## Week 4 – Neural Network Representation

### Nonlinear Hypothesis :

$\boxed{\text{Nonlinear Hypothesis}} \rightarrow h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1 x_2 + \theta_4 x_1^2 x_2 + \cdots)$

If we just consider quandratic features or less:

Total number of combination

$$= \frac{n \times (n-1)}{2} + n = \frac{n(n+1)}{2}$$

where n is number of feature

Example   Let there are 3 features.

The highest degree of combination of features = 2

$\therefore$ Number of quandratic feature-combination

$$= \frac{3 \times 2}{2} = 3$$

Number of first-order feature-combination = 3

$\therefore$ Total feature combination = 3 + 3 = 6

## Week 4 – Neural Network Representation

Nonlinear Hypothesis :

$\boxed{\text{Nonlinear Hypothesis}} \rightarrow h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1 x_2 + \theta_4 x_1^2 x_2 + \cdots)$

If we just consider quandratic features or less:

Total number of combination

$$= \frac{n \times (n-1)}{2} + n = \frac{n(n+1)}{2}$$

where n is number of feature

Example Let there are 3 features.

The highest degree of combination of features = 2

∴ Number of quandratic feature-combination

$$= \frac{3 \times 2}{2} = 3$$

Number of first order feature-combination = 3

∴ Total feature combination = 3 + 3 = 6

- Similarly for feature combination upto order 3, number of combination is in order of $n^3$.

