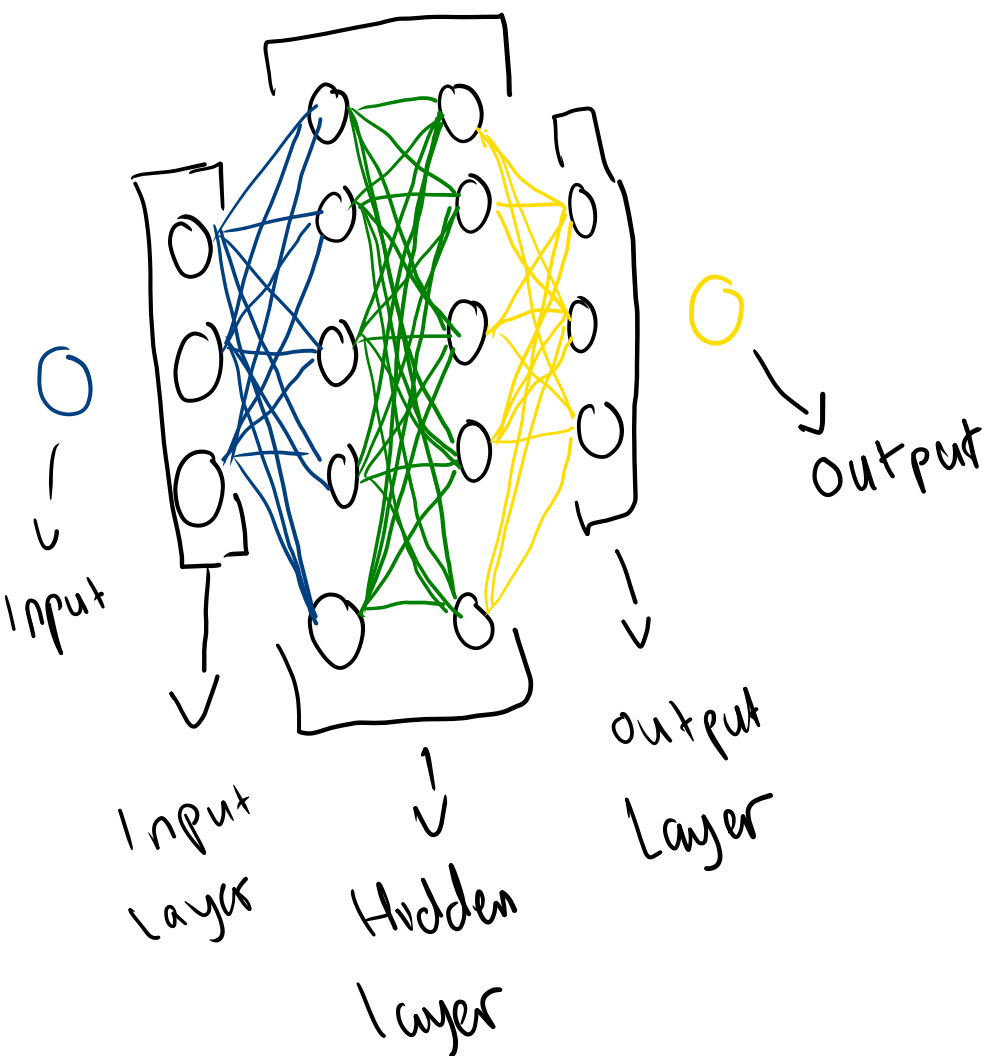


# Penjelasan Multi Layer Perceptron (by Wildan Aziz)



$$Sgd = \frac{1}{1 + e^{-x}}$$

- pada gambar tersebut merupakan MLP sederhana dengan 2 Hidden layer
- Input layer direpresentasikan sebagai nilai tabular 2 dimensi,  $A^{[0]} = (614, 7) \vee (614, 2)$
- Hidden layer merupakan operasi yang digunakan untuk memproses informasi dari hidden layer dengan fungsi aktivasi

