

Welcome to

# Astar!

AI and Robotics: An Introduction by HKU Astar

*Astar's*  
*Rough Introduction to*  
**Machine Learning**

# Machine Learning



1. Artificial Intelligence and Machine Learning
2. One of the General Machine Learning Paradigms: Supervised Learning
3. Solving a Classification Task with the Linear Model
4. Neural Networks
  - a. Multilayer Perceptron (MLP)
  - b. Convolutional Neural Networks (CNN)
  - c. Recurrent Neural Networks (RNN)
5. Some Ideas and Fancy Models
  - a. Representation Learning
  - b. The Encoder-Decoder Architecture and Autoencoder
  - c. Generative Adversarial Network (GAN)
  - d. Reinforcement Learning
6. Research Areas
7. Limitations
8. Resources

# Machine Learning



1. Artificial Intelligence and Machine Learning
2. One of the General Machine Learning Paradigms: Supervised Learning
3. Solving a Classification Task with the Linear Model
4. Neural Networks
  - a. Multilayer Perceptron (MLP)
  - b. Convolutional Neural Networks (CNN)
  - c. Recurrent Neural Networks (RNN)
5. Some Ideas and Fancy Models
  - a. Representation Learning
  - b. The Encoder-Decoder Architecture and Autoencoder
  - c. Generative Adversarial Network (GAN)
  - d. Reinforcement Learning
6. Research Areas
7. Limitations
8. Resources

# Artificial Intelligence and Machine Learning

# Where is Artificial Intelligence?



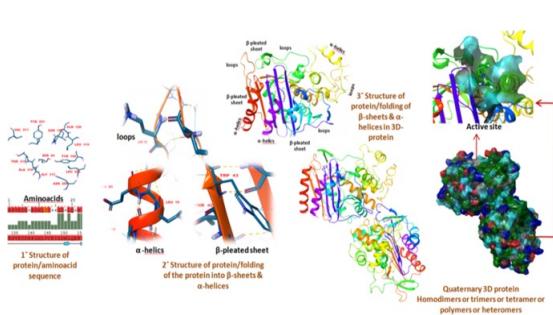
Alpha Go



Face Recognition



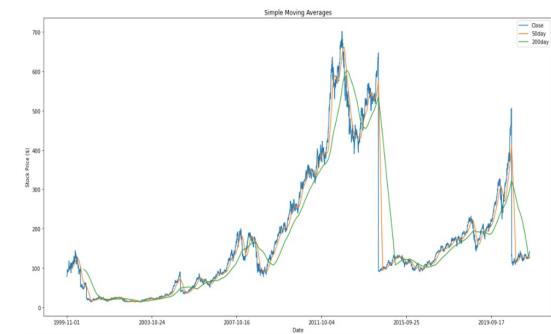
Recommender Systems



Drug Discovery



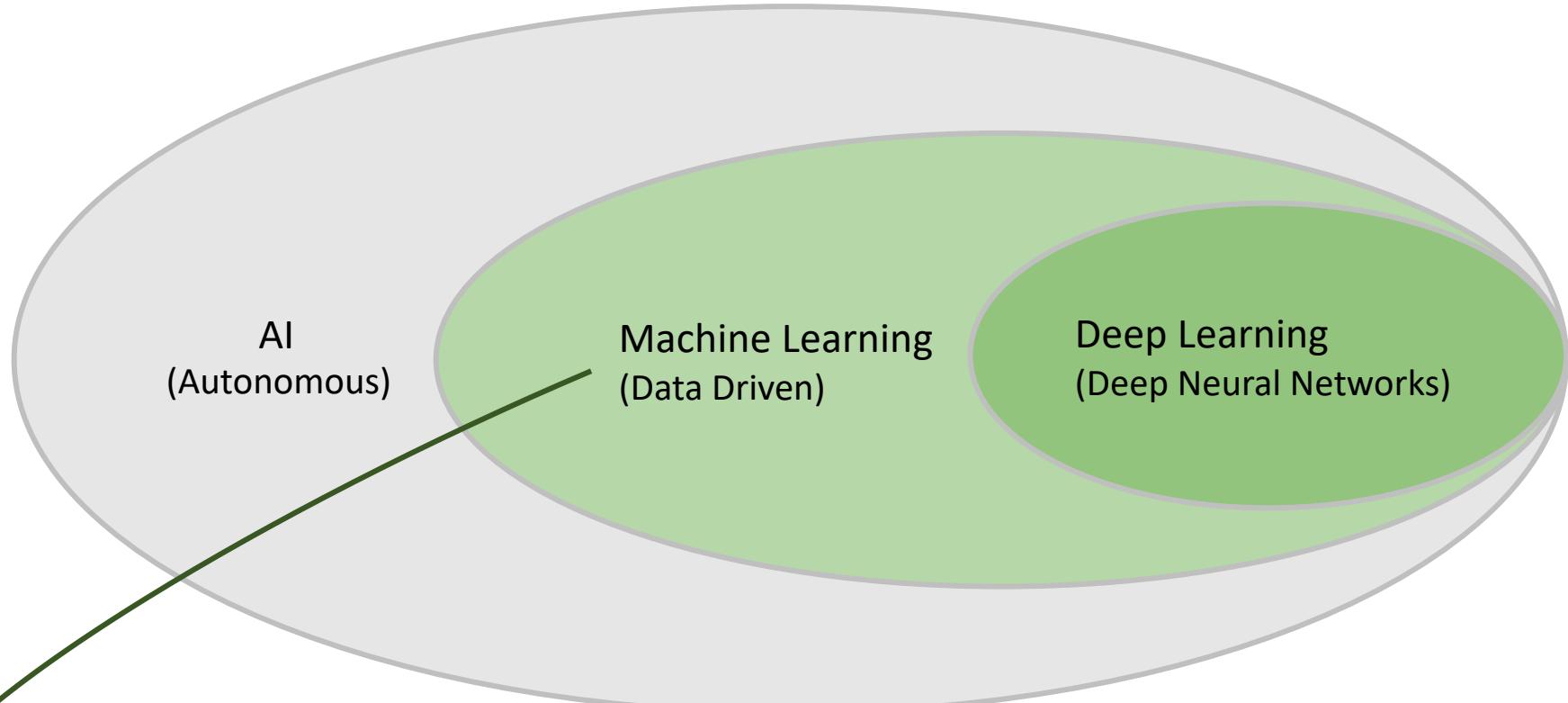
Autonomous Driving



Market Prediction

*Artificial Intelligence and Machine Learning*

# What is Machine Learning?



Field of study that gives computers the ability to learn without being explicitly programmed. *Arthur Samuel*

