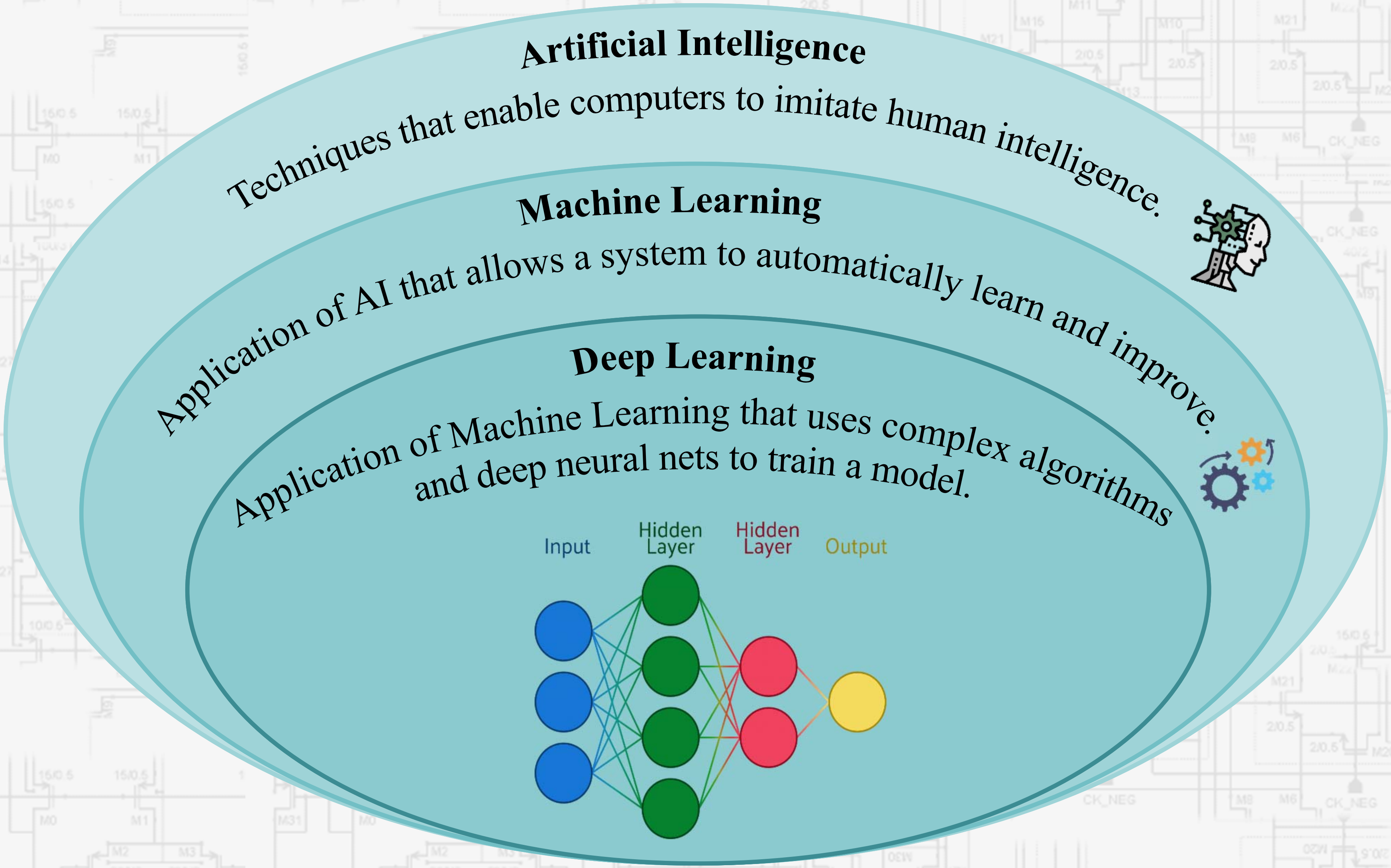
Introduction to Deep Learning

Shanq-Jang Ruan, Ph. D.
Distinguished Professor
Dept. Electronic and Computer Engineering
National Taiwan University of Science and Technology

Outline

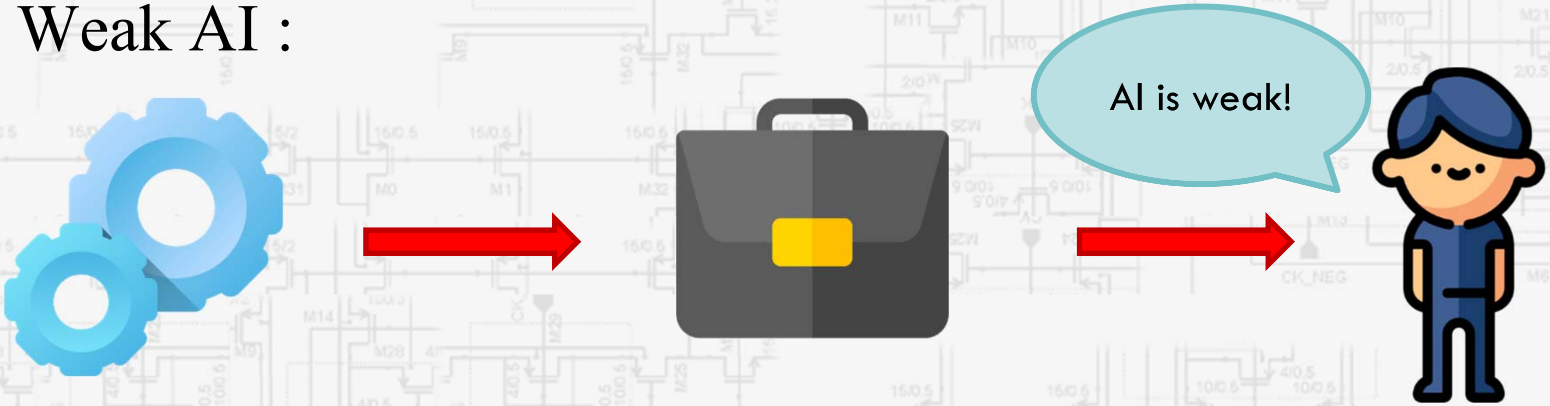
- What is AI?
- Biological Inspiration
- Perceptron
- Neural Networks
- The History of AI
- Applications for Deep Learning
- Conclusion

What is AI?

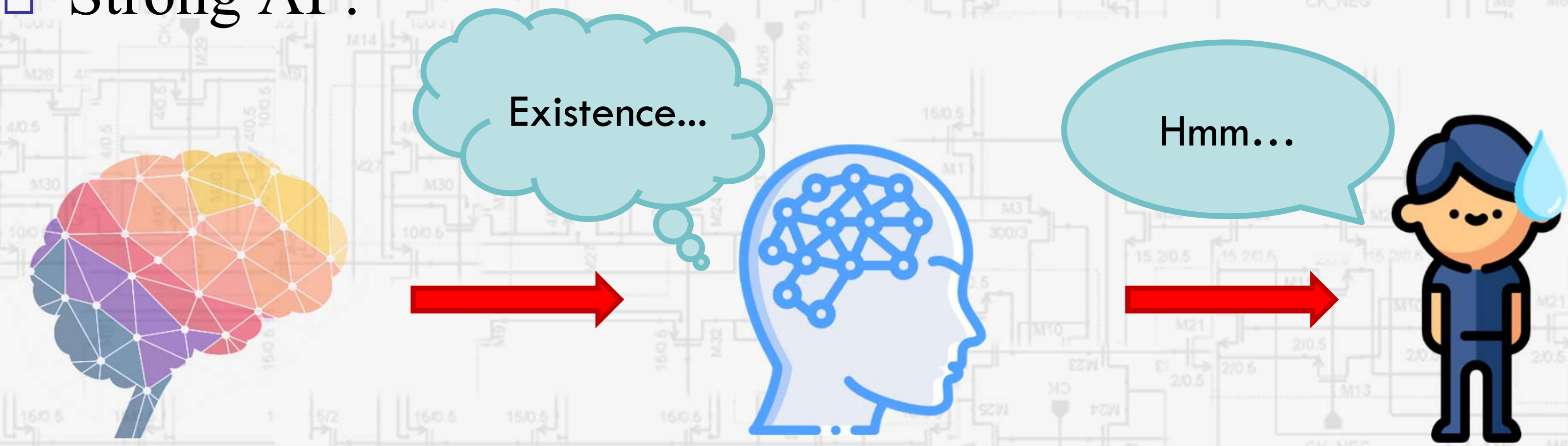

Artificial Intelligence
Techniques that enable computers to imitate human intelligence.
Machine Learning
Application of AI that allows a system to automatically learn and improve.
Deep Learning
Application of Machine Learning that uses complex algorithms
and deep neural nets to train a model.
Input
Hidden Layer
Hidden Layer
Output

What is AI?

- Weak AI :



  - Weak AI can help humans to finish time-consuming tasks.
- Strong AI :



  - Strong AI has consciousness, objective thoughts, self-awareness…

What is AI?

- Four stages:

Level 1
Program
Controlling

Ex: NEURO FUZZY washing machine in the 90s.

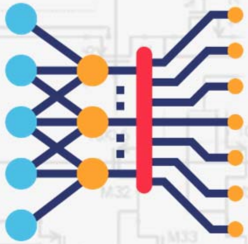
Level 2
Traditional
AI

Ex: IQ test solving, maze problem, diagnostic program.

Level 3
Machine
Learning

Learn the relationship between input and output.

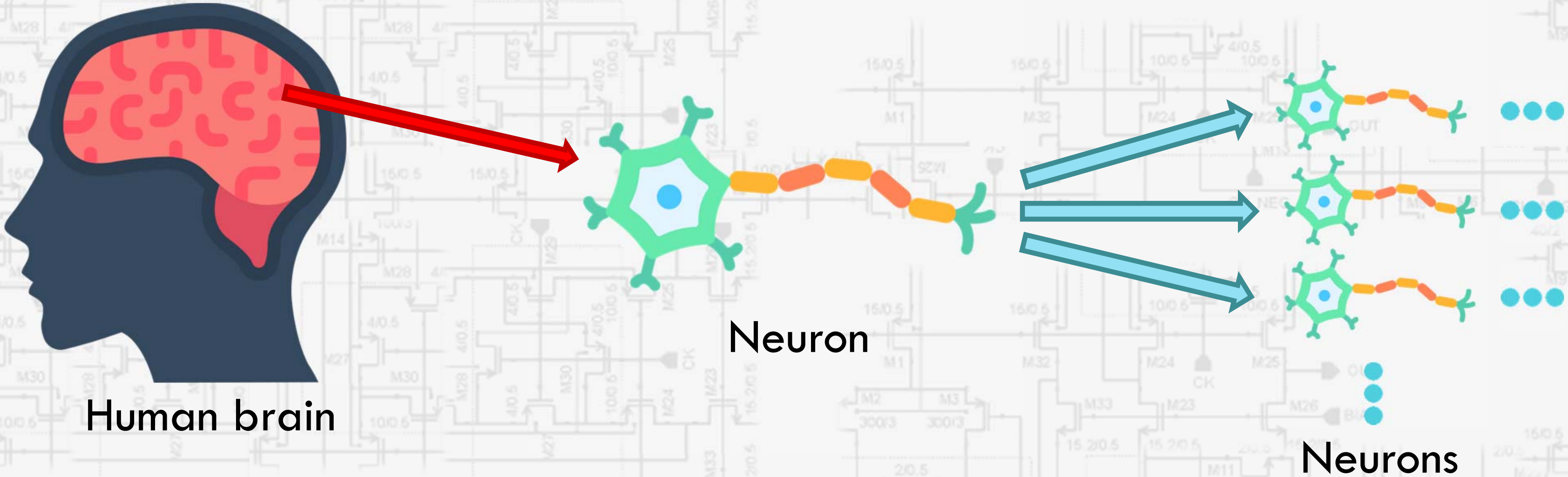
Level 4
Deep
Learning



Learn the features and increase the ability of recognition by itself.

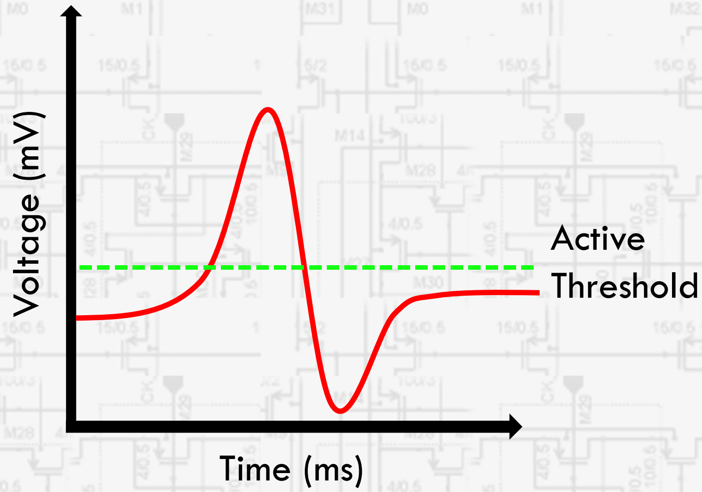
Biological Inspiration

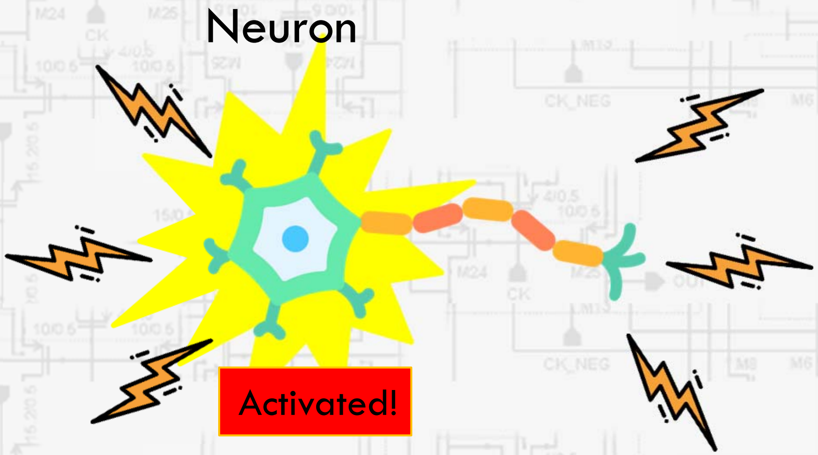
- Biological neural networks (brains) are composed of roughly 86 billion neurons connected to many other neurons.



