다.

Neural-network 설명

Q & A

감사합니다