# Machine Learning



1. Artificial Intelligence and Machine Learning
2. One of the General Machine Learning Paradigms: Supervised Learning
3. Solving a Classification Task with the Linear Model
4. Neural Networks
  - a. Multilayer Perceptron (MLP)
  - b. Convolutional Neural Networks (CNN)
  - c. Recurrent Neural Networks (RNN)
5. Some Ideas and Fancy Models
  - a. Representation Learning
  - b. The Encoder-Decoder Architecture and Autoencoder
  - c. Generative Adversarial Network (GAN)
  - d. Reinforcement Learning
6. Research Areas
7. Limitations
8. Resources

# A General Machine Learning Paradigm : Supervised Learning

# How can a machine learn ?

数 据 Dataset

\* 本课关注：  
有监督学习  
Supervised Learning

模 型 Model

这是什么问题的数据？

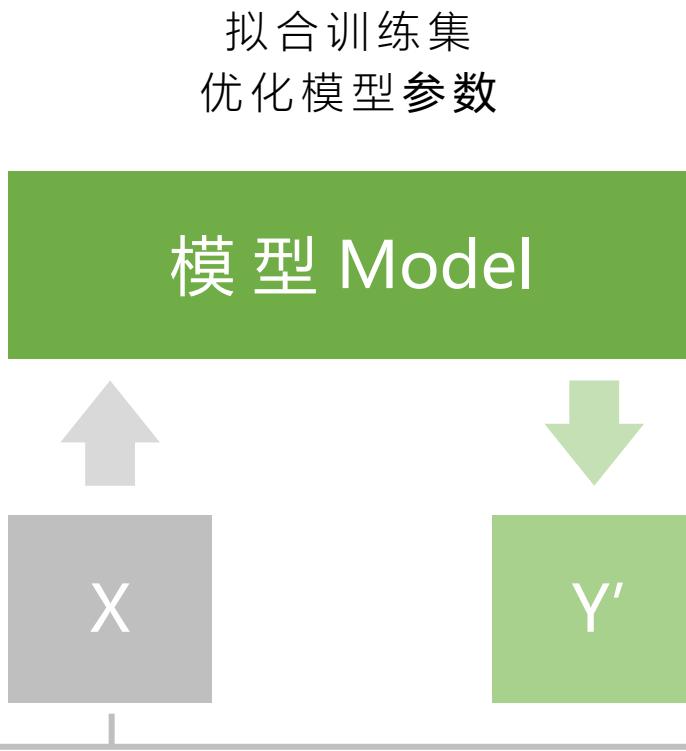
- 一个小车识别的数据集？（看一看）
- 一个西瓜评估的数据集？
- 一个推荐系统的数据集？

这个问题适合什么模型？

- 线性模型？( Linear Model )
- 多层感知机？( Multilayer Perceptron )
- 卷积神经网络？( Convolutional Neural Network )

# A General Machine Learning Paradigm : Supervised Learning

## How can a machine learn ?



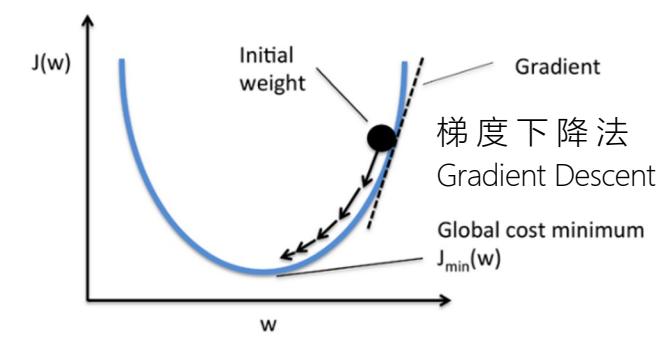
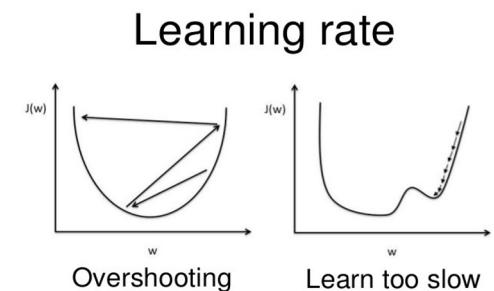
$$\nabla \theta = \frac{\partial}{\partial \theta} \frac{1}{n} \sum_{i=0}^n \|f^\theta(x) - y\|_2^2$$

均方误差  
Mean Square Error

损失函数  
Loss Function

怎么选择 / 设计损失函数 ?