- Neurons are cells within the nervous system, which transmit information to other nerve cells.

Biological Inspiration

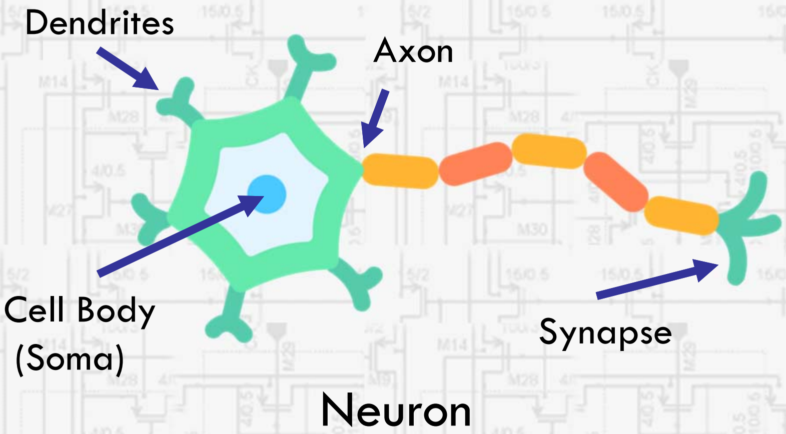


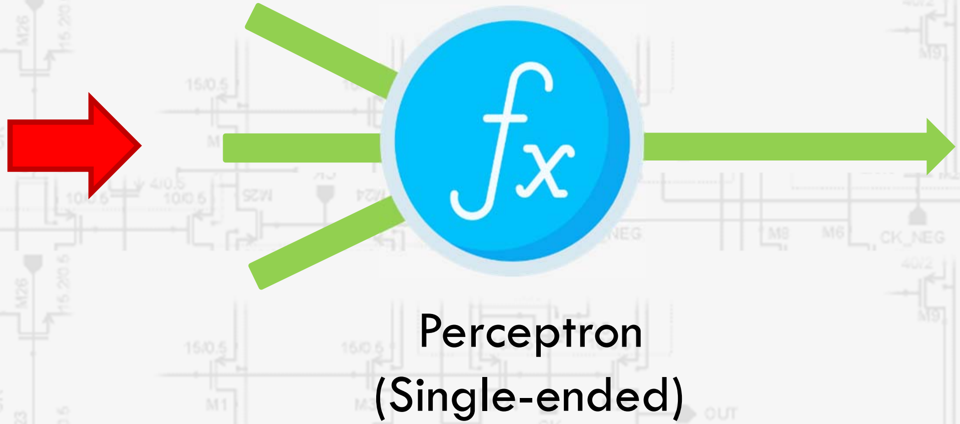


- When the voltage potentials received by the neuron exceed the active threshold, the neuron will be activated and propagates the information.

Biological Inspiration

- Let's see the neuron as a **function** that can be activated by exceeding a threshold value of the signal, which is the so-called **perceptron**.





- The concept matches up with the input connection functionality performed by dendrites in the biological neuron and the summation functionality provided by the soma.

Perceptron

- Connection weight

Weight (W)

  - Weights on connections in a neural network are coefficients that scale (amplify or minimize) the input signal to a given neuron in the network.

- Bias

B

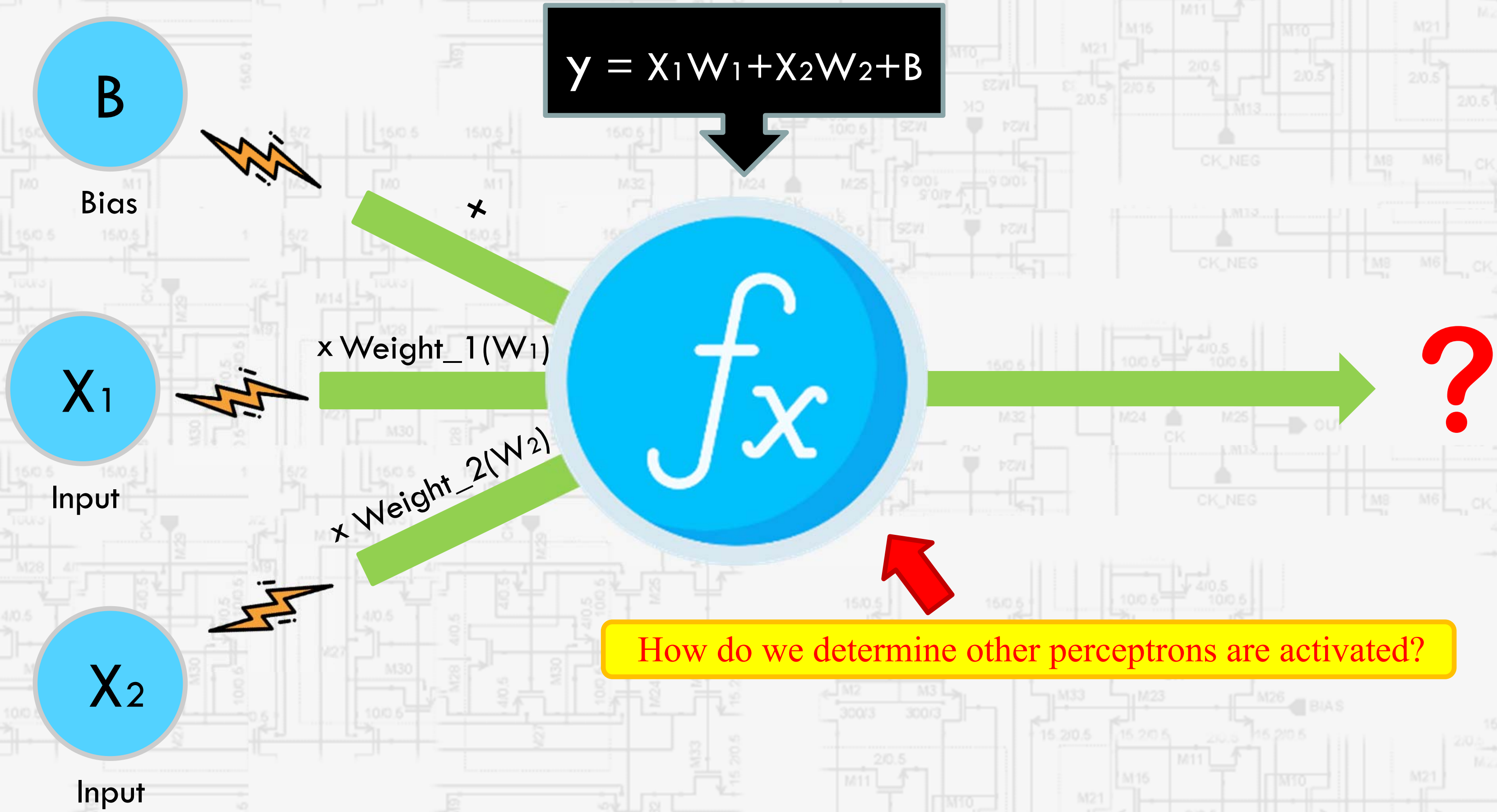
  - Biases are scalar values added to the input to ensure that at least a few nodes per layer are activated regardless of signal strength.

- Activation function

f_x

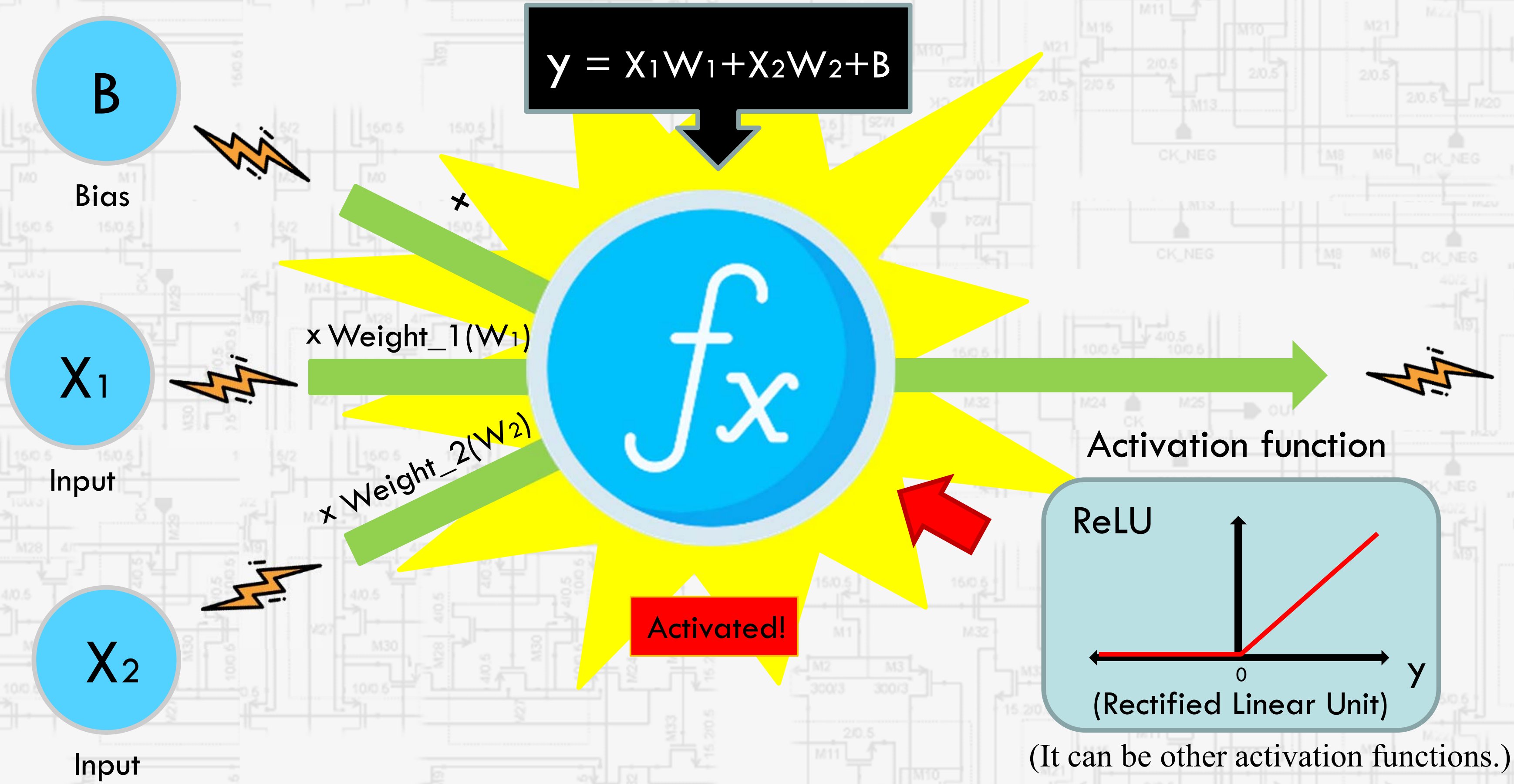
  - The function that governs the artificial neuron's behavior. When an artificial neuron passes on a nonzero value to another artificial neuron, it is activated.

Perceptron



- The perceptron **imitates the functionality of the neuron**, which receives information from other perceptron and propagates the message when it's activated.