## Neurons and Brains

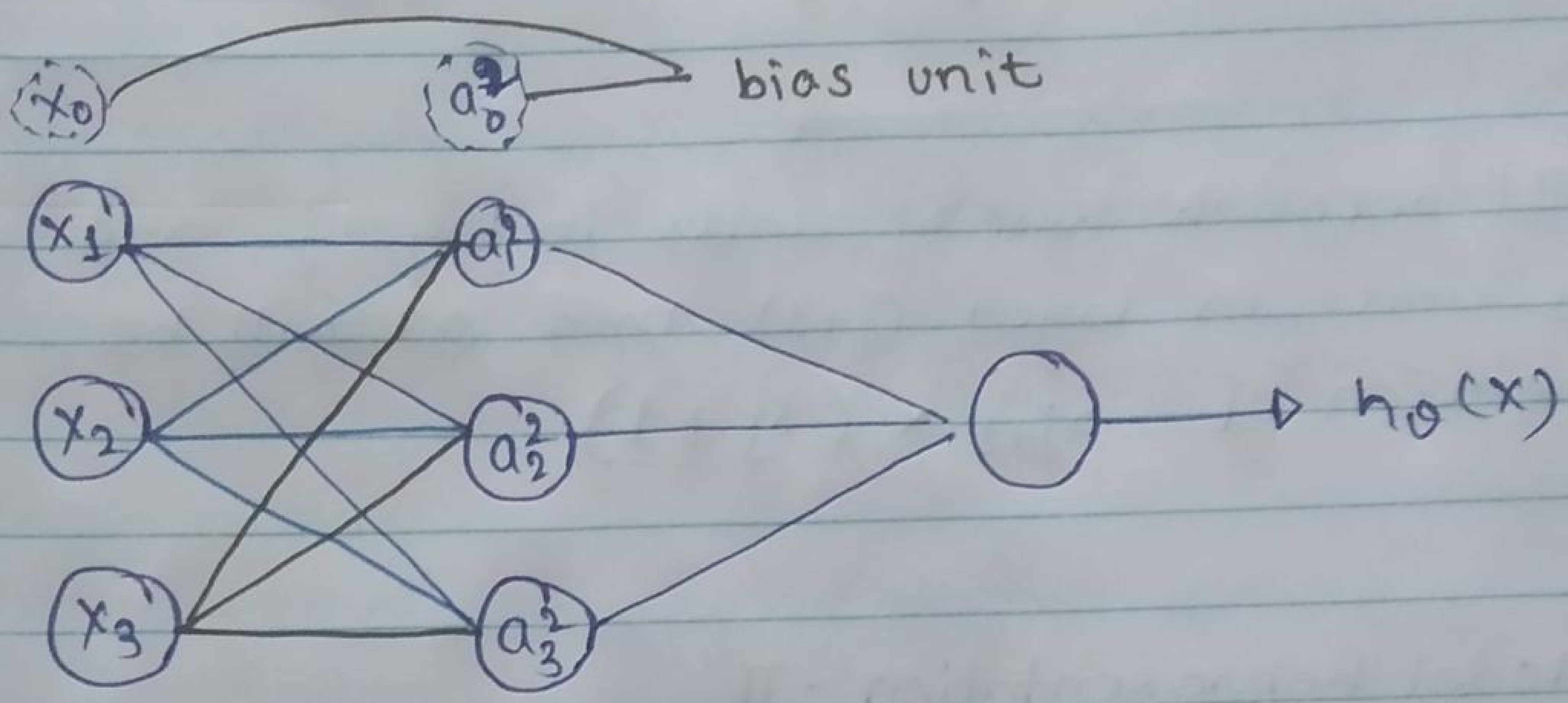
- Inspiration behind Neural Network

There is a theory (One learning algorithm) that human intelligence stems from a single algorithm. The ideas arise from experiments suggesting that the portion of your brain dedicated to processing sound from your ears also handle sight for your eyes. This is possible only while your brain is in the earlier stages of development, but it implies that the brain is – at its core – a general purpose machine that can be tuned to specific tasks.

— The man behind Google Brain: Andrew Ng and the quest for AI

## Model Representation

### A sample Neural Network model




$a_i^j \longrightarrow$ activation of unit $i$ in layer $j$

$\theta^j \longrightarrow$ Matrix of weights controlling function mapping from layer $j$ to layer $j+1$

$$
\begin{cases}
a_1^2 = g(\theta_{10}^1 x_0 + \theta_{11}^1 x_1 + \theta_{12}^1 x_2 + \theta_{13}^1 x_3) \\
a_2^2 = g(\theta_{20}^1 x_0 + \theta_{21}^1 x_1 + \theta_{22}^1 x_2 + \theta_{23}^1 x_3) \\
a_3^2 = g(\theta_{30}^1 x_0 + \theta_{31}^1 x_1 + \theta_{32}^1 x_2 + \theta_{33}^1 x_3)
\end{cases}
\quad (1)
$$

bias unit

$$h_\Theta(x) = a_1^3 = g\left(\Theta_{10}^2 a_0^2 + \Theta_{11}^2 a_1^2 + \Theta_{12}^2 a_2^2 + \Theta_{13}^2 a_3^2\right) \quad \text{—— (2)}$$

- If network has $s_j$ units in layer $j$ and $s_{j+1}$ units in layer $(j+1)$ then $\Theta^j$ will be dimension of $s_{j+1} \times (s_j + 1)$

## Model Representation - II

- Forward Propagation - Vectorized Implementation

write

$$\left.\begin{aligned} a_1^2 &= g(z_1^2) \\ a_2^2 &= g(z_2^2) \\ a_3^2 &= g(z_3^2) \end{aligned}\right\} \text{Compare with eq}^n\text{ set (1)} \quad \text{—— (3)}$$

$$z^2 = \begin{bmatrix} z_0^2 \\ z_1^2 \\ z_2^2 \\ z_3^2 \end{bmatrix}$$

And

$$X = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

$$\therefore \; z^2 = \theta^1 a^1$$

## Explanation

$$z^2 = \begin{bmatrix} z_1^2 \\ z_2^2 \\ z_3^2 \end{bmatrix}$$

$$= \begin{bmatrix} \theta^1_{10} x_0 + \theta^1_{11} x_1 + \theta^1_{12} x_2 + \theta^1_{13} x_3 \\ \theta^1_{20} x_0 + \theta^1_{21} x_1 + \theta^1_{22} x_2 + \theta^1_{23} x_3 \\ \theta^1_{30} x_0 + \theta^1_{31} x_1 + \theta^1_{32} x_2 + \theta^1_{33} x_3 \end{bmatrix}$$