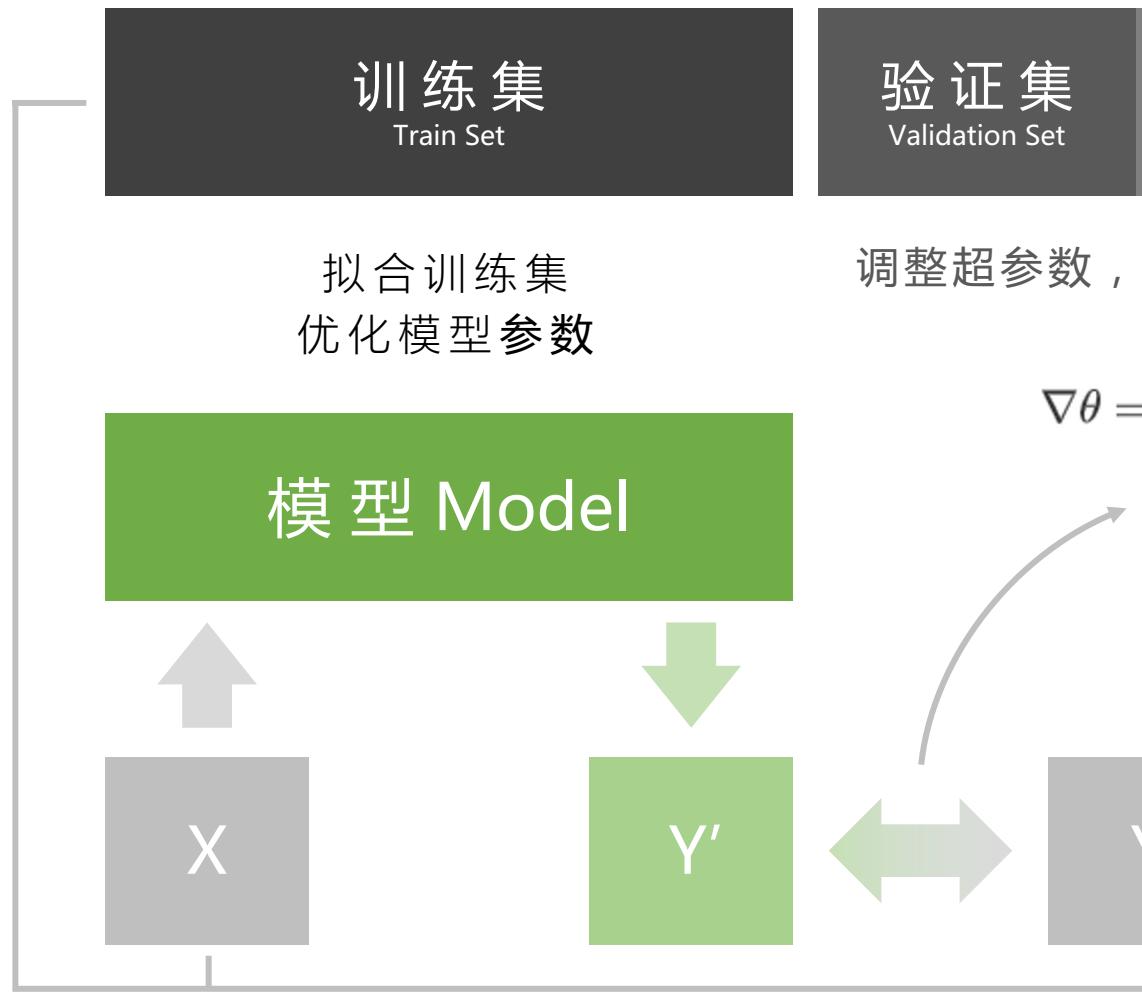
- 小车识别 ?
- 西瓜评估 ?
- 推荐系统 ?



优化  
Optimization

# A General Machine Learning Paradigm : Supervised Learning

## How can a machine learn ?

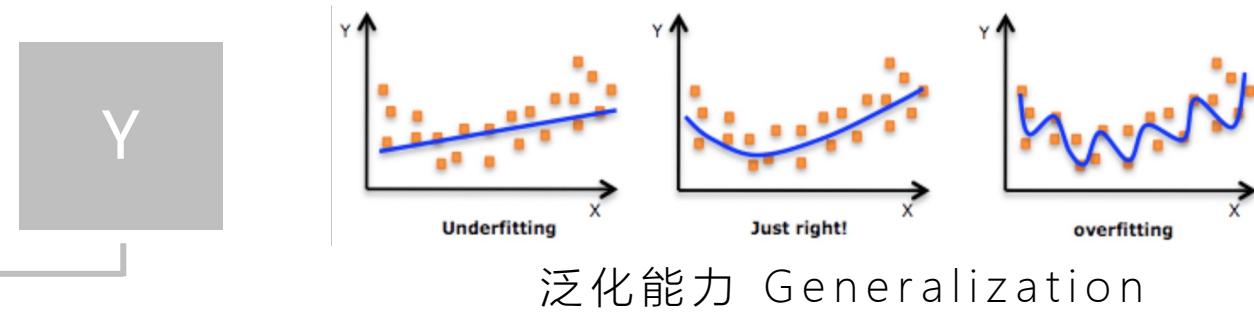


拟合训练集  
优化模型参数

调整超参数，观察泛化性能

$$\nabla \theta = \frac{\partial}{\partial \theta} \frac{1}{n} \sum_{i=0}^n \|f^\theta(x) - y\|_2^2$$
 均方误差  
$$\theta = \theta - \alpha \nabla \theta$$
 Mean Square Error