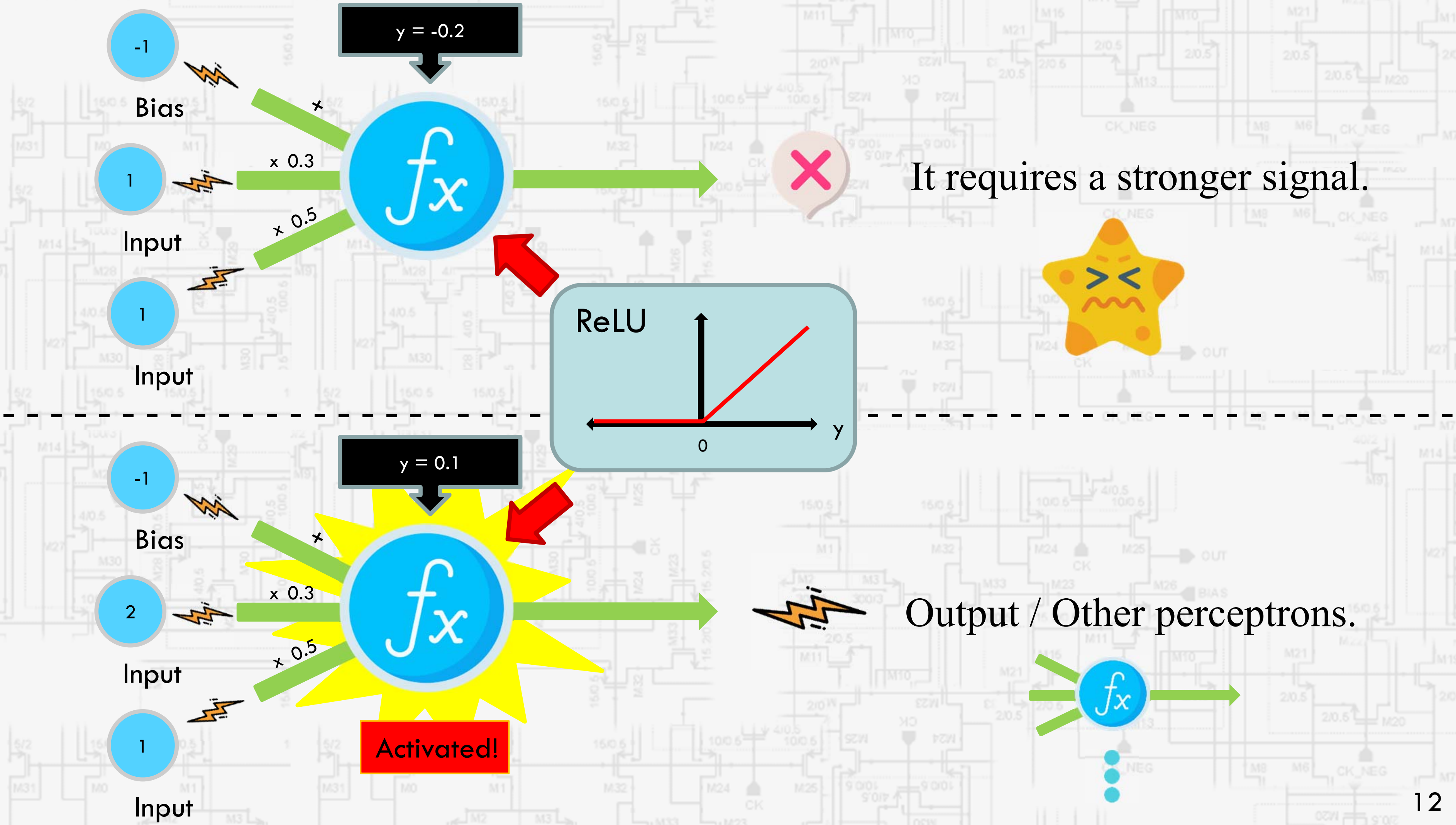
Perceptron



(It can be other activation functions.)

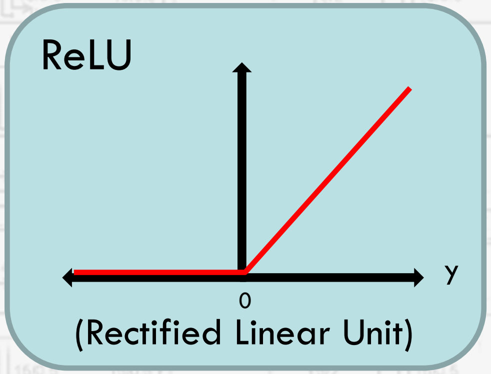
- The perceptron itself is the activation function that determines whether it is activated. Note that you can choose the activation function to suit your purpose.

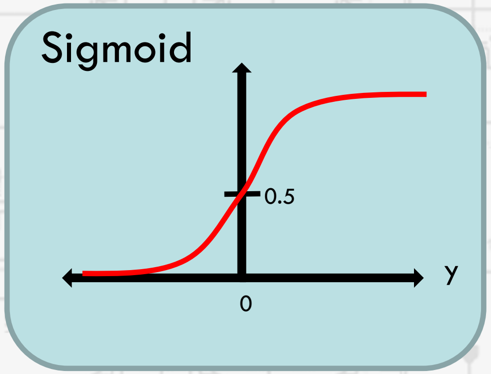
Perceptron

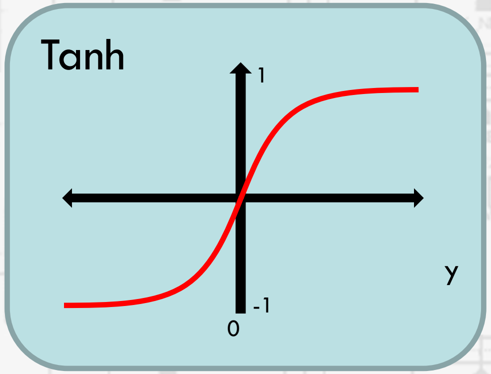

y = -0.2
-1
Bias
x
x 0.3
1
Input
x 0.5
1
Input
It requires a stronger signal.
ReLU
y
0
y = 0.1
-1
Bias
x
x 0.3
2
Input
x 0.5
1
Input
Activated!
Output / Other perceptrons.

Activation Function

- Common Activation Functions:







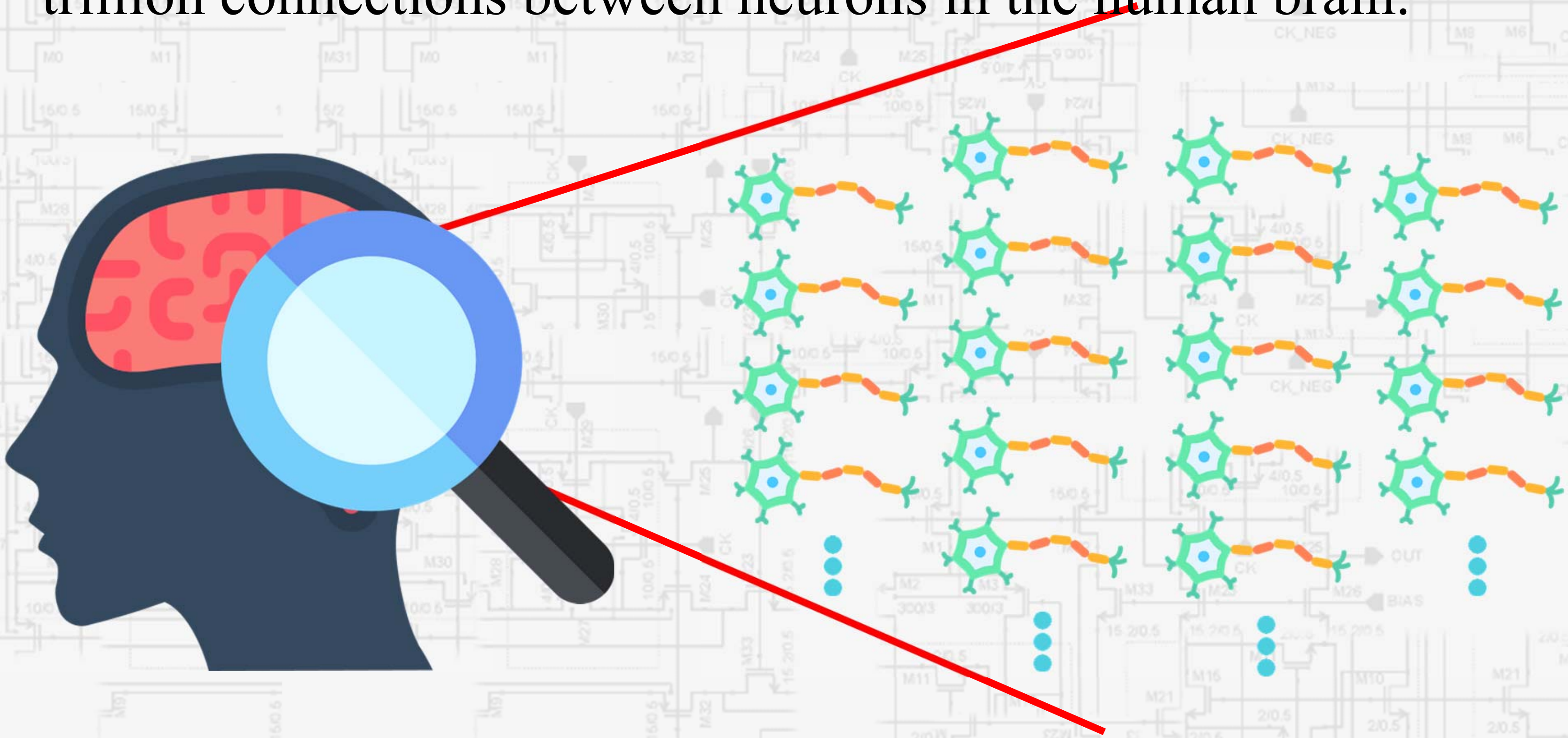
- All the activation functions must be **nonlinear.**

- If neural networks exploit linear function as an activation function, adding layers becomes useless work. Even the deeper network cannot achieve better performance.

- There are more activation functions such as LeakyReLU, Maxout, ELU, SELU, softplus, and so on...

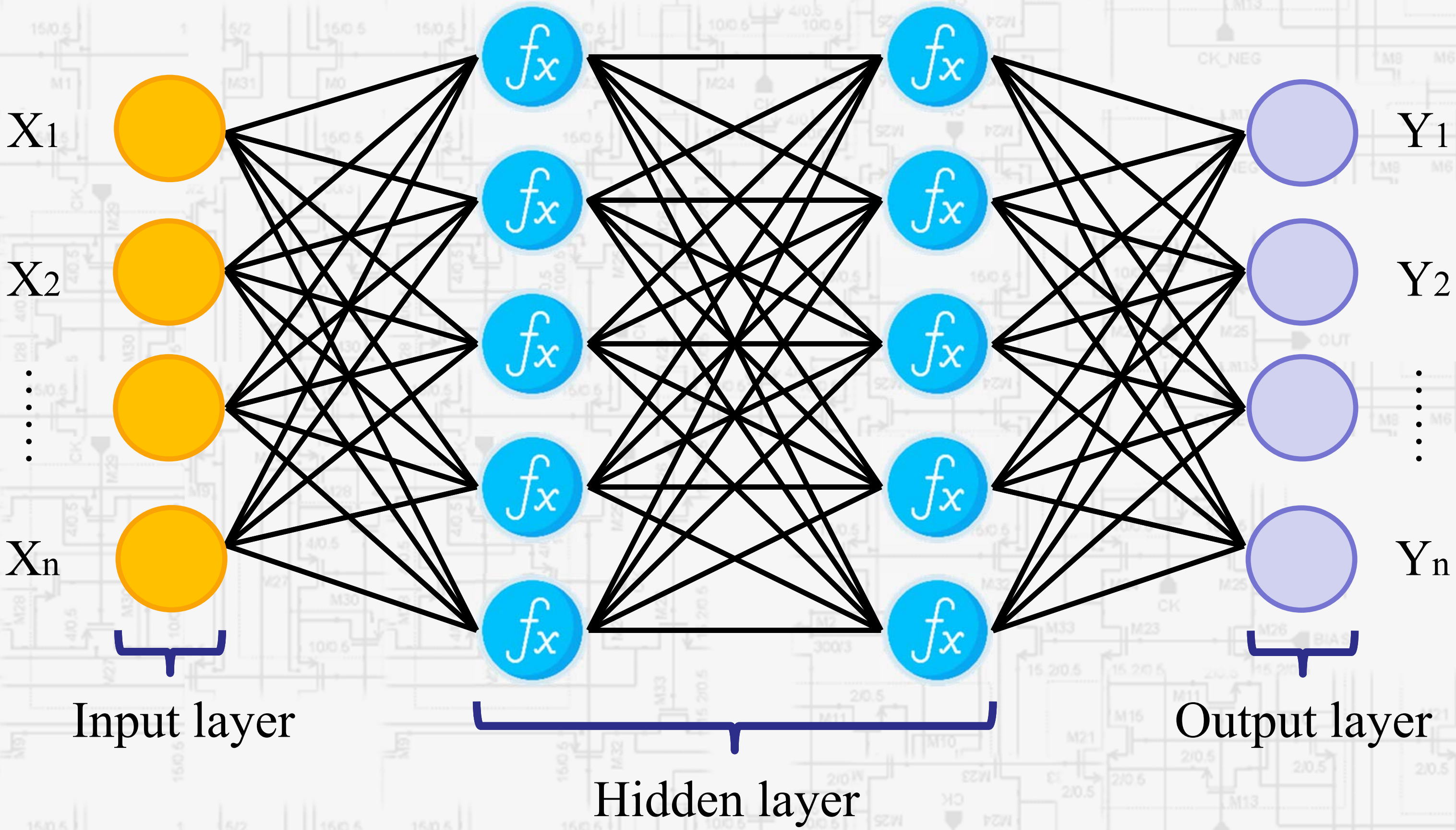
Neural Networks

- Researchers conservatively estimate there are more than 500 trillion connections between neurons in the human brain.



- What if we replace all the neurons with perceptrons?