$$= \begin{bmatrix} \theta^1_{10} & \theta^1_{11} & \theta^1_{12} & \theta^1_{13} \\ \theta^1_{20} & \theta^1_{21} & \theta^1_{22} & \theta^1_{23} \\ \theta^1_{30} & \theta^1_{31} & \theta^1_{32} & \theta^1_{33} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

$= \theta^1 x \quad (x \equiv a^1)$

$\therefore \boxed{z^2 = \theta^1 a^1}$

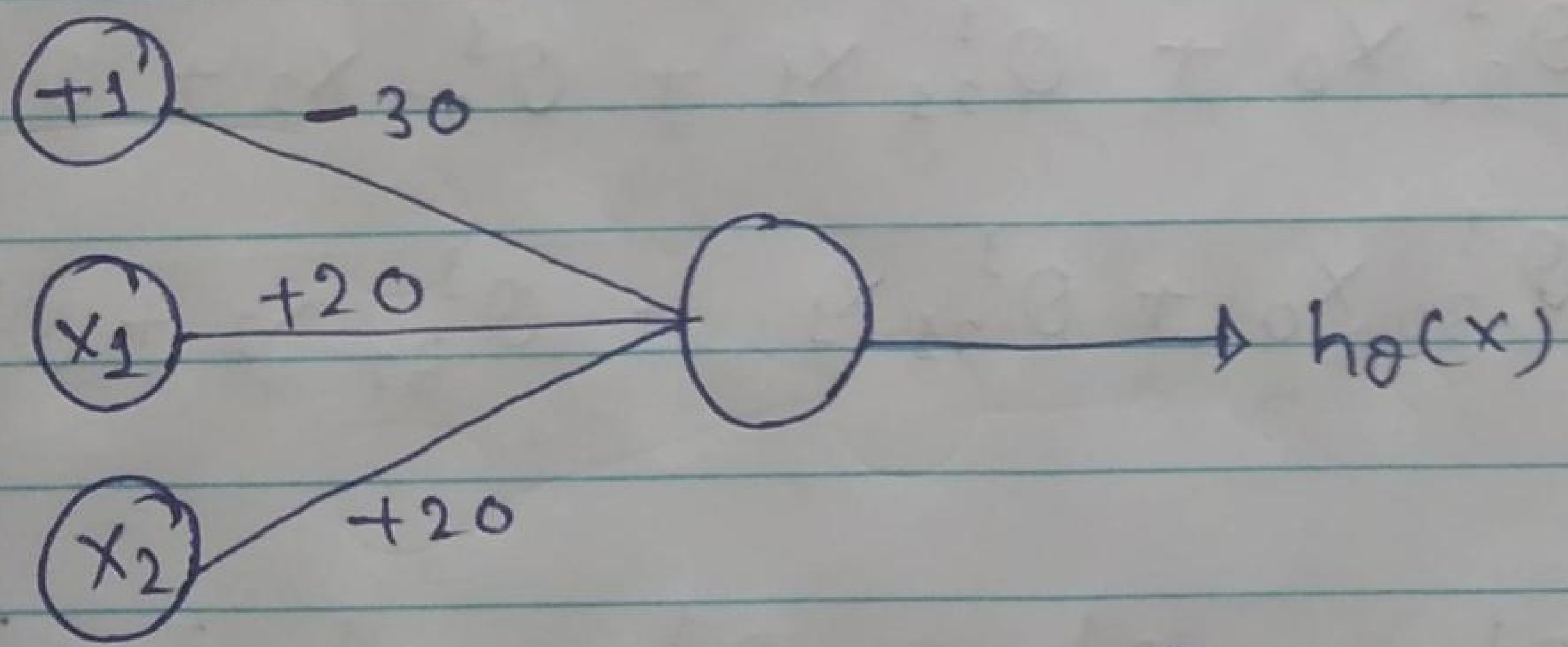
$\boxed{a^2 = g(z^2)}$ → add $a_0^2 = 1$