损失函数  
Loss Function



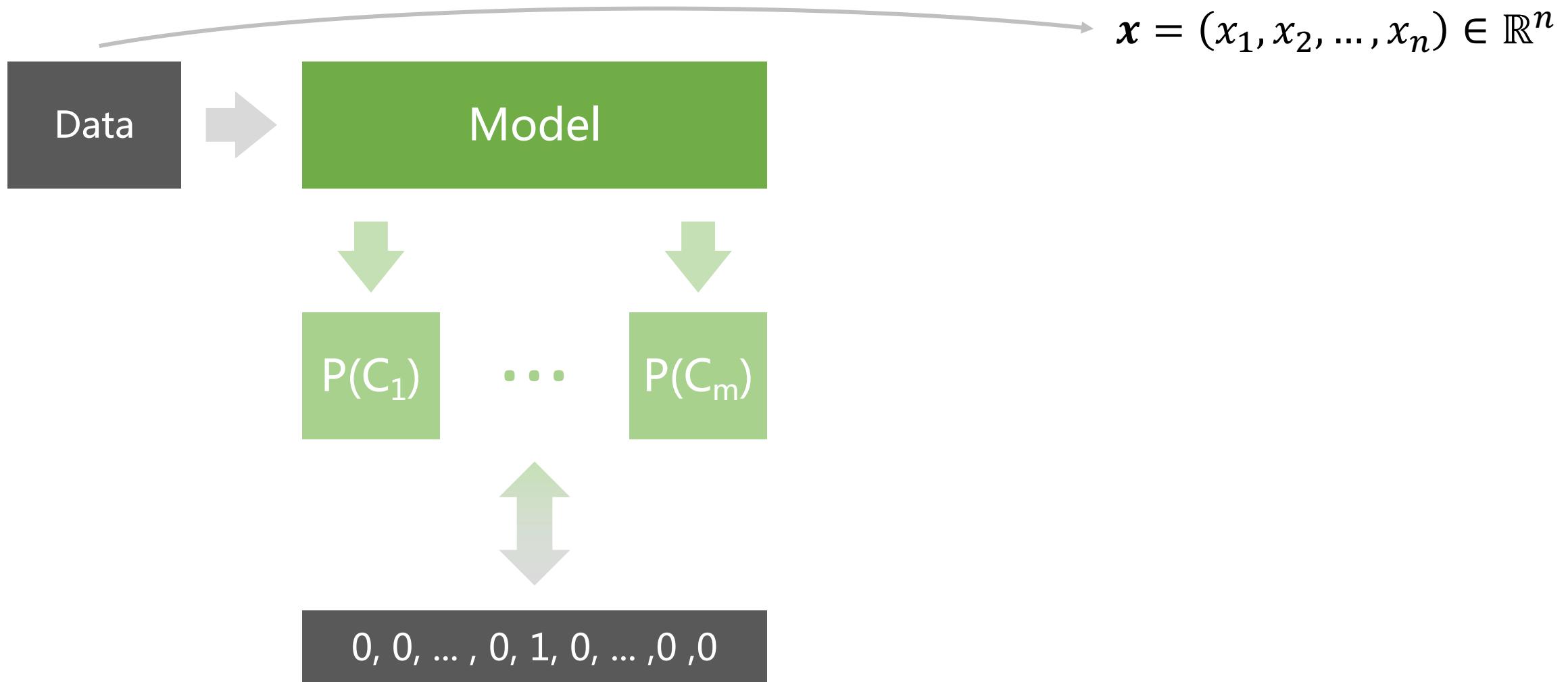
# Machine Learning



1. Artificial Intelligence and Machine Learning
2. One of the General Machine Learning Paradigms: Supervised Learning
3. Solving a Classification Task with the Linear Model
4. Neural Networks
  - a. Multilayer Perceptron (MLP)
  - b. Convolutional Neural Networks (CNN)
  - c. Recurrent Neural Networks (RNN)
5. Some Ideas and Fancy Models
  - a. Representation Learning
  - b. The Encoder-Decoder Architecture and Autoencoder
  - c. Generative Adversarial Network (GAN)
  - d. Reinforcement Learning
6. Research Areas
7. Limitations
8. Resources

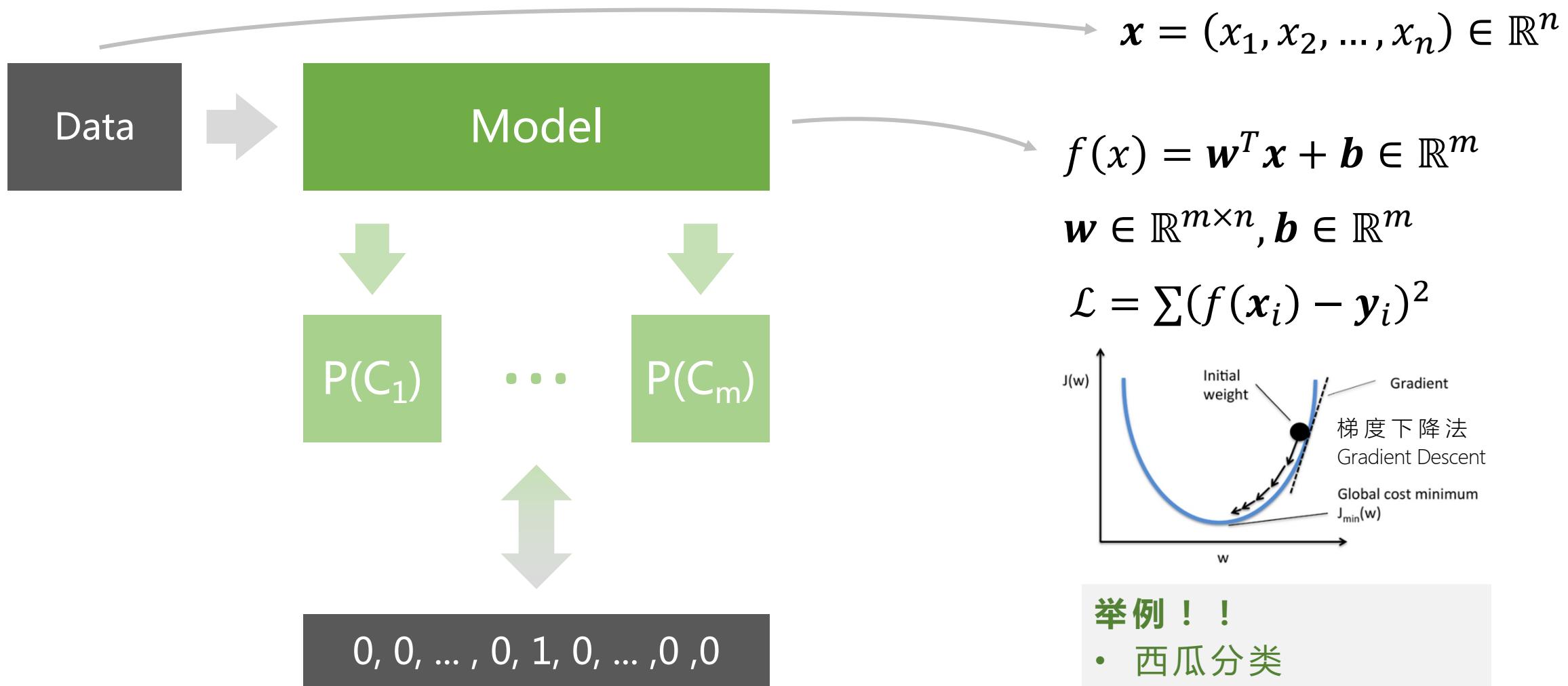
Solving a Classification Task with Linear Model