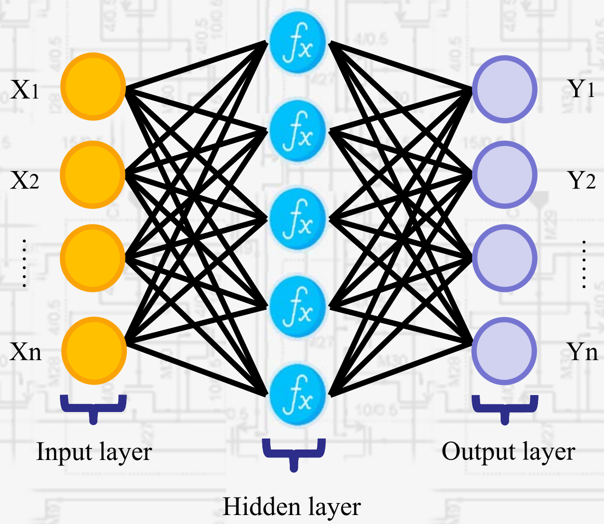
Neural Networks



Neural Networks

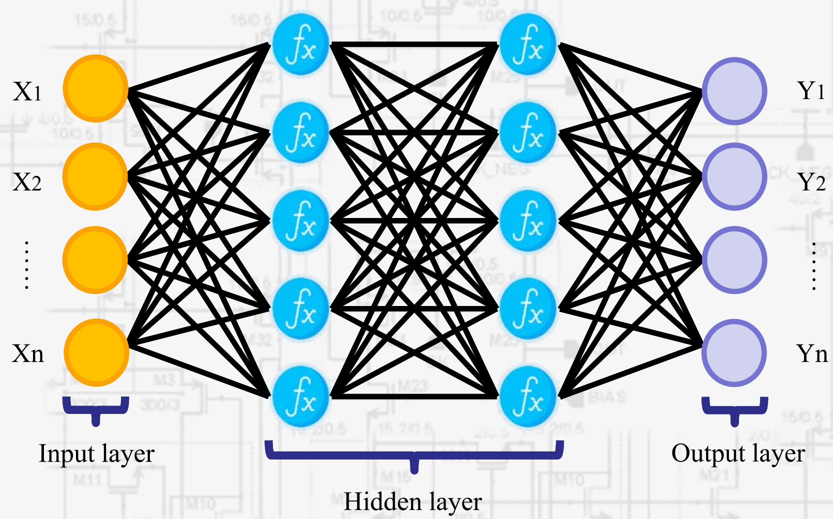
- In this course, we will define deep learning as neural networks with a large number of parameters and layers in fundamental network architectures.

Simple Neural Network



Number of layers: 2

Deep Neural Network



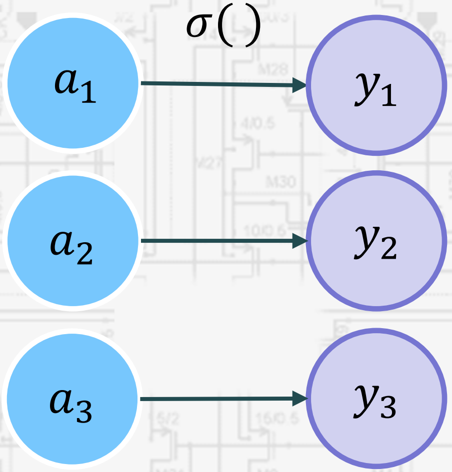
Number of layers: > 2

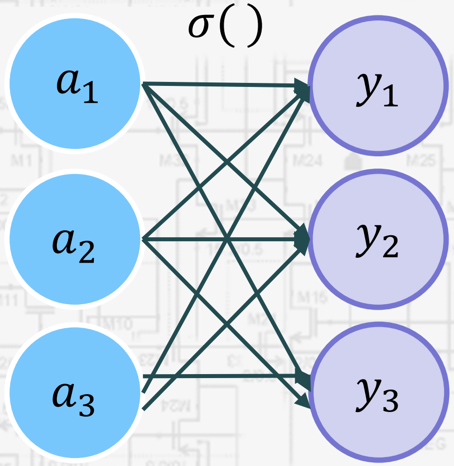
Output Layer

- According to applications of the neural network, we will exploit different output layers to fit the problem we want to solve.
- Regression : Identity function
  - A function that always returns the same value that was used as its argument.
- Classification : Softmax function

Identity function : $y_k = a_k$

Softmax function : $y_k = \frac{\exp(a_k)}{\sum_{i=1}^{n} \exp(a_i)}$





Softmax Function

- Each output of the softmax function is in range (0,1), and the sum of them is 1.
- As the result of the property, the outputs of softmax can be regards as a probability.
- To sum up, the softmax function converts the results of the neural network to a probability distribution.
- Softmax function only exploits in the training phase, why?

Softmax function :

$$y_k = \frac{exp(a_k)}{\sum_{i=1}^{n} exp(a_i)}$$

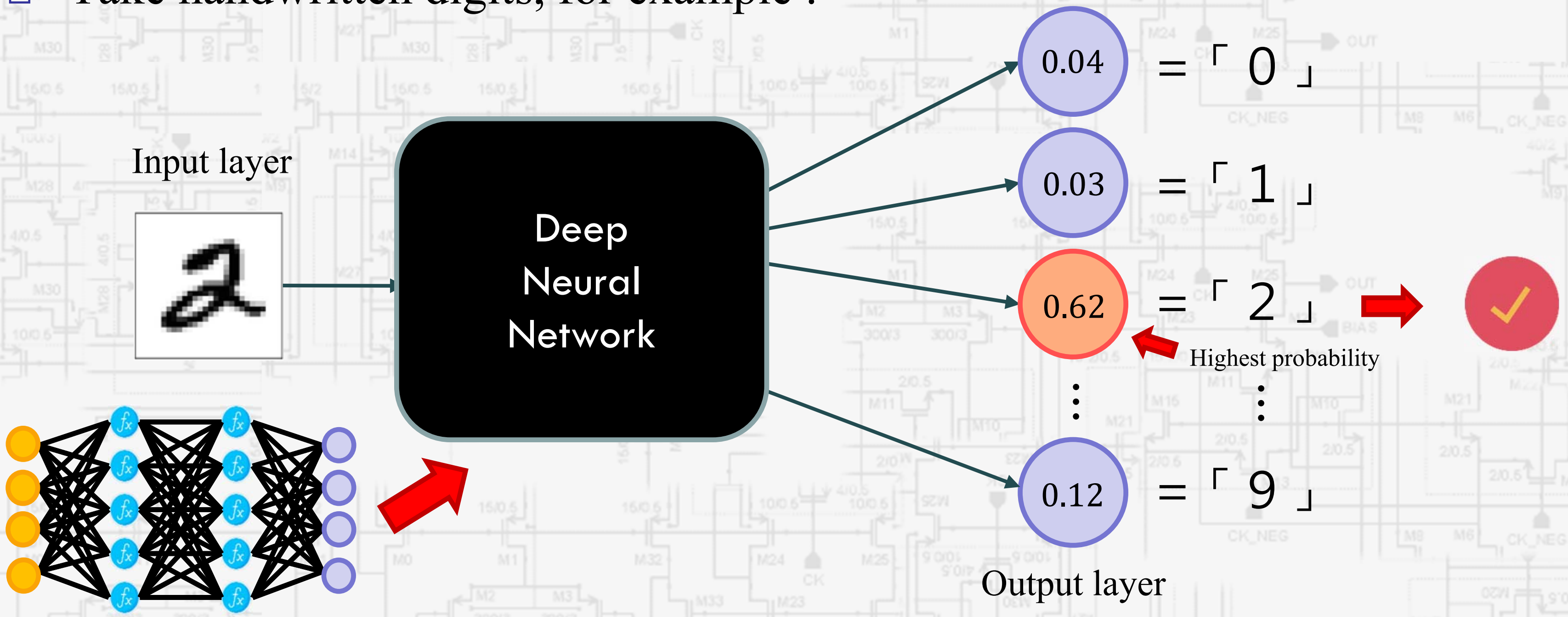
$$\begin{bmatrix} 1.2 \\ 0.9 \\ 0.4 \end{bmatrix} \xrightarrow{\text{Softmax}} \begin{bmatrix} 0.46 \\ 0.34 \\ 0.20 \end{bmatrix}$$

Probability distribution

Softmax Function

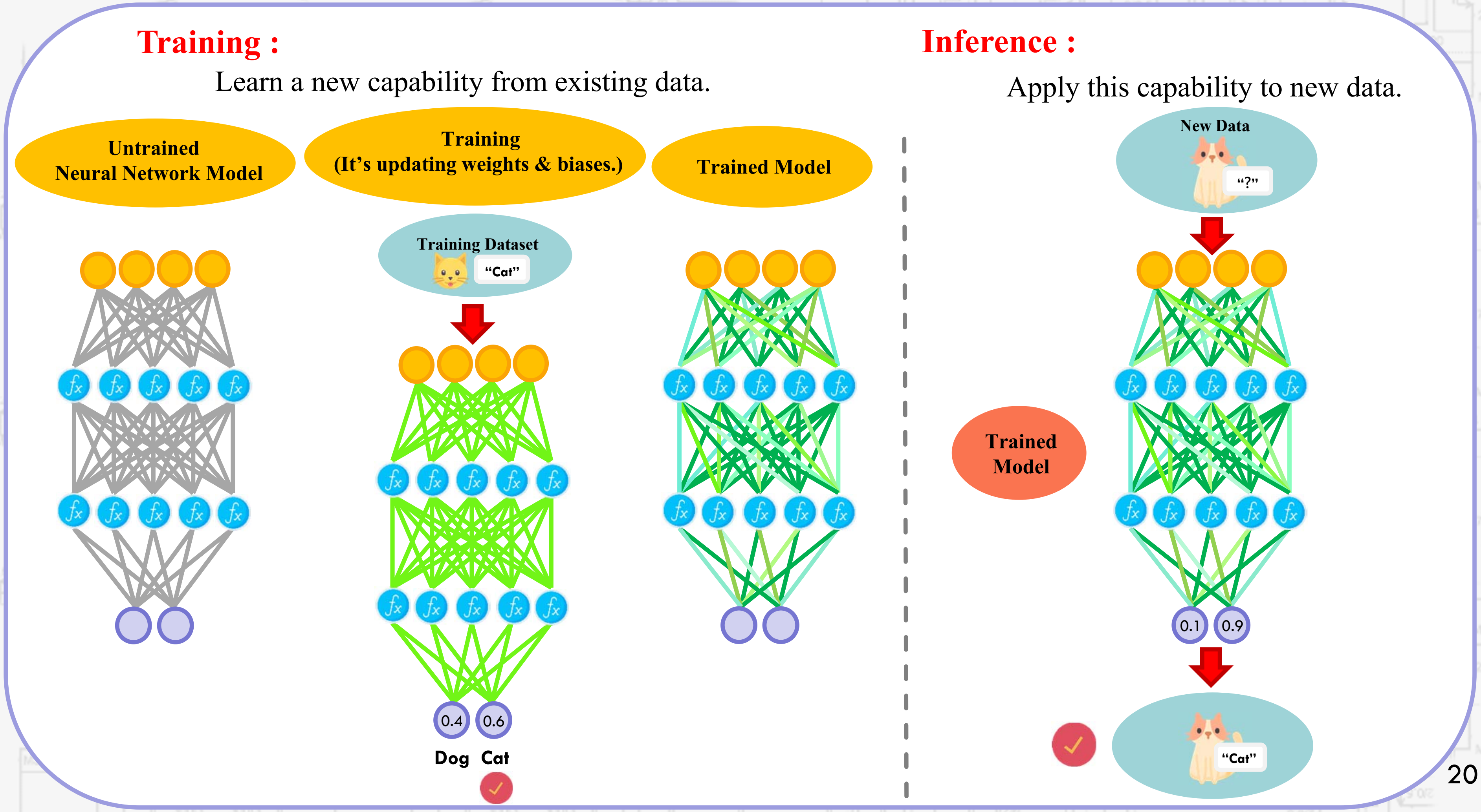
- In classification, the number of neurons in the output layer is equal to the number of categories you want to classify.
- An output represents the **probability** of a category to which an input might belong.
- Take handwritten digits, for example :

Softmax function : $y_k = \frac{\exp(a_k)}{\sum_{i=1}^{n} \exp(a_i)}$



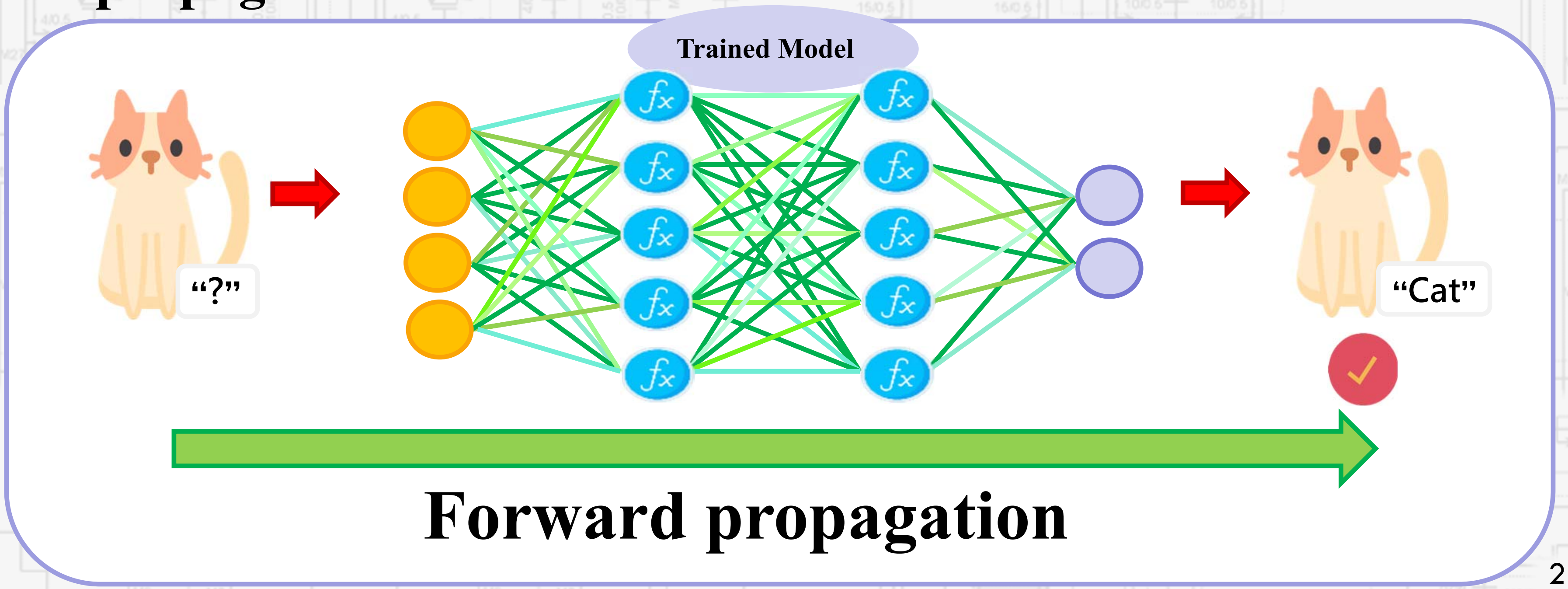
Two Phases of Deep Learning

- There are two phases in deep learning :