Similarly, $z^3 = \theta^2 a^2$

and, $a^3 = g(z^3)$

## Neural Network - application

- $\boxed{x_1 \text{ AND } x_2}$




$$h_\theta(x) = g[\underbrace{-30}_{\theta^1_{10}} + \underbrace{20}_{\theta^1_{11}} x_1 + \underbrace{20}_{\theta^1_{12}} x_2]$$

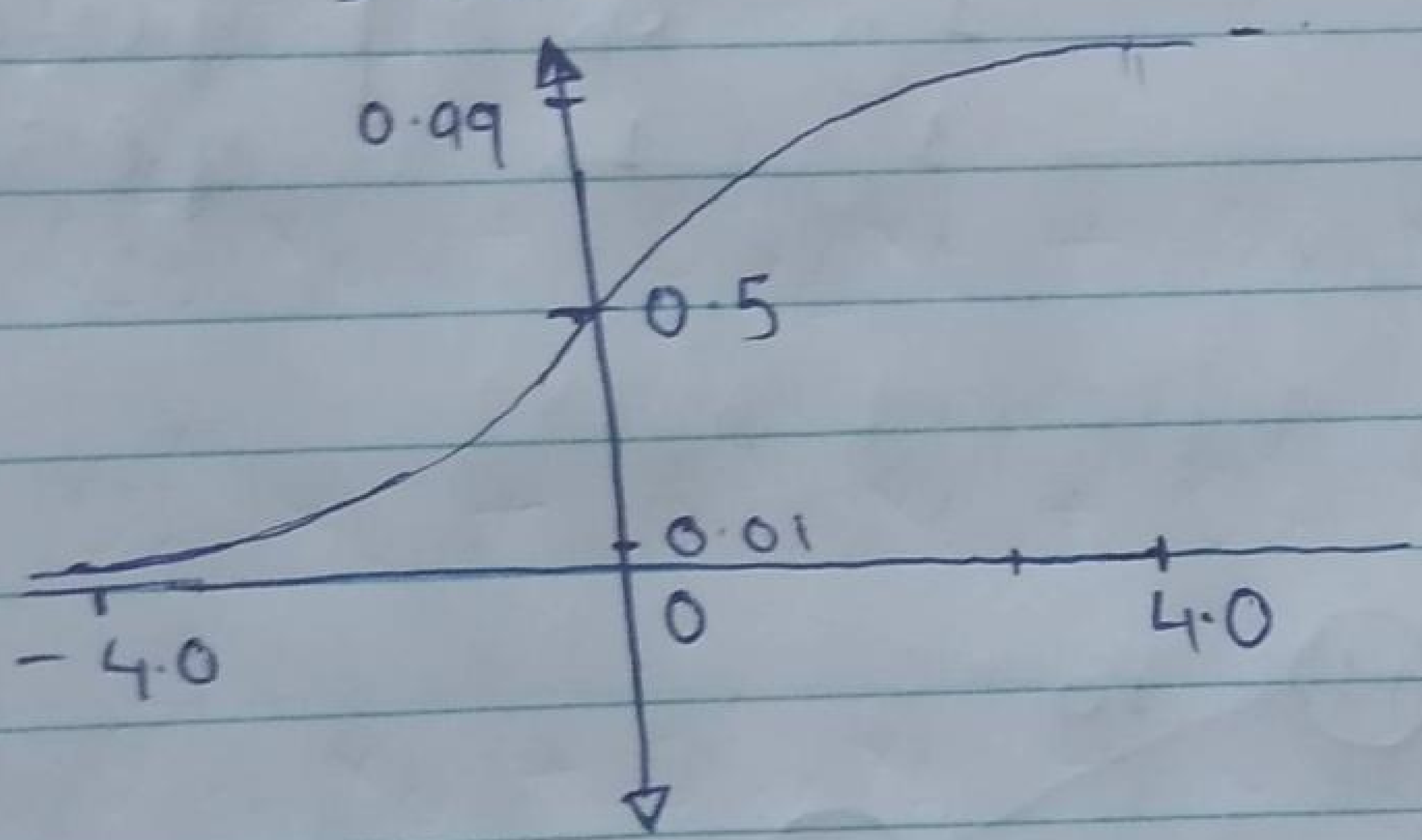
## Output table

| $x_1$ | $x_2$ | $h_\theta(x)$ |
|---|---|---|
| 0 | 0 | $g(-30) \approx 0$ |
| 0 | 1 | $g(-10) \approx 0$ |
| 1 | 0 | $g(-10) \approx 0$ |
| 1 | 1 | $g(10) \approx 1$ |