# What is classification ?



Solving a Classification Task with Linear Model

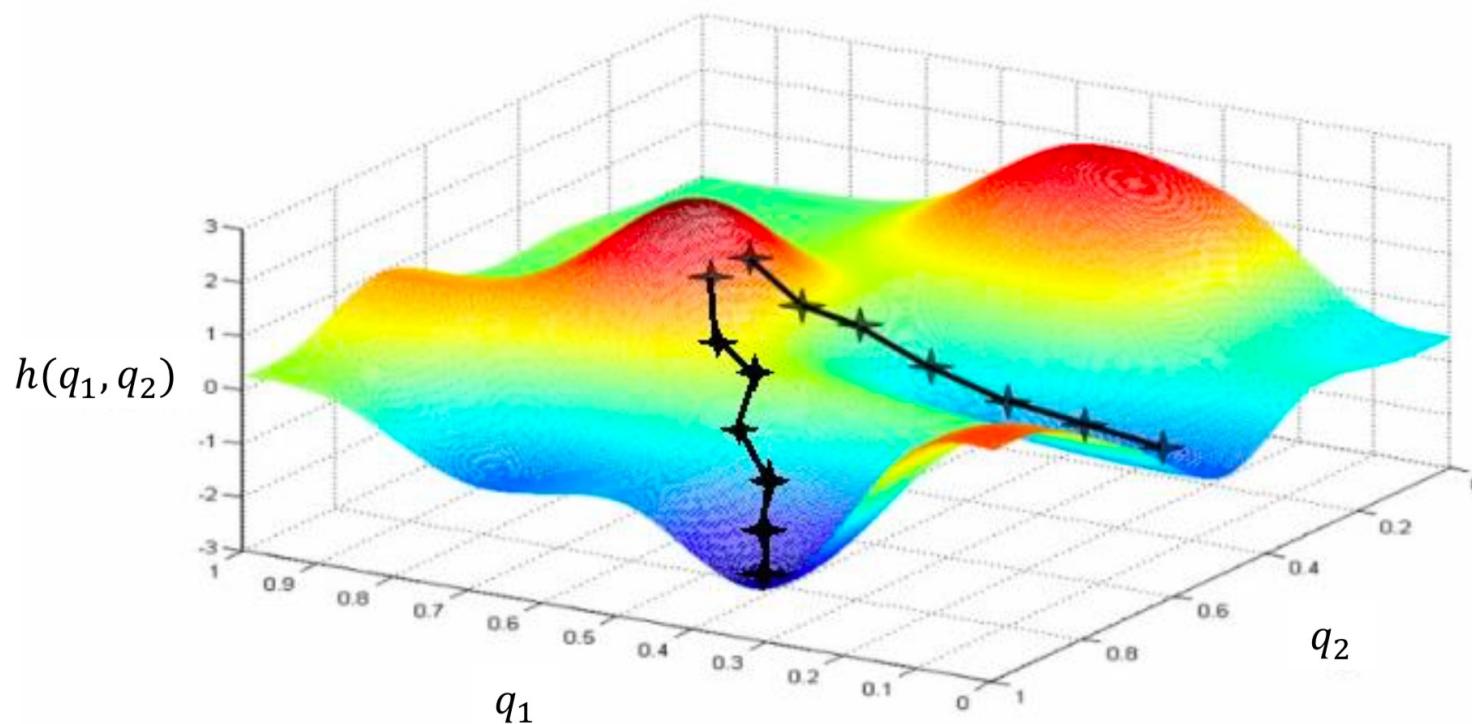
# What is a Linear Model ?



Solving a Classification Task with Linear Model

# What is a Linear Model ?

Visualization of Gradient Descent:



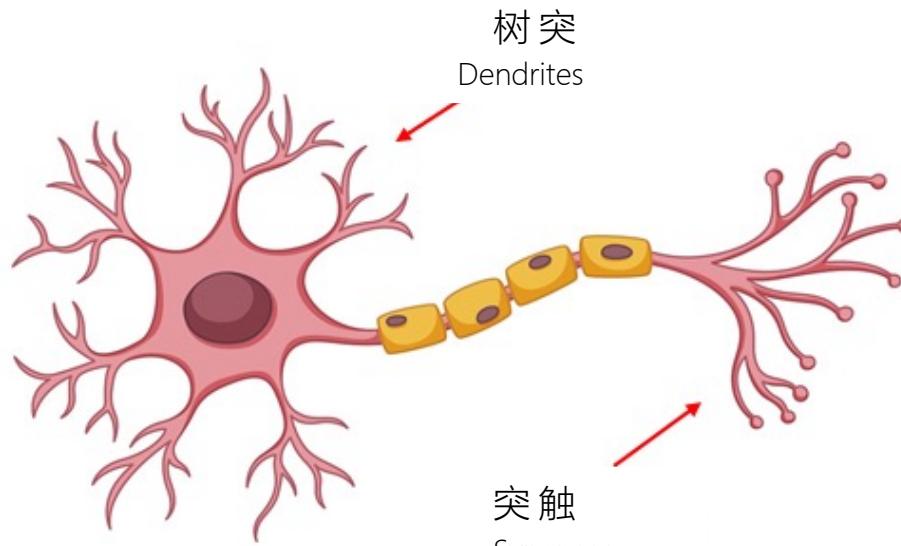
# Machine Learning



1. Artificial Intelligence and Machine Learning
2. One of the General Machine Learning Paradigms: Supervised Learning
3. Solving a Classification Task with the Linear Model
4. Neural Networks
  - a. Multilayer Perceptron (MLP)
  - b. Convolutional Neural Networks (CNN)
  - c. Recurrent Neural Networks (RNN)
5. Some Ideas and Fancy Models
  - a. Representation Learning
  - b. The Encoder-Decoder Architecture and Autoencoder
  - c. Generative Adversarial Network (GAN)
  - d. Reinforcement Learning
6. Research Areas
7. Limitations
8. Resources