Forward Propagation

- After a neural network is trained, it is deployed to run inference - to classify, recognize, and process new inputs without updating parameters.
- The inference(predict) processing is also known as "**forward propagation.**"



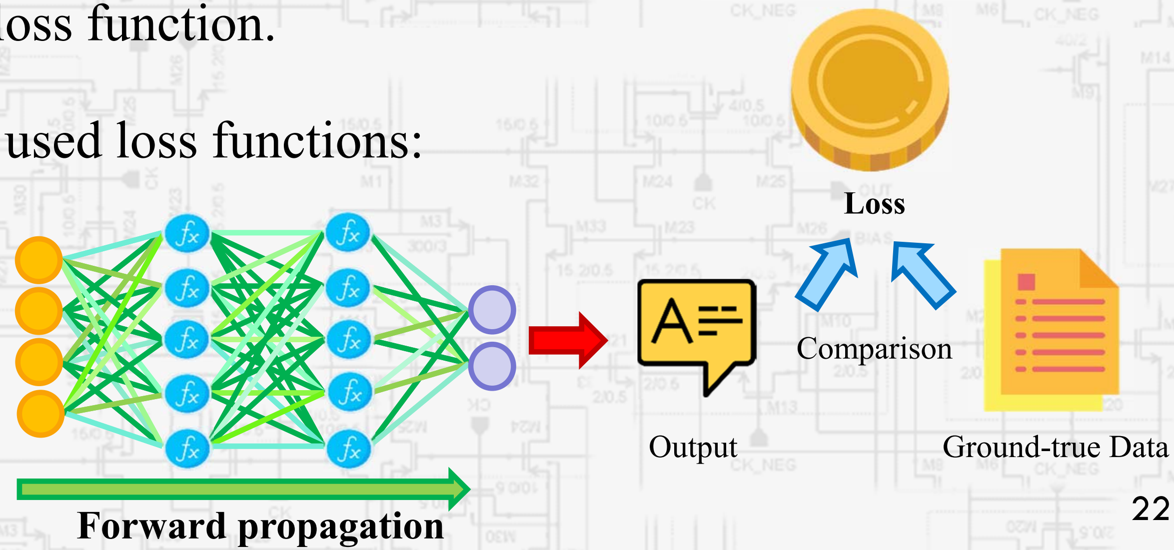
Loss Function

- Before mentioning backward propagation, we have to know about loss function, **gradient**, and **gradient descent** first.

- Loss function is a criterion that evaluates the performance of neural networks. It qualifies the agreement between the predicted output and the ground truth output.

- Neural networks calculate the **loss** of training data and find a set of parameters at the minimum value of loss function.

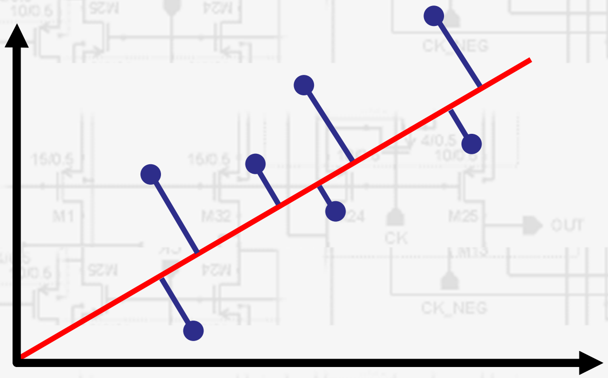
- There are two commonly used loss functions:
  - Mean square error.
  - Cross-entropy error.



Mean Square Error

- Mean square error (MSE) is a measure of the quality of an estimator : The difference between the estimators and what is estimated, is always non-negative, and values closer to zero are better.

$$E = \frac{1}{k}\sum_k (y_k - t_k)^2$$



$$t_k = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \qquad y_k = \begin{pmatrix} 0.4 \\ 0.6 \end{pmatrix} \Rightarrow E = 0.16$$

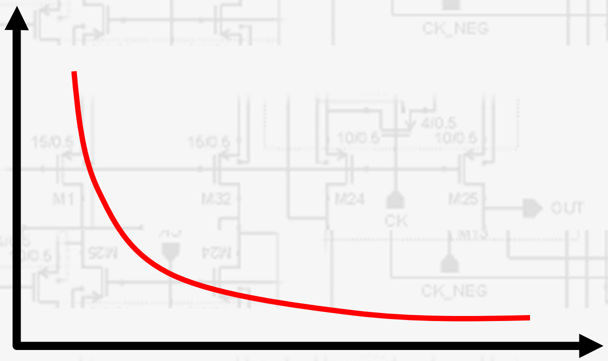
Training data (one-hot encoding)

Outputs of the network

Cross-Entropy

- Cross-entropy measures the difference between two probability distributions. If outputs approximate to corresponding labels, the result of cross-entropy is close to zero.

$$E = -\sum_k t_k \log y_k$$



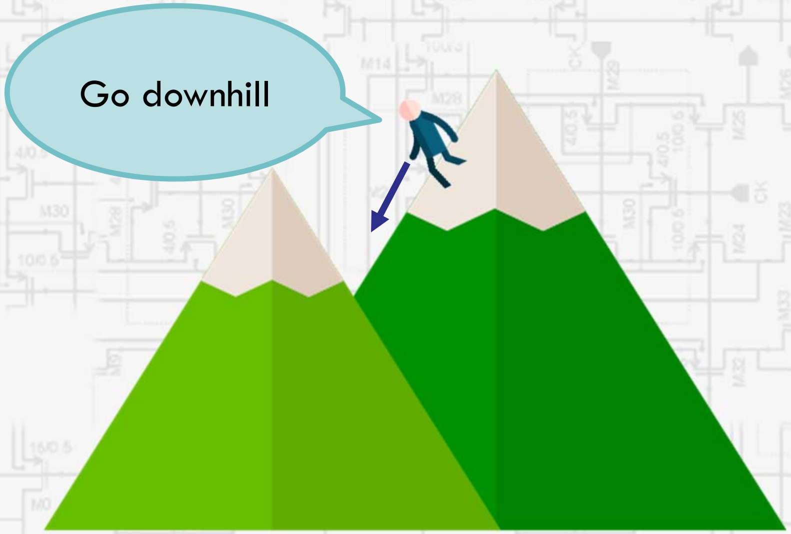
$$t_k = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad y_k = \begin{pmatrix} 0.4 \\ 0.6 \end{pmatrix} \Rightarrow E = 0.736$$

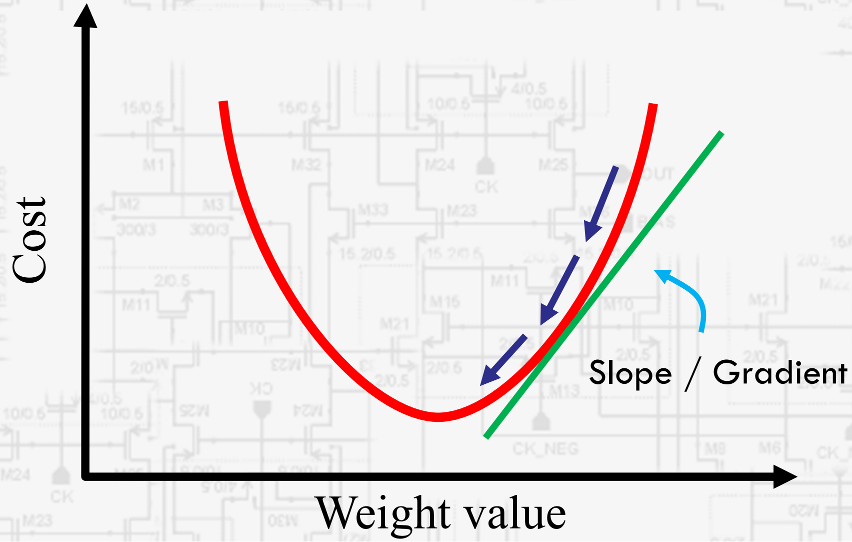
Training data (one-hot encoding)

Outputs of the network

Gradient Descent

- Neural networks will find the best solution of parameters in the training phase while **minimizing the loss function**.
- In most cases, these parameters cannot be solved analytically, but they can be approximated well with iterative optimization algorithms like gradient descent.
- If we want to **minimize the loss function**, the parameters are updated to the negative direction of differential value (**gradient or slope**).

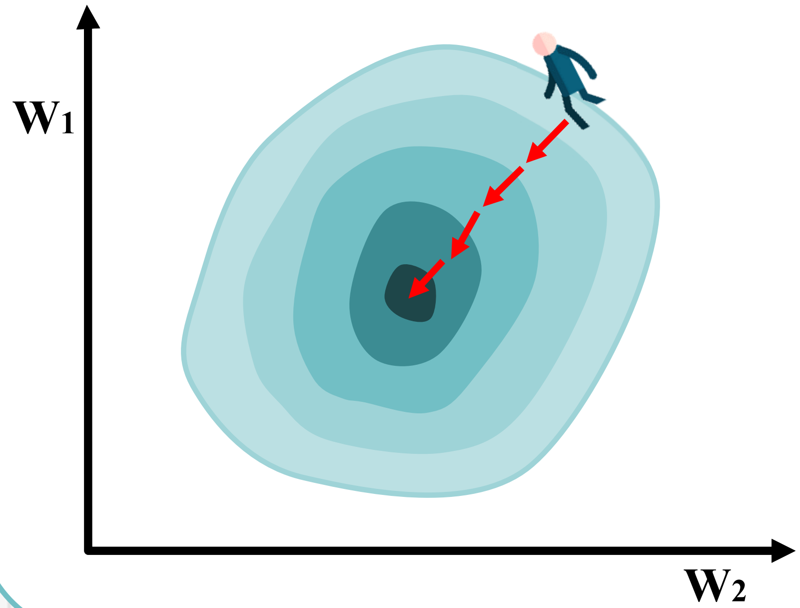




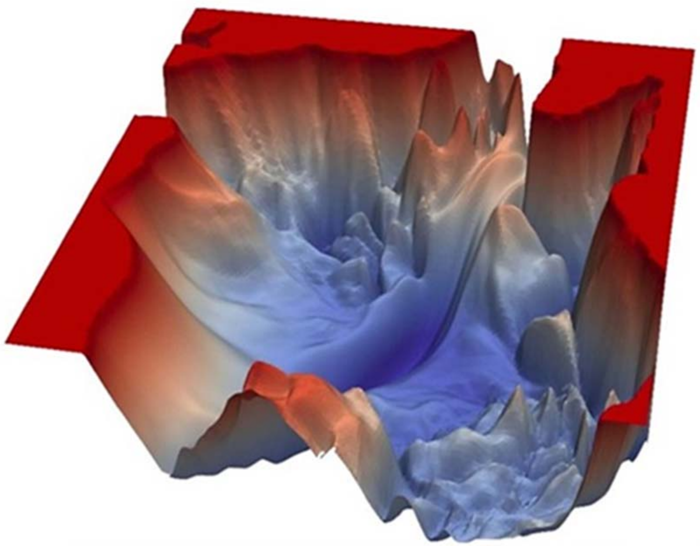
Gradient Descent

- Gradients in deep learning can be calculated by : $\frac{\partial \boldsymbol{L}}{\partial \boldsymbol{W}}$
  - $\boldsymbol{L}$ is the loss function.
  - $\boldsymbol{W}$ is all weights in a neural network.
- If there are only two weights in loss function :

Loss Contour Line



The real condition may be :



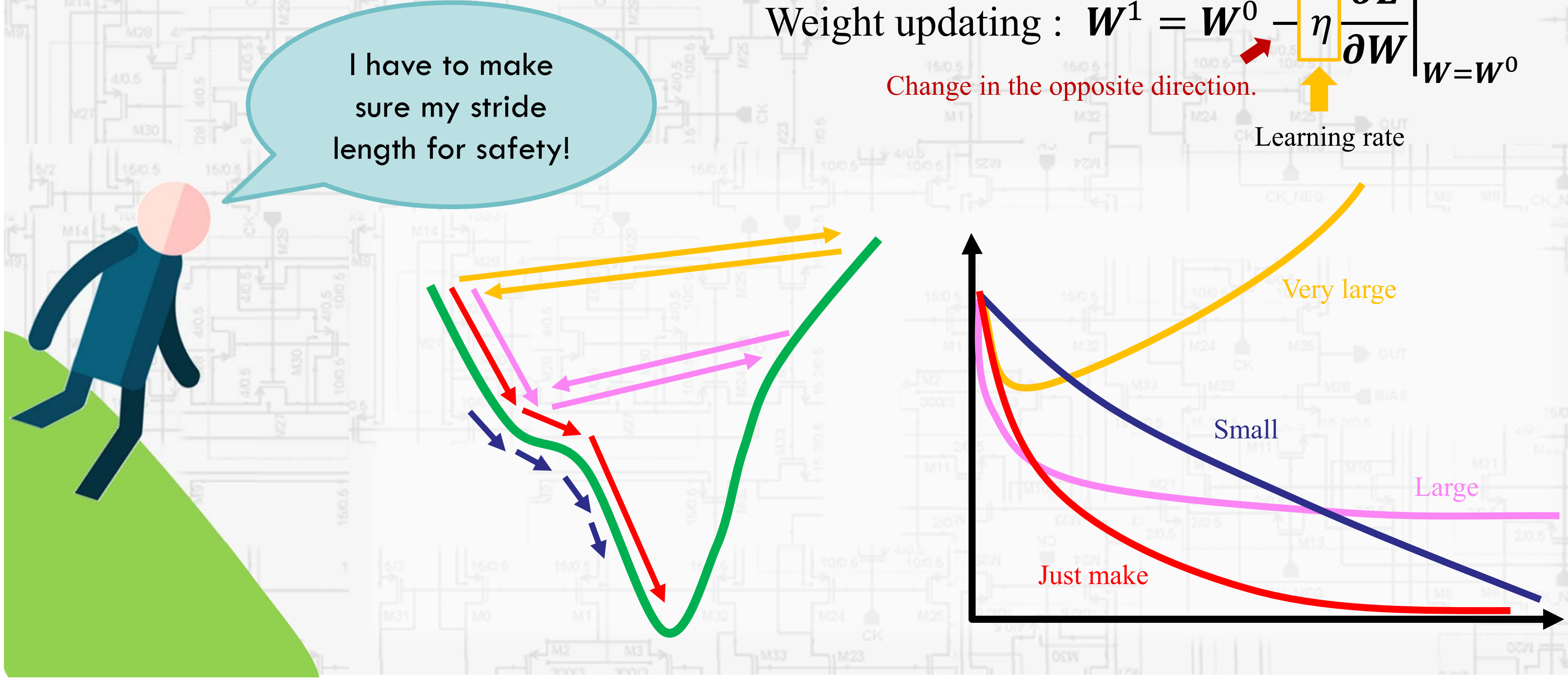
Learning Rate

- **Learning rate** decides how far the step is to the next position on the loss function.
- It is also a kind of hyper-parameter determined by humans. Thus we have to set the value carefully.