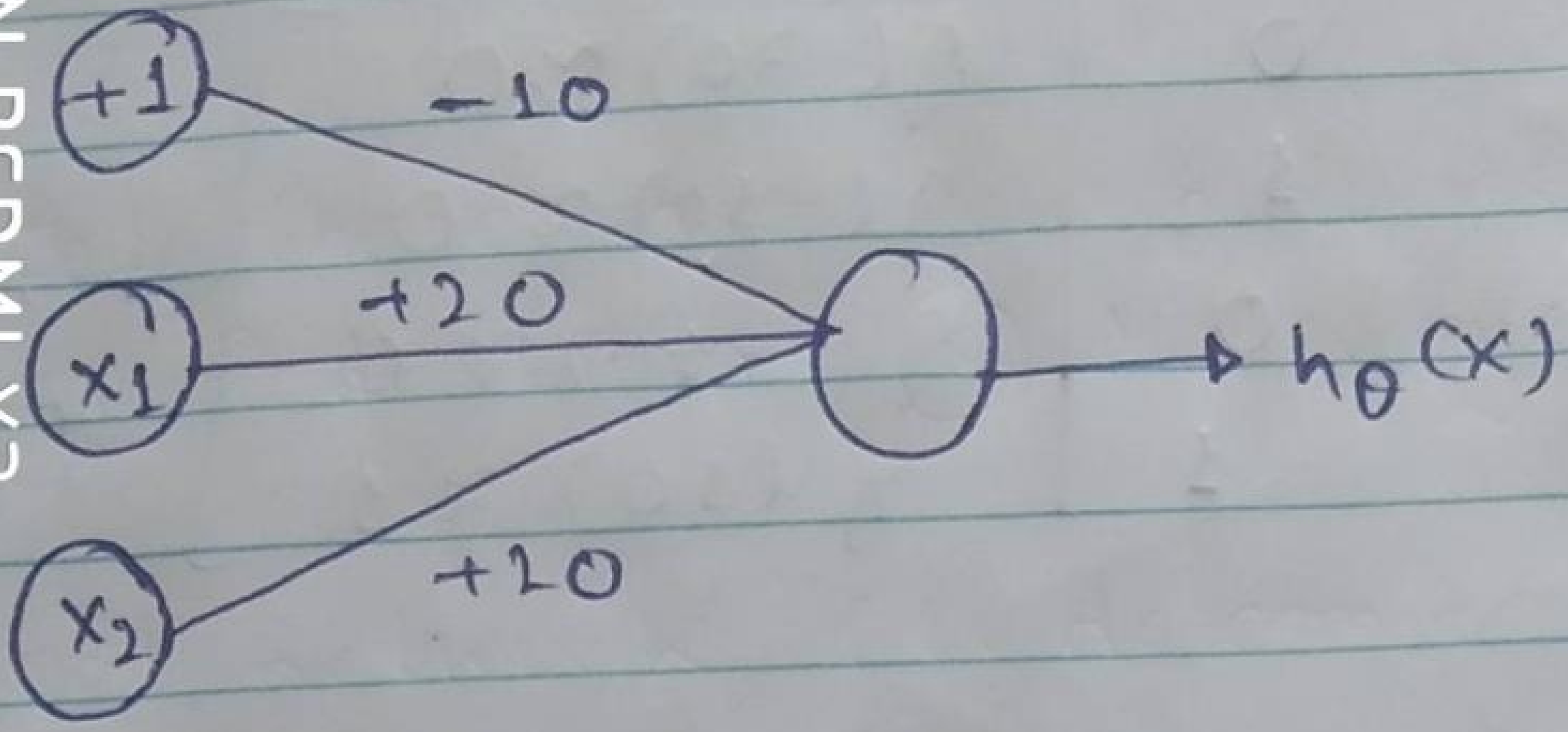
→ AND gate

## Notes on Logistic function

$$y = f(x) = \frac{1}{1 + e^{-x}}$$



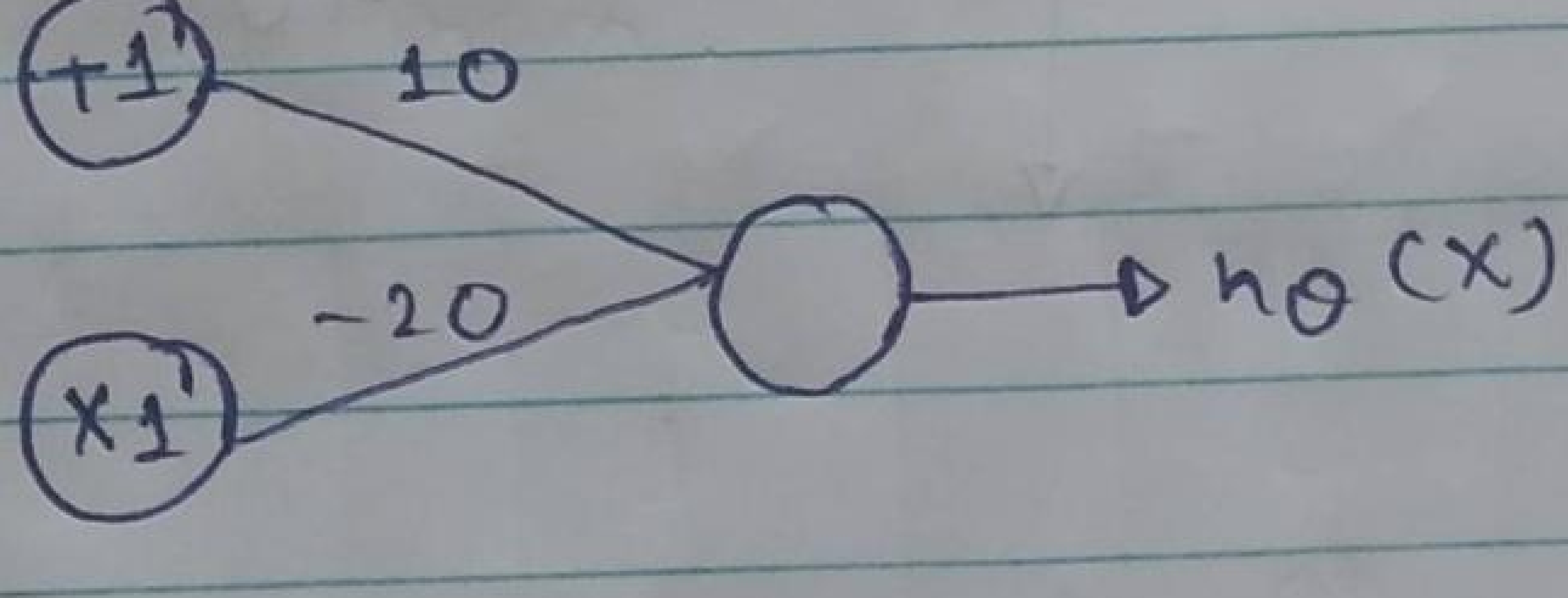