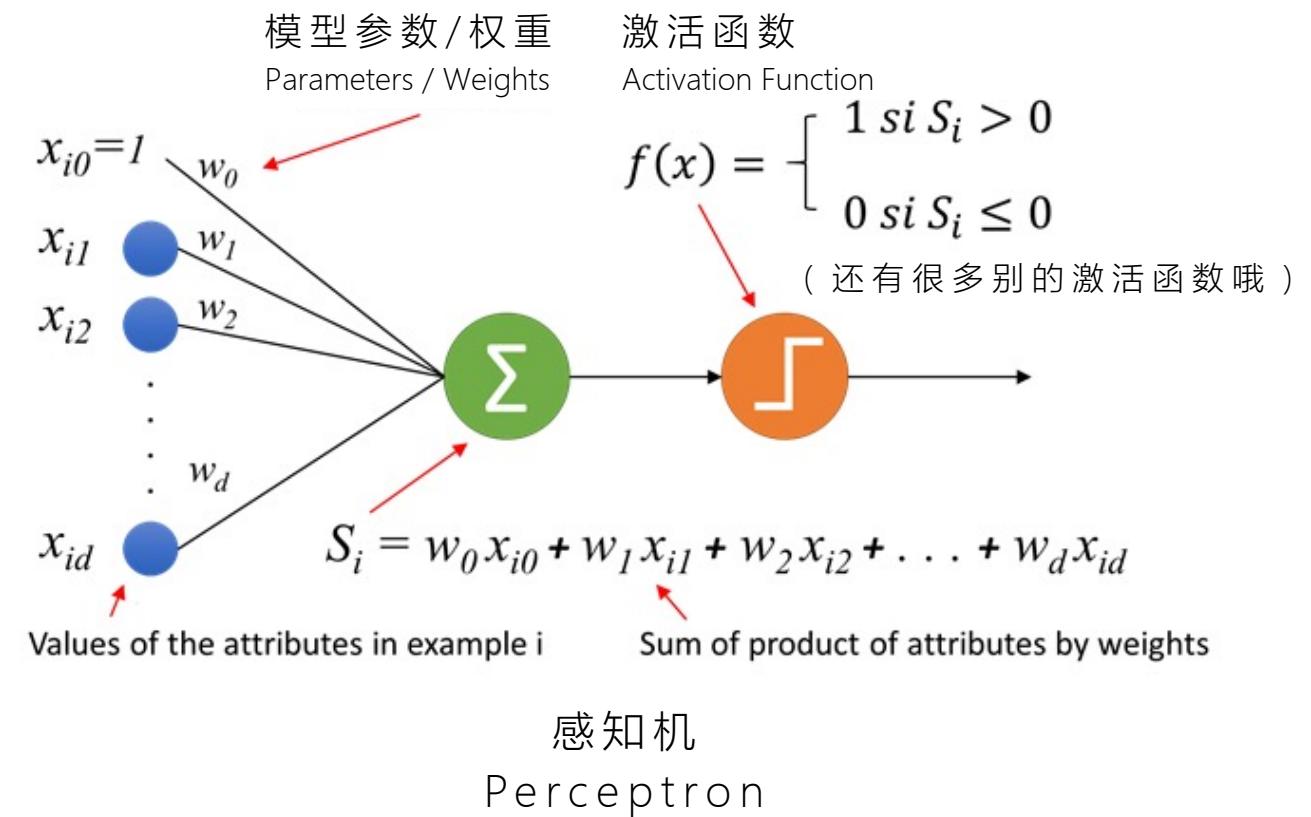
# Neural Networks

## The simplest example: A perceptron



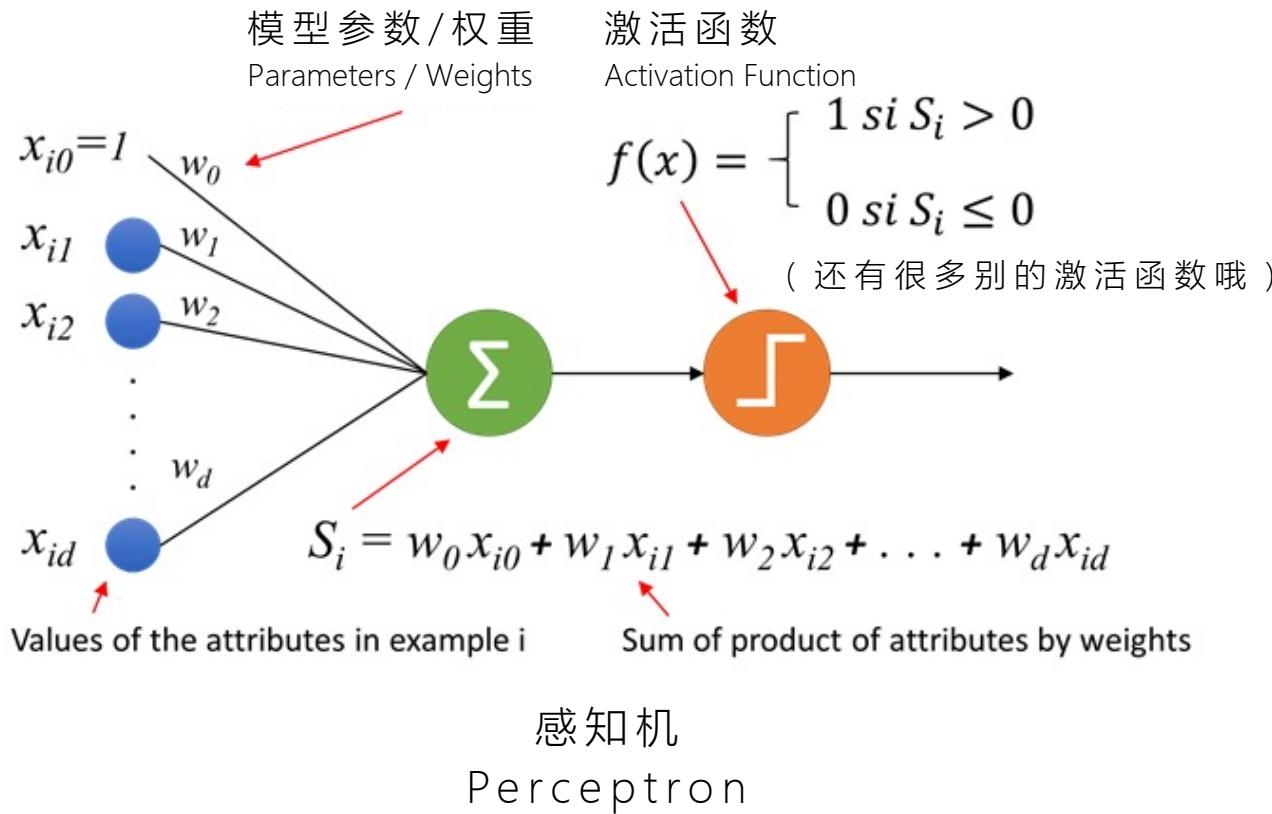
神经元

Neuron



# Neural Networks

## The simplest example: A perceptron

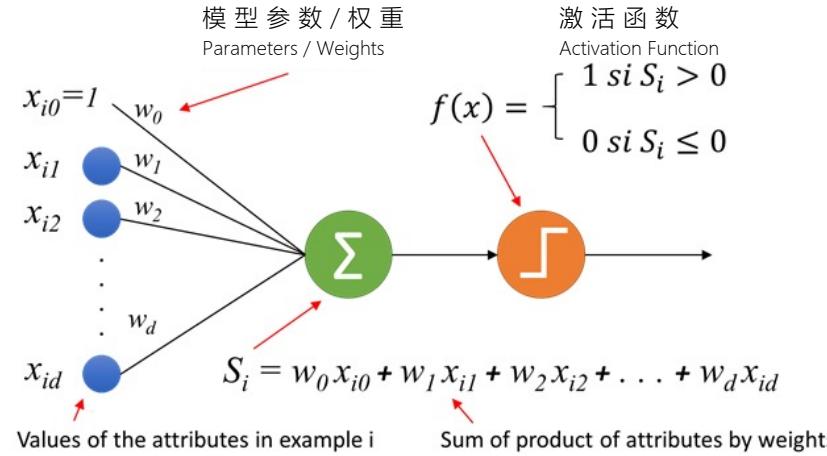


$$y = f_{activation} (\sum \mathbf{w}^T \mathbf{x})$$

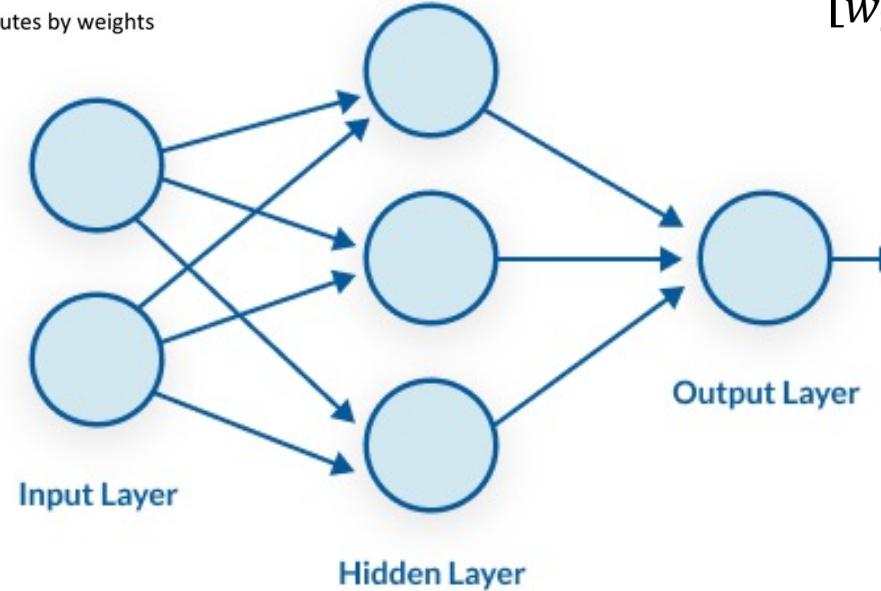
$$f\left( \begin{bmatrix} w_0 & w_1 & \cdots & w_n \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix} \right) = y \in \mathbb{R}$$

