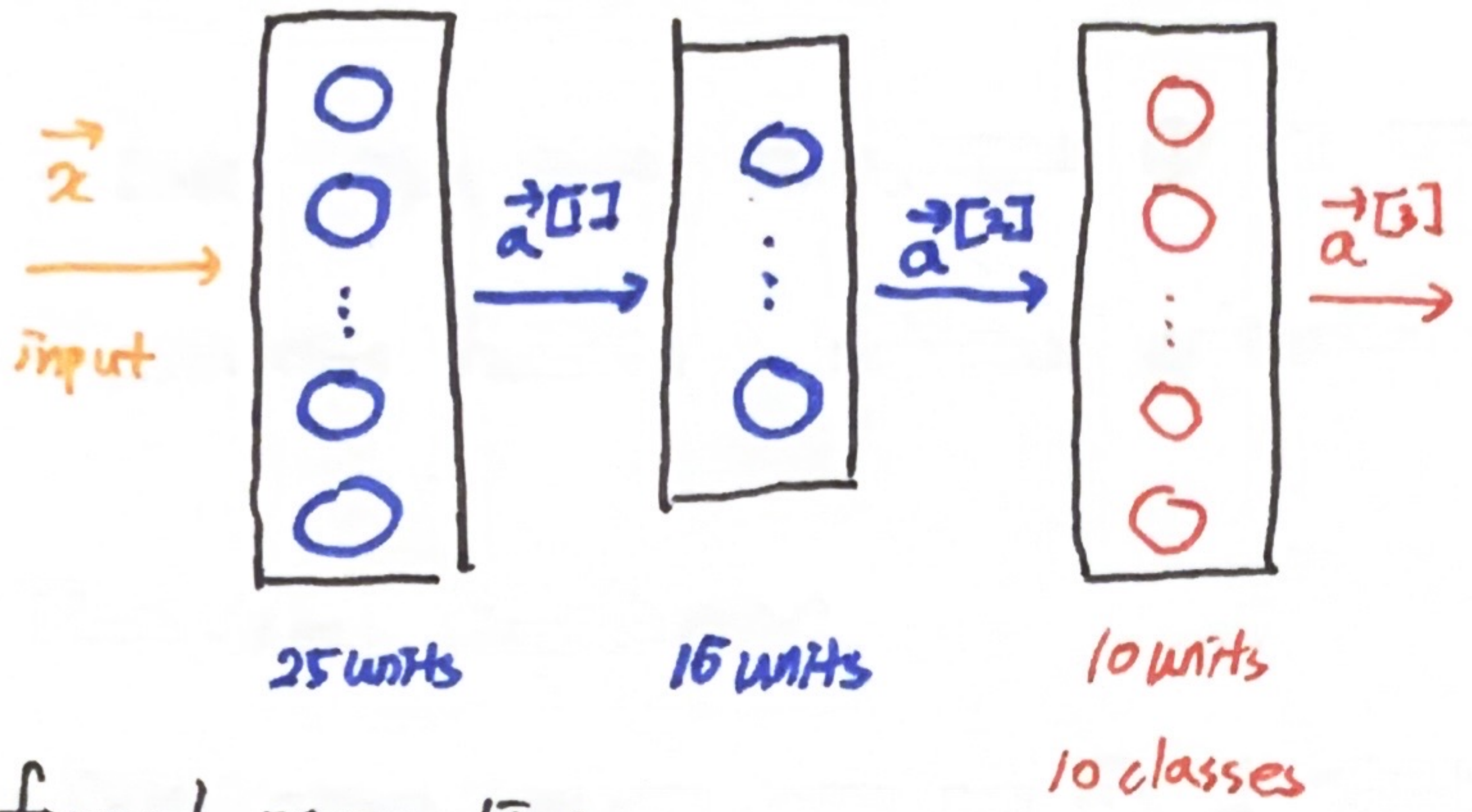
<Multiclass Classification - Neural Network with Softmax output>

ex) handwritten digits (classes: 0 ~ 9)



foward propagation

① given a $\vec{x}$ to 1st layer

② $\vec{a}^{[1]}$ gets computed exactly the same as before (relu activation function)

③ $\vec{a}^{[2]}$ gets computed exactly the same as before (relu activation function)

④ $\vec{a}^{[3]}$ gets computed to make results of 10 classes (softmax regression with output layer)

$\vec{a}^{[2]} \rightarrow$

$z_1^{[3]} = \vec{w}_1^{[3]} \cdot \vec{a}^{[2]} + b_{10}^{[3]} \rightarrow$ softmax $\rightarrow a_1^{[3]} = \dfrac{e^{z_1^{[3]}}}{e^{z_1^{[3]}} + \cdots + e^{z_{10}^{[3]}}} = P(y=1|\vec{x})$

$\vdots$

$z_{10}^{[3]} = \vec{w}_{10}^{[3]} \cdot \vec{a}^{[2]} + b_{10}^{[3]} \rightarrow$ softmax $\rightarrow a_{10}^{[3]} = \dfrac{e^{z_{10}^{[3]}}}{e^{z_1^{[3]}} + \cdots + e^{z_{10}^{[3]}}} = P(y=10|\vec{x})$

* Softmax regression VS other activation function (logistic, sigmoid ...)

① logistic, sigmoid, linear function

$$a_1^{[3]} = g(z_1^{[3]}),\quad a_2^{[3]} = g(z_2^{[3]})$$

· a_1 : only z_1의 함수값

· a_2 : only z_2의 함수값

⇒ 각각의 activation value($a_1, a_2 \cdots$)를 얻기 위해서는 각각의 $z_1, z_2 \cdots$에 대한 함수(g: sigmoid or logistic or linear)를 취함

② Softmax function

$$a_1^{[3]} = \frac{e^{z_1^{[3]}}}{e^{z_1^{[3]}} + \cdots + e^{z_{10}^{[3]}}}$$

$$\vdots$$

$$a_{10}^{[3]} = \frac{e^{z_{10}^{[3]}}}{e^{z_1^{[3]}} + \cdots + e^{z_{10}^{[3]}}}$$

$$\Rightarrow a_j = \frac{e^{z_j}}{\sum_{k=1}^{N} e^{z_k}}$$

⇒ 각각의 activation value($a_1, a_2 \cdots$)를 얻기위해선 모든 z값에 대한 softmax를 취해야함

= $a_1 \sim a_{10}$을 계산하기위해선 $g(z_1 \sim z_{10})$을 동시에 계산해야함