$\boxed{x_1 \text{ OR } x_2}$




Output

| $x_1$ | $x_2$ | $h_\theta(x)$ |
|---|---|---|
| 0 | 0 | $g(-10) \approx 0$ |
| 0 | 1 | $g(+10) \approx 1$ |
| 1 | 0 | $g(+10) \approx 1$ |
| 1 | 1 | $g(+30) \approx 1$ |

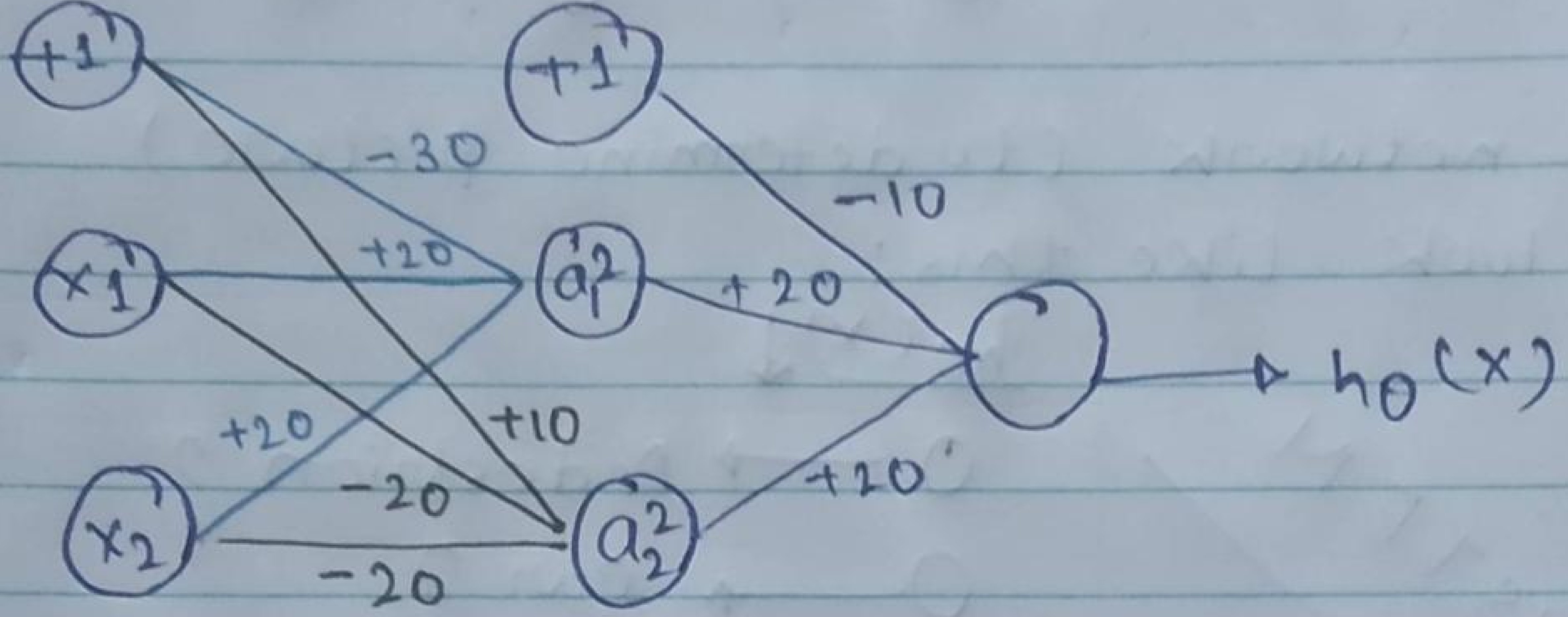
- Negation




| $x_1$ | $h_\theta(x)$ |
|---|---|
| 0 | $g(10) \simeq 1$ |
| 1 | $g(-10) \simeq 0$ |