# Neural Networks

## More perceptrons !



感知机  
Perceptron

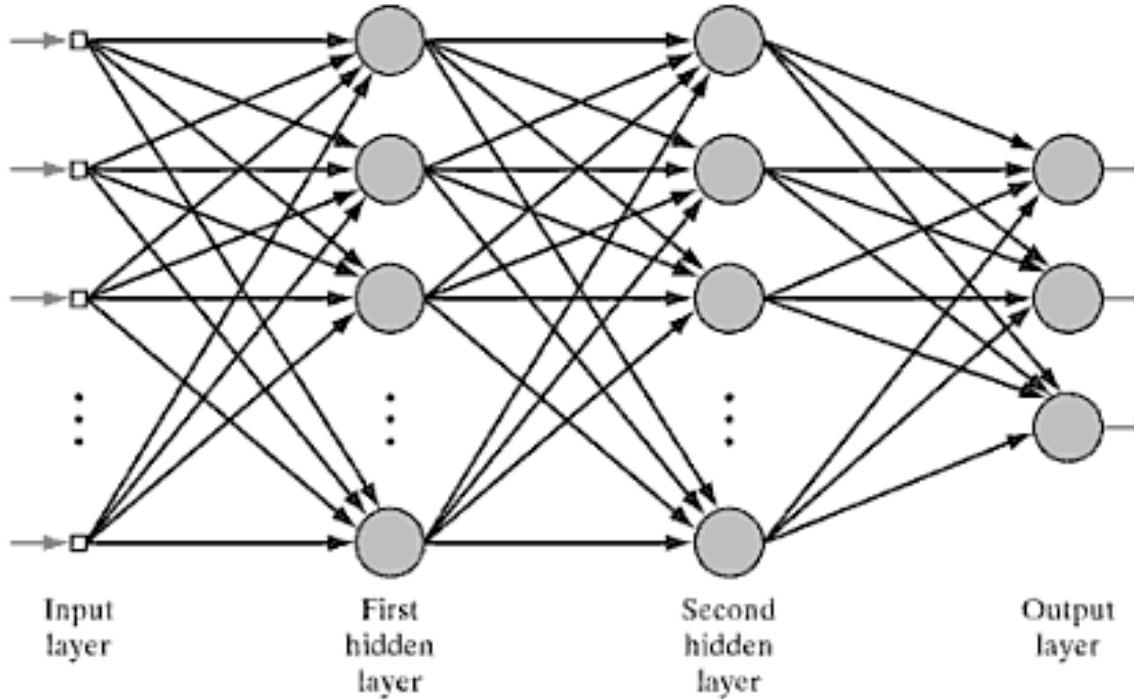


$$\mathbf{y} = f_{\text{activation}} (\sum \mathbf{w}^T \mathbf{x})$$

$$f \left( \begin{bmatrix} w_{10} & w_{11} & \dots & w_{1n} \\ w_{20} & w_{21} & \dots & w_{2n} \\ w_{30} & w_{31} & \dots & w_{3n} \end{bmatrix} \begin{bmatrix} x'_0 \\ x'_1 \\ \vdots \\ x'_n \end{bmatrix} \right) = \begin{bmatrix} x'_0 \\ x'_1 \\ \vdots \\ x'_m \end{bmatrix} \in \mathbb{R}^m$$

# Neural Networks

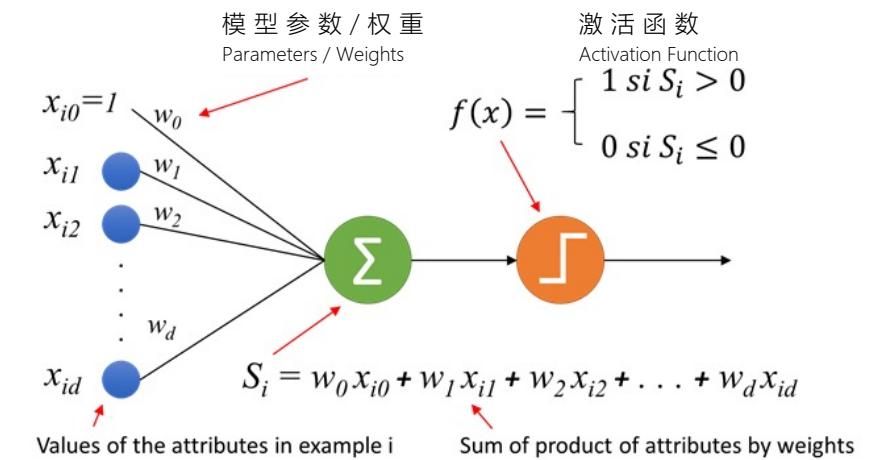
# Multilayer Perceptron \(\geq \nabla \leq\)/



多层感知机 / 前馈神经网络  
Multilayer Perceptron (MLP)  
Feed Forward Network (FFN)

The same for every layer:

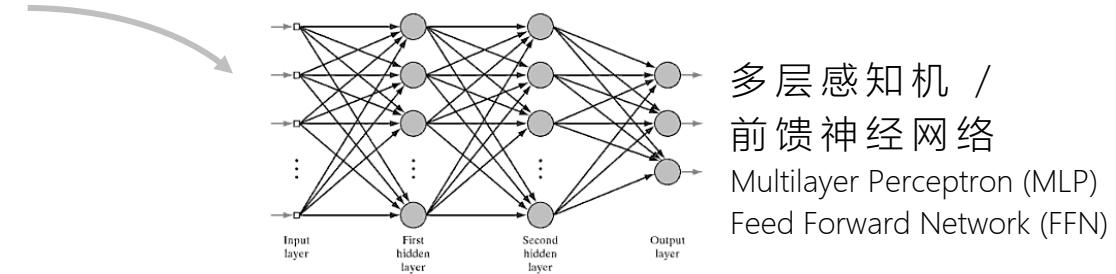