$$Z^{[1]} = W^{[1]} \cdot A^{[0]} + B^{[1]}$$

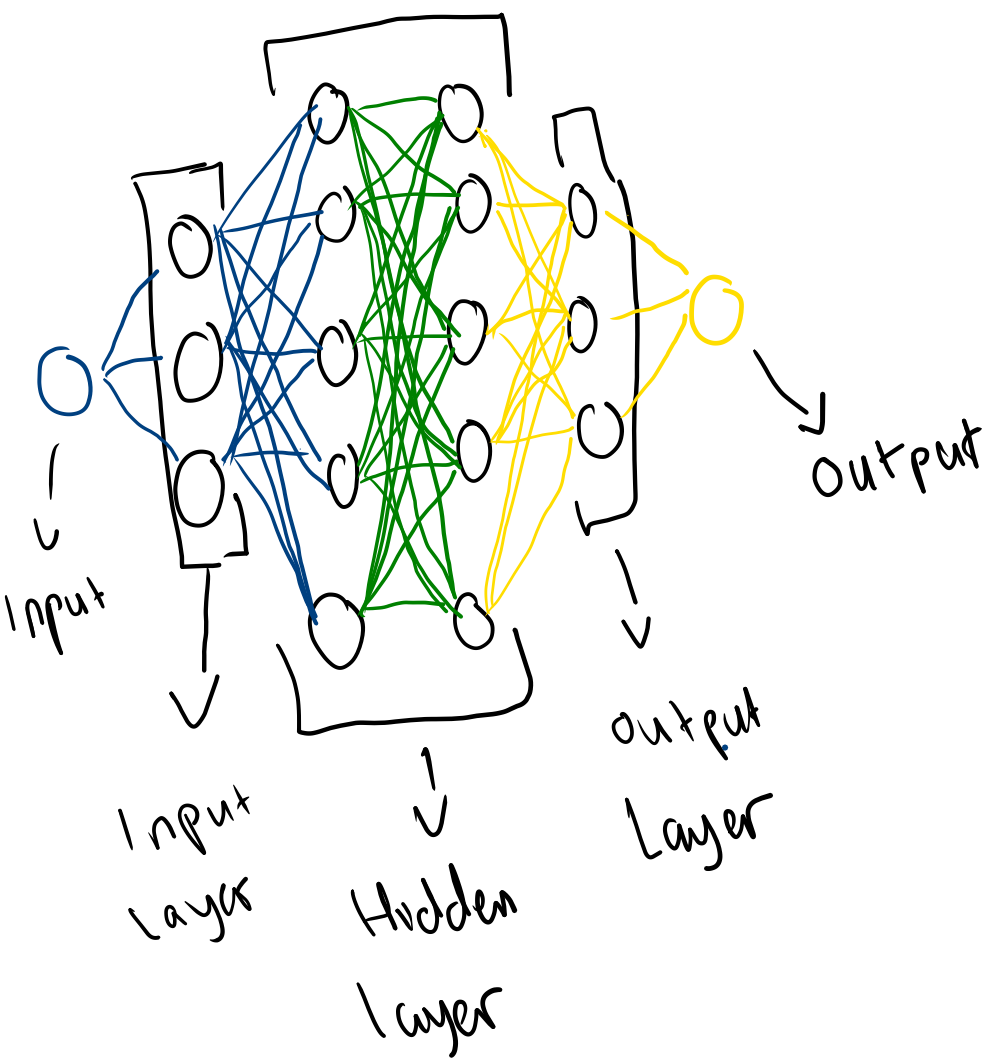
Forward propagation

$\downarrow$  bobot

$\downarrow$  Input Layer

$\downarrow$  bias

# Penjelasan Multi Layer Perceptron (by Wildan Aziz)

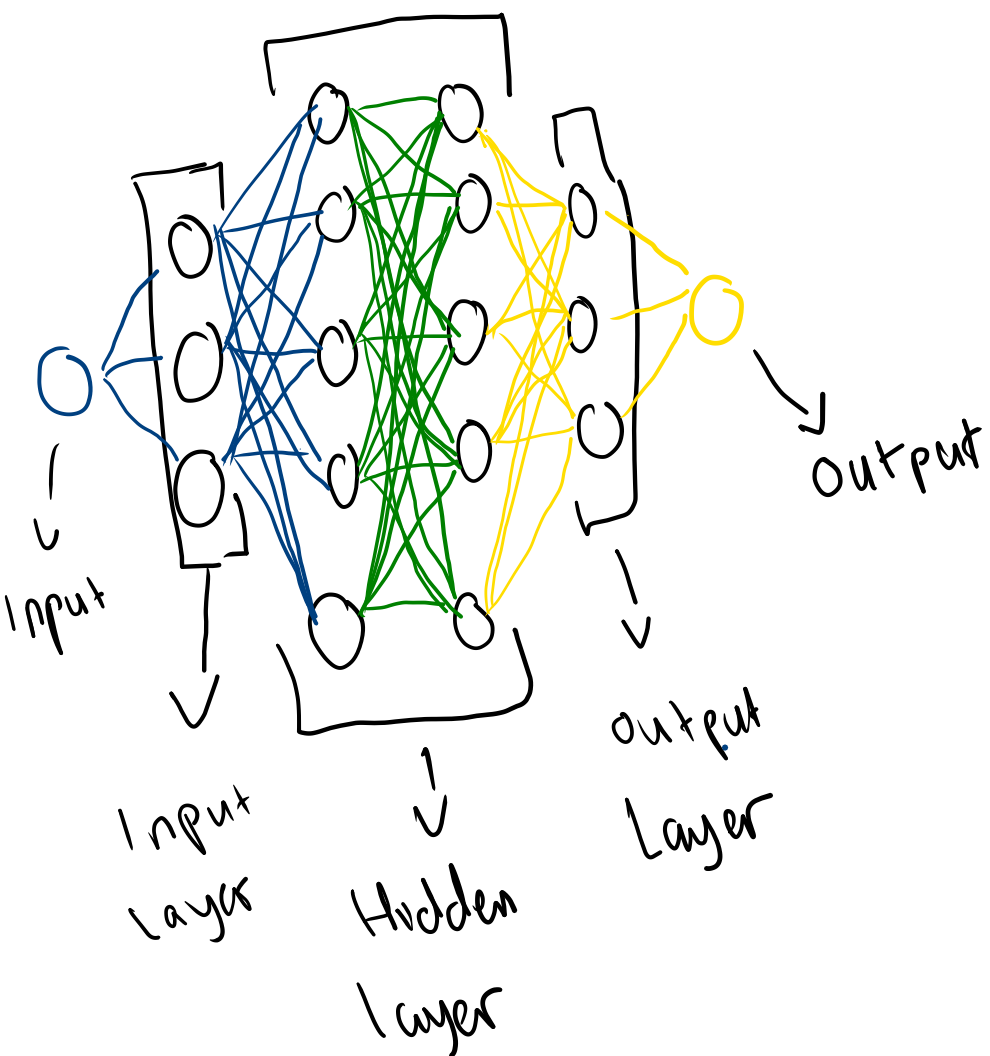


$$Sgd = \frac{1}{1 + e^{-x}}$$

- Pada proses menuju hidden layer selanjutnya terdapat fungsi aktivasi ReLu untuk menon liniearkan  $f(x) = \max(0, x)$   
apabila nilainya di bawah 0 maka = 0  
apabila nilainya maksimum kembalikan nilai tersebut
- ketika hendak mencapai bagian output maka akan di Flatten menggunakan fungsi aktivasi sigmoid

$$sgd = \frac{1}{1 + e^{-x}}$$

# Penjelasan Multi Layer Perceptron (by Wildan Azz)



- setelah dari output maka akan kembali lagi menggunakan back propagation dan pastinya di output encoding terlebih dahulu untuk menjaga ke konsistenan data
- menghitung error