Weight updating : $\boldsymbol{W}^1 = \boldsymbol{W}^0 - \boxed{\eta} \frac{\partial \boldsymbol{L}}{\partial \boldsymbol{W}}\bigg|_{\boldsymbol{W}=\boldsymbol{W}^0}$

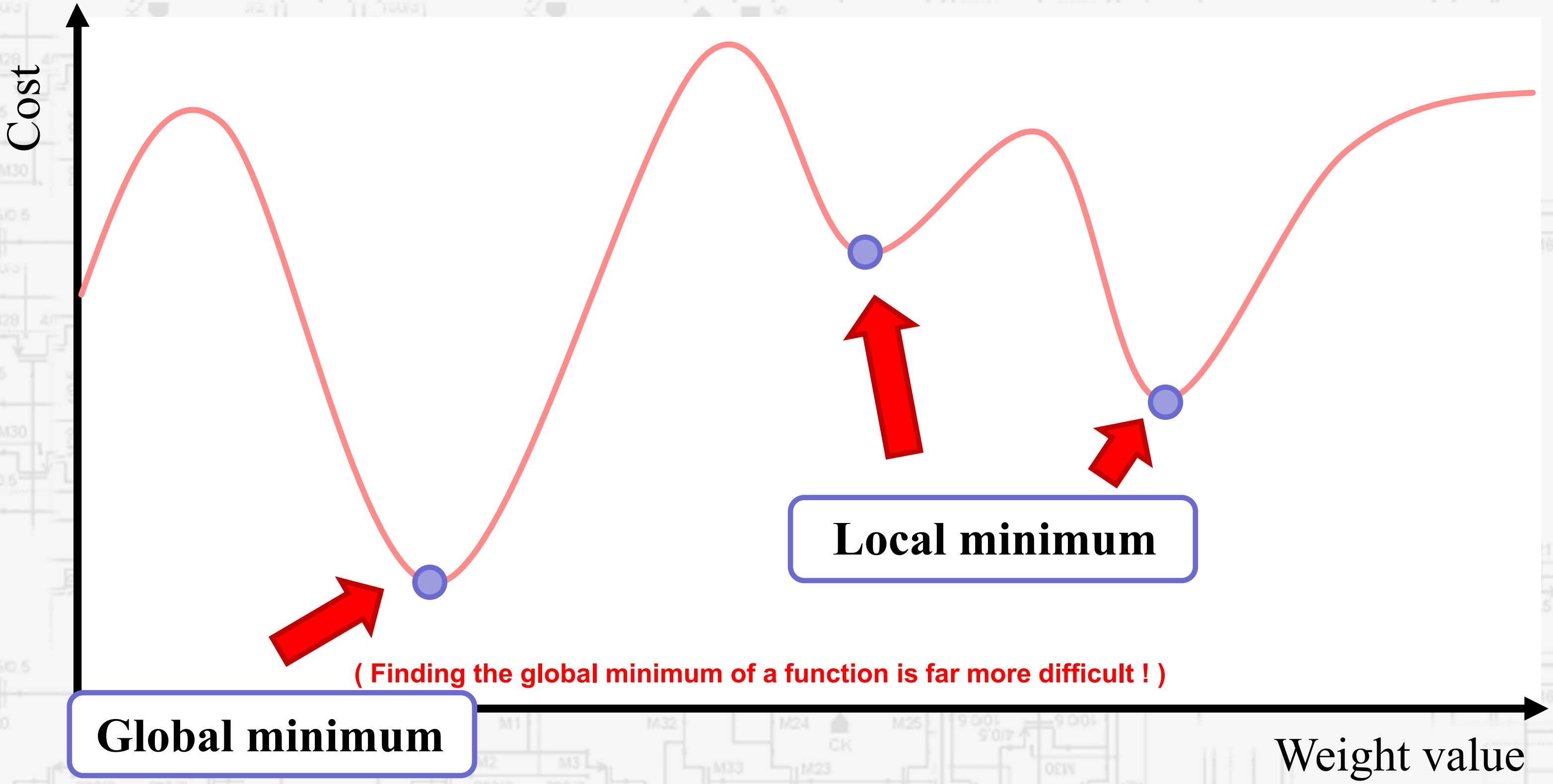
Change in the opposite direction.

Learning rate



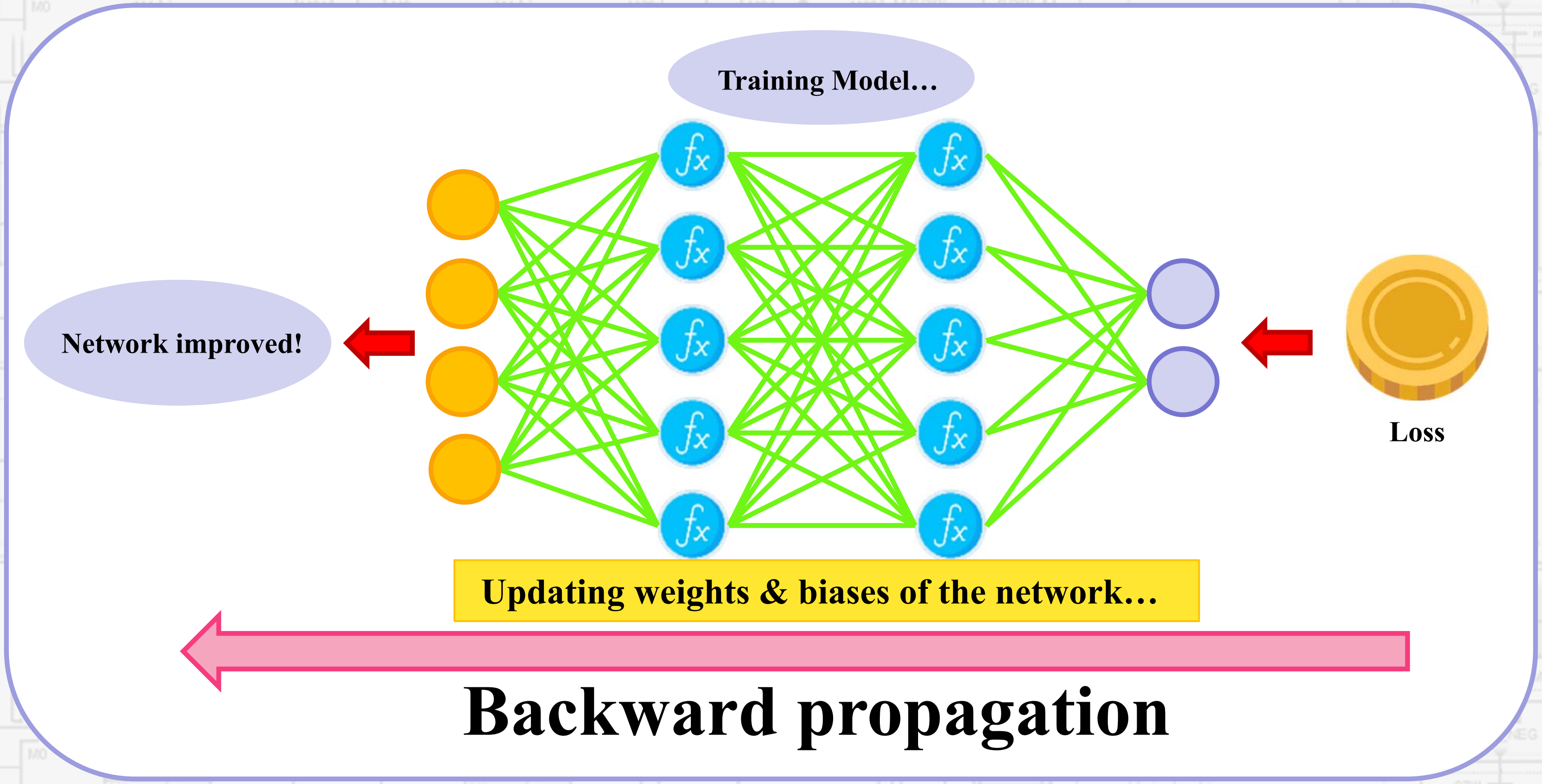
Critical Point

- A **local minimum** of a function is a point where the function value is smaller than the nearby points.
- A **global minimum** is a point where the function value is smaller than at all other feasible points.



Backward Propagation

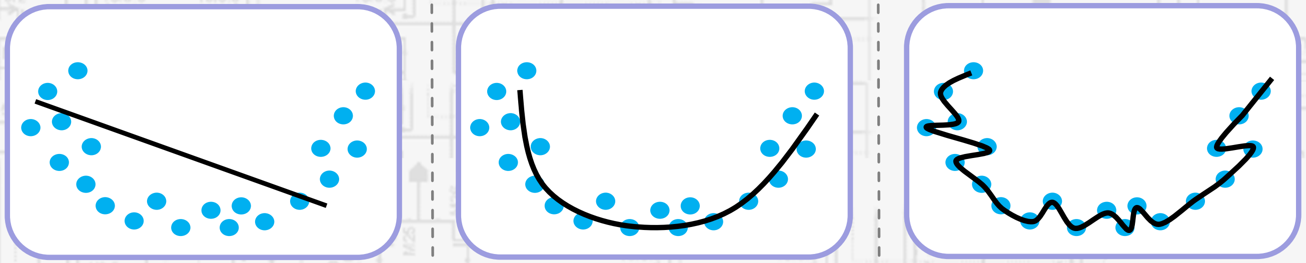
- When the loss function has been calculated. We can apply it to **backward propagation**, utilizing the gradients and learning rate to **update** the weight.



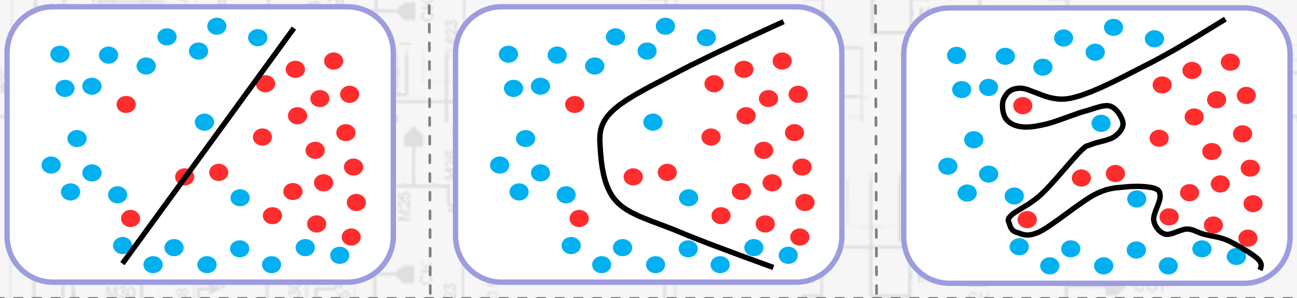
Overfitting & Underfitting

	Underfitting	Just make	Overfitting
Symptoms	· High training error. · Training error close to testing error. · High bias.	· Training error slightly lower than testing error.	· Very low training error. · Training error much lower than test error. · High variance.