$\boxed{x_1 \text{ XNOR } x_2}$

$x_1$ XNOR $x_2 \equiv$ (NOT $x_1$) AND (NOT $x_2$)




<u>Output</u>

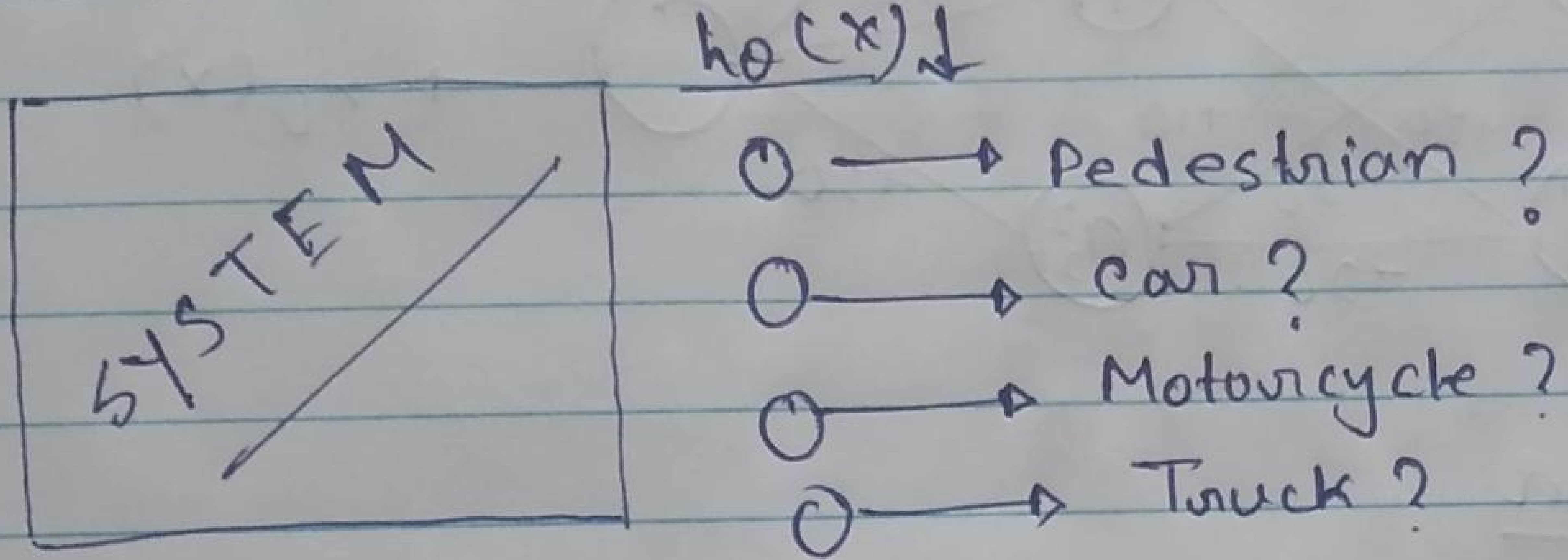
| $x_1$ | $x_2$ | $a_1^2$ | $a_2^2$ | $h_\theta(x)$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 |

## Multiclass classification

### One-vs-all technique

Let, there are four pictures → (1) car (2) Pedestrian (3) Motorcycle (4) Truck

Neural network (to determine picture) should look like this:




different different $h_\theta(x)$

$y^i$ is one of them:

$$\begin{bmatrix}1\\0\\0\\0\end{bmatrix}\quad\begin{bmatrix}0\\1\\0\\0\end{bmatrix}\quad\begin{bmatrix}0\\0\\1\\0\end{bmatrix}\quad\begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$