$$dZ^{[2]} = A^{[2]} - Y \rightarrow \text{bias}$$

back propagation ←      → output

- update parameter, layer 2

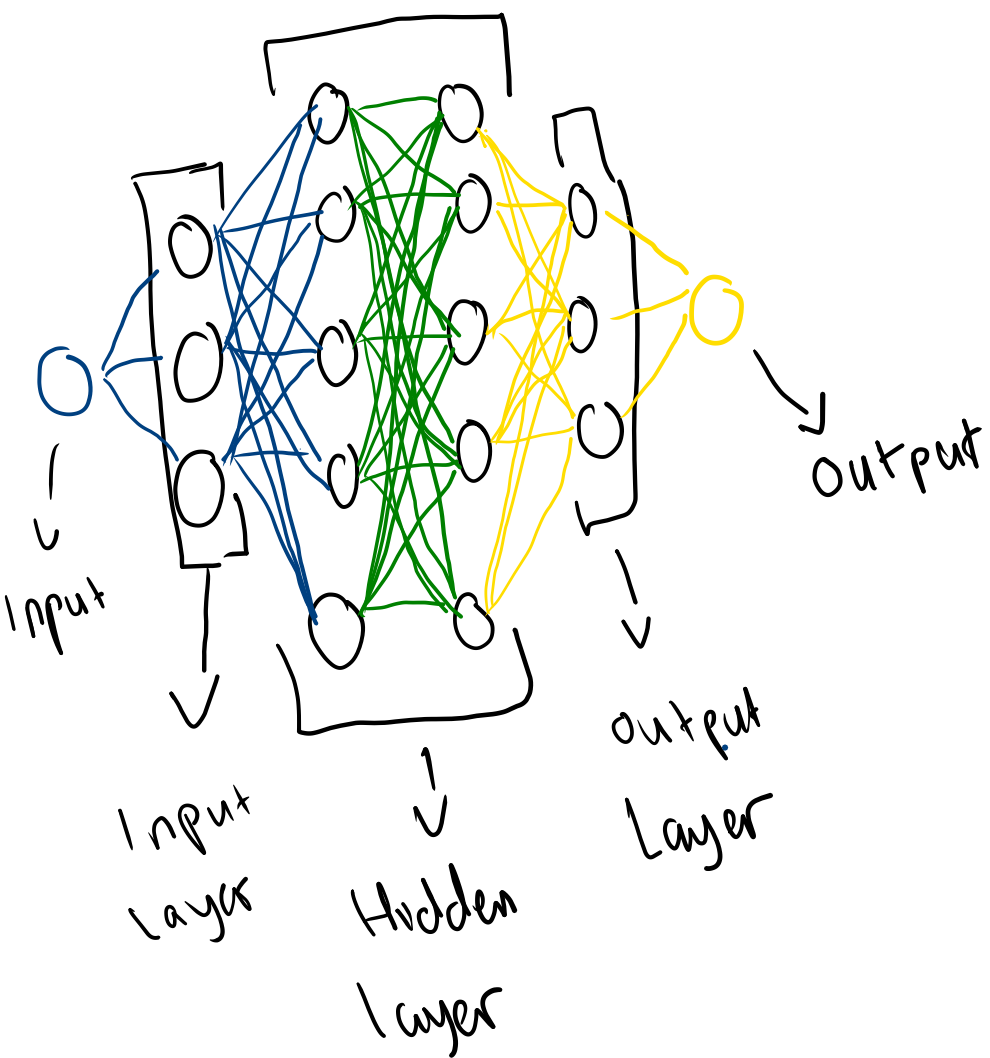
$$dW^{[2]} = \frac{1}{m} dZ^{[2]} A^{[1]T}$$

$$dB^{[2]} = \frac{1}{m} \sum dZ^{[2]}$$

di transpor agar sesuai dengan forward prop

$$Sgd = \frac{1}{1 + e^{-x}}$$

# Penjelasan Multi Layer Perceptron (by Wildan Aziz)



- propagasi nilai error ke hidden layer

$$\delta Z^{[1]} = W^{[2]T} \cdot \delta Z^{[2]}, \quad g'(Z^{[1]})$$

↳ turunan fungsi aktivasi

- Update parameter untuk layer 1

$$\delta W^{[1]} = \frac{1}{m} \delta Z^{[1]} X^T$$

$$\delta b^{[1]} = \frac{1}{m} \sum \delta Z^{[1]}$$

$$Sgd = \frac{1}{1 + e^{-x}}$$