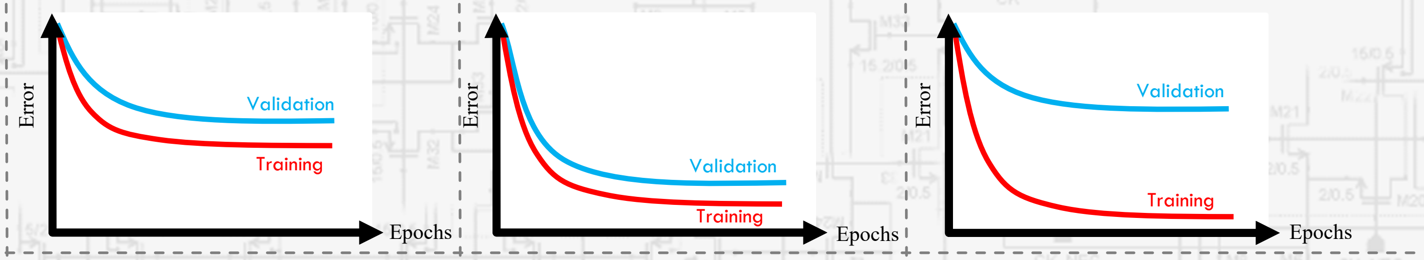
Regression



Classification

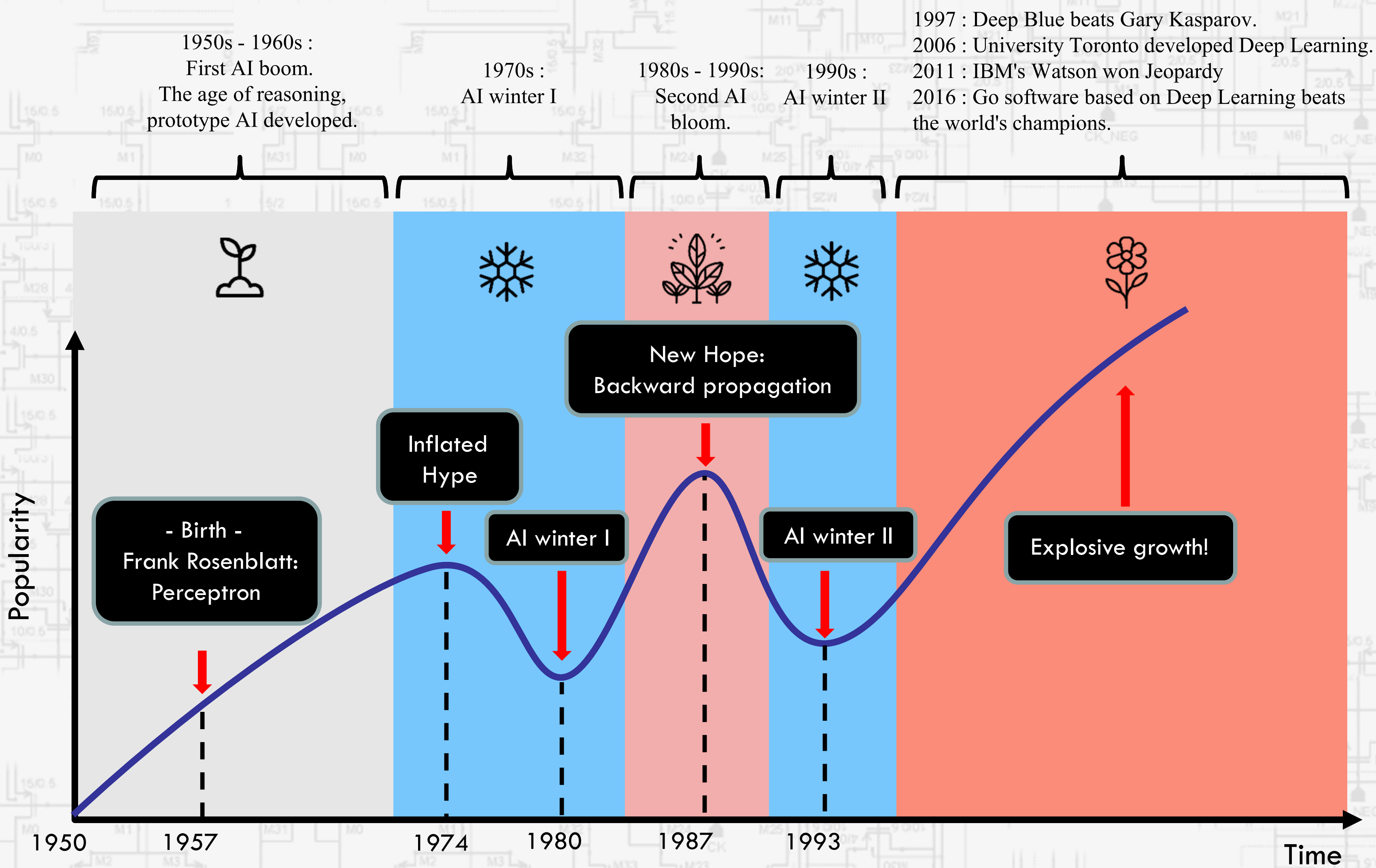


Deep learning

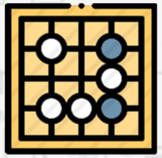


	Underfitting	Just make	Overfitting
Possible remedies	· Complexify model. · Add more features. · Train longer.		· Perform regularization. · Get more data.

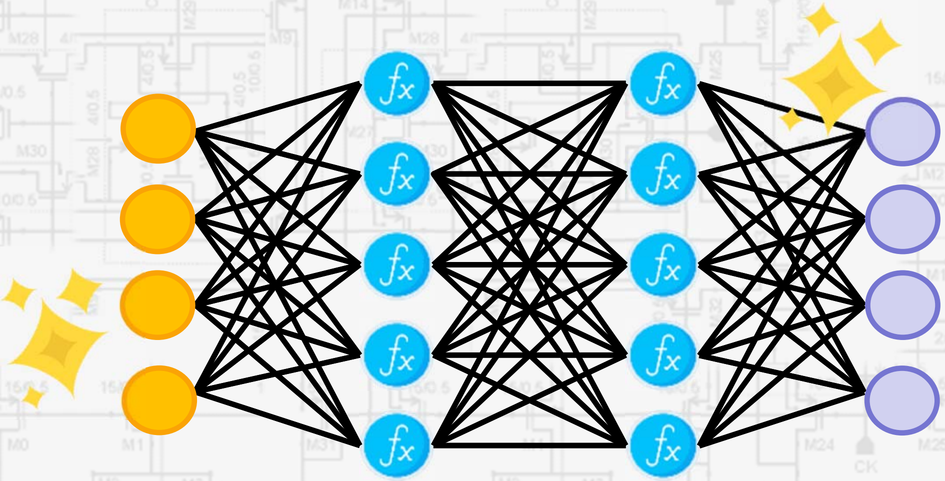
The History of AI

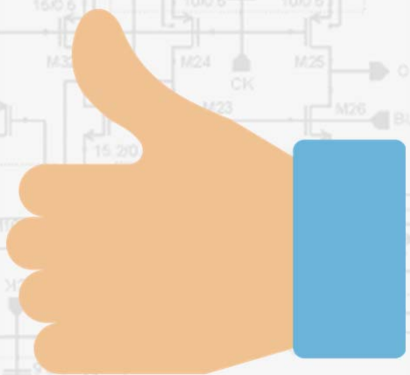


What Can Deep Learning Do?

- Image recognition
  - Deep learning can reach a high accuracy that humans cannot accomplish.
- Game
  - AlphaGo 
  - The computer can learn by itself and even better than humans. 

- There are more and more applications of deep learning.

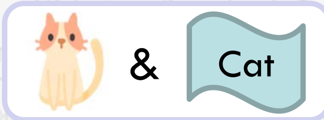




Learning Algorithms

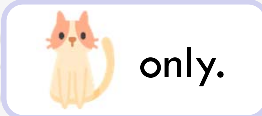
Learning Algorithms

Supervised Learning



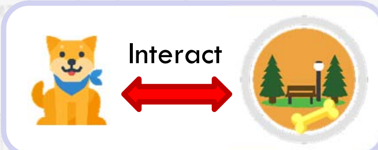
Supervised learning requires a **labeled dataset.**
The network can learn from it to make inferences or predictions of the problem.

Unsupervised Learning



Unsupervised learning is the opposite of supervised learning.
There is **no labeled dataset** in unsupervised learning.

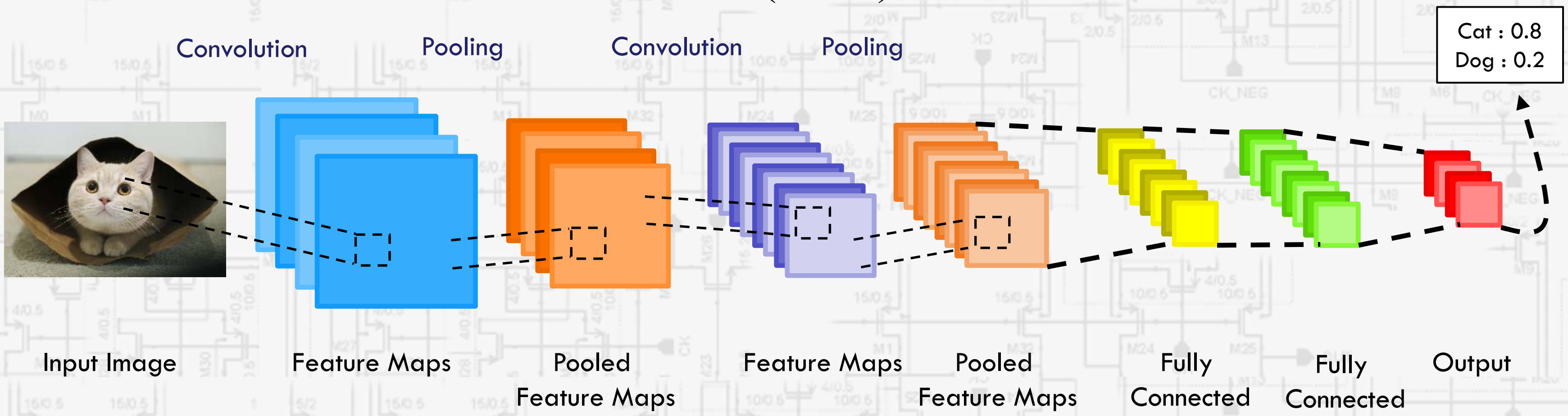
Reinforce Learning



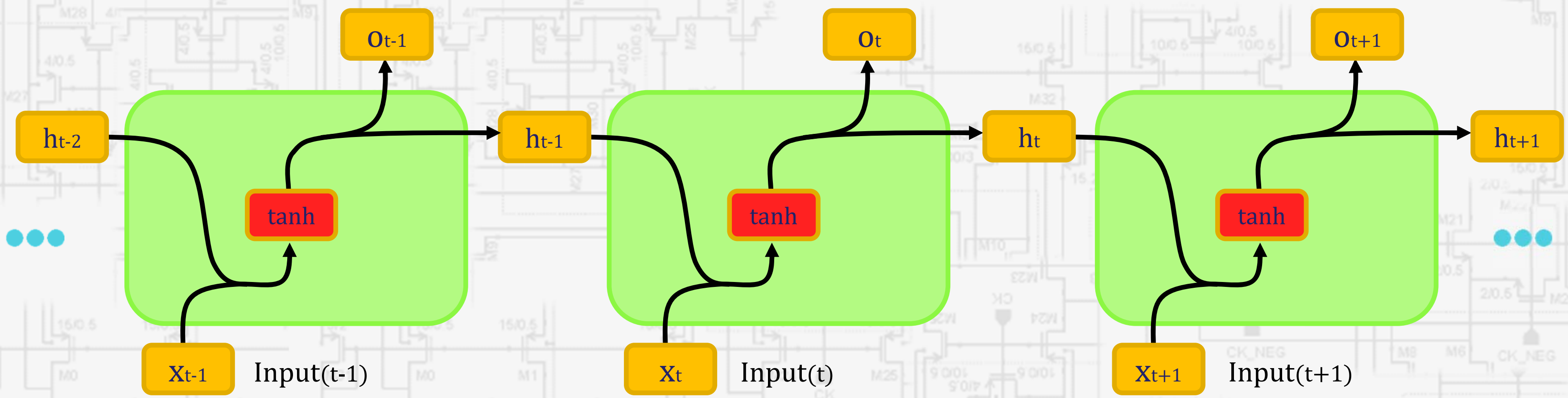
Reinforce learning model will learn to **react to the environment** by itself, with a system composed of **reward, state, and action**.

Basic Model of Neural Network

- Basic convolutional neural network (CNN) :

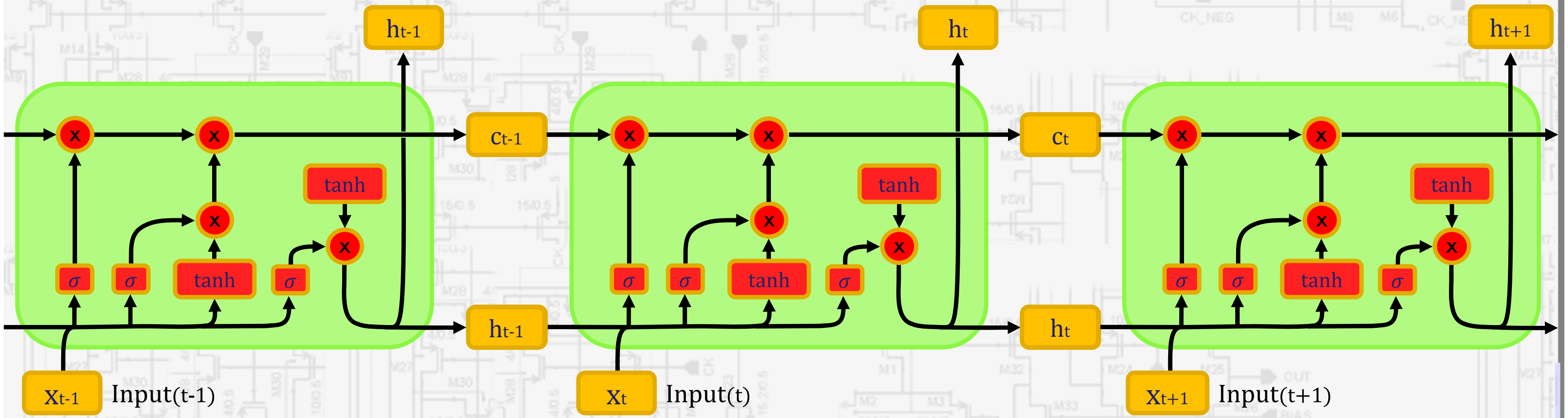


- Basic recurrent neural network (RNN) :



Advanced Model of Neural Network

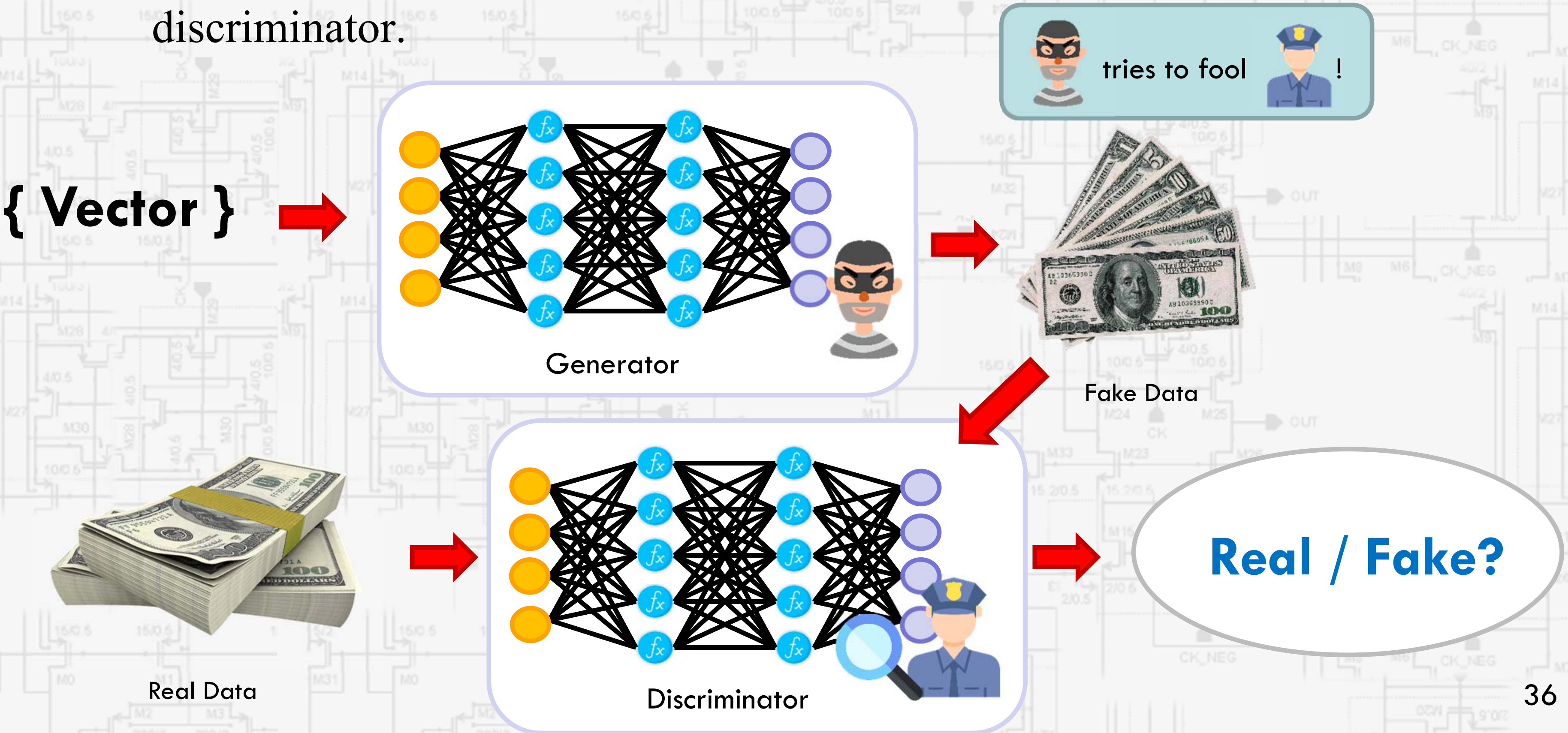
- Long short-term memory (LSTM) :
  - LSTM enables RNN to remember inputs over a long time.



  - It also solves the problem such as vanishing gradient and exploding gradient.

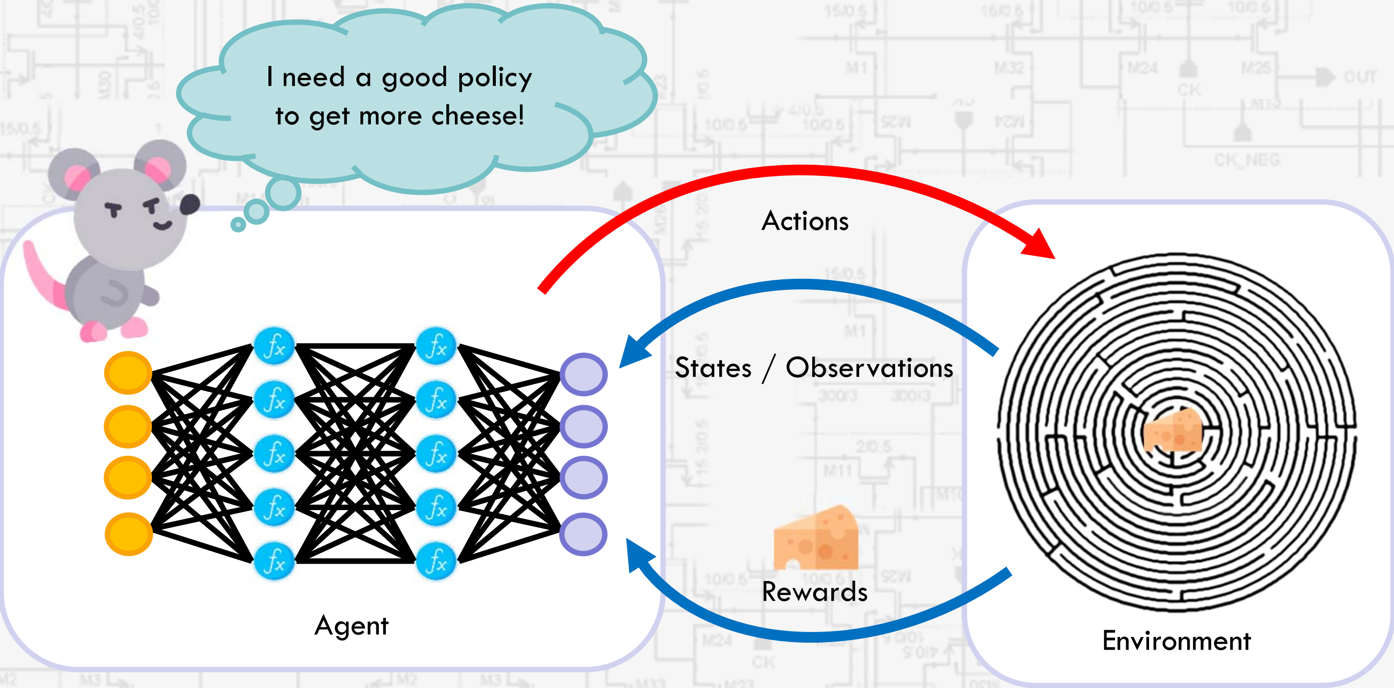
Advanced Model of Neural Network

- Generative adversarial network (GAN) :
  - GAN is a potential network that can generate image/voice/text data.
  - Basic GAN architecture includes two networks. The generator and the discriminator.



Advanced Model of Neural Network

- Deep Q network (DQN) :
  - The mission of DQN is to find an optimized **policy(strategy)** for winning more rewards.
  - In DQN, we will put the agent in the environment. It will learn better policy during interacting with the environment.



Applications

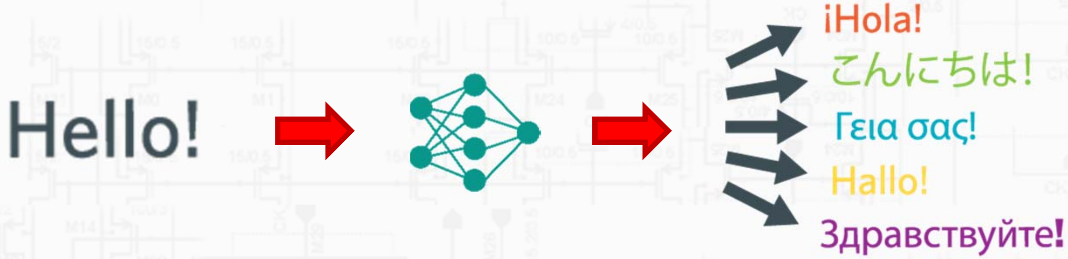
- Image segmentation :

- Object detection :

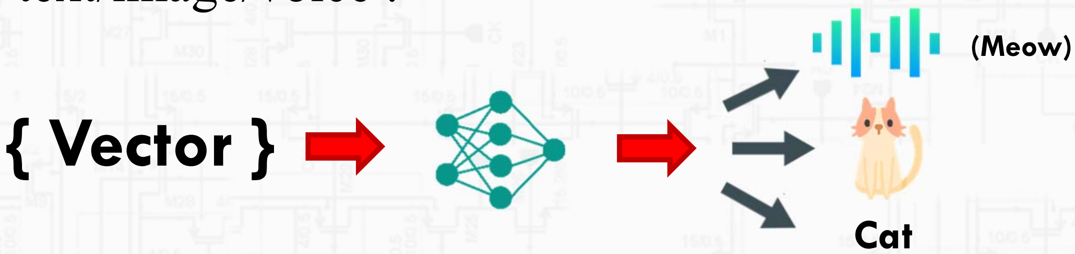
- Speech recognition :

Applications

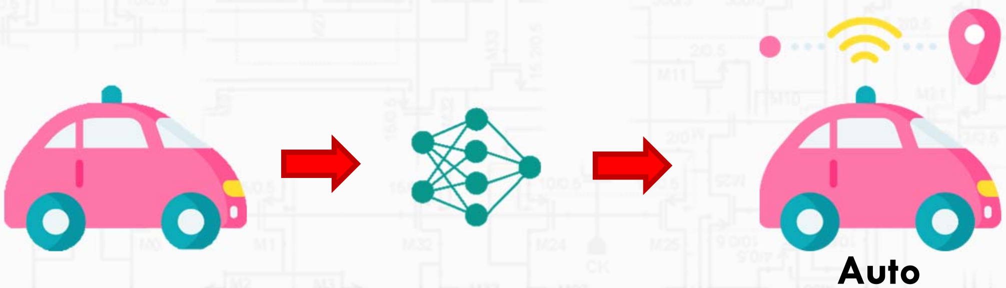
- Language translation :



- Generate text/image/voice :



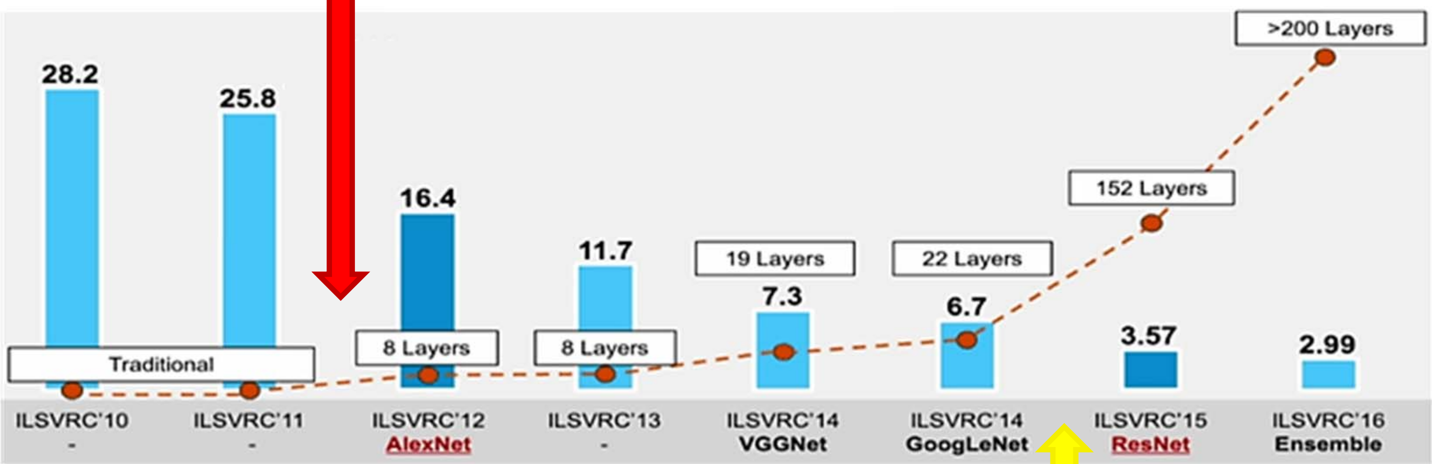
- Self-Driving System :



ILSVRC

- ImageNet Large Scale Visual Recognition Challenge.
- Deep models first perform good performance in commercial applications.

Era of deep learning is beginning.



Break through human recognition performance.

Conclusion

- **Biological Concept**
  - Deep Neural Networks were derived from the biological concept of the perceptron.
- **Variety of Deep Neural Networks**
  - Various architecture such as CNN, RNN, LSTM, GAN, DQN, and so on...
- **Application of Deep Neural Networks**
  - Image segmentation, object detection, speech recognition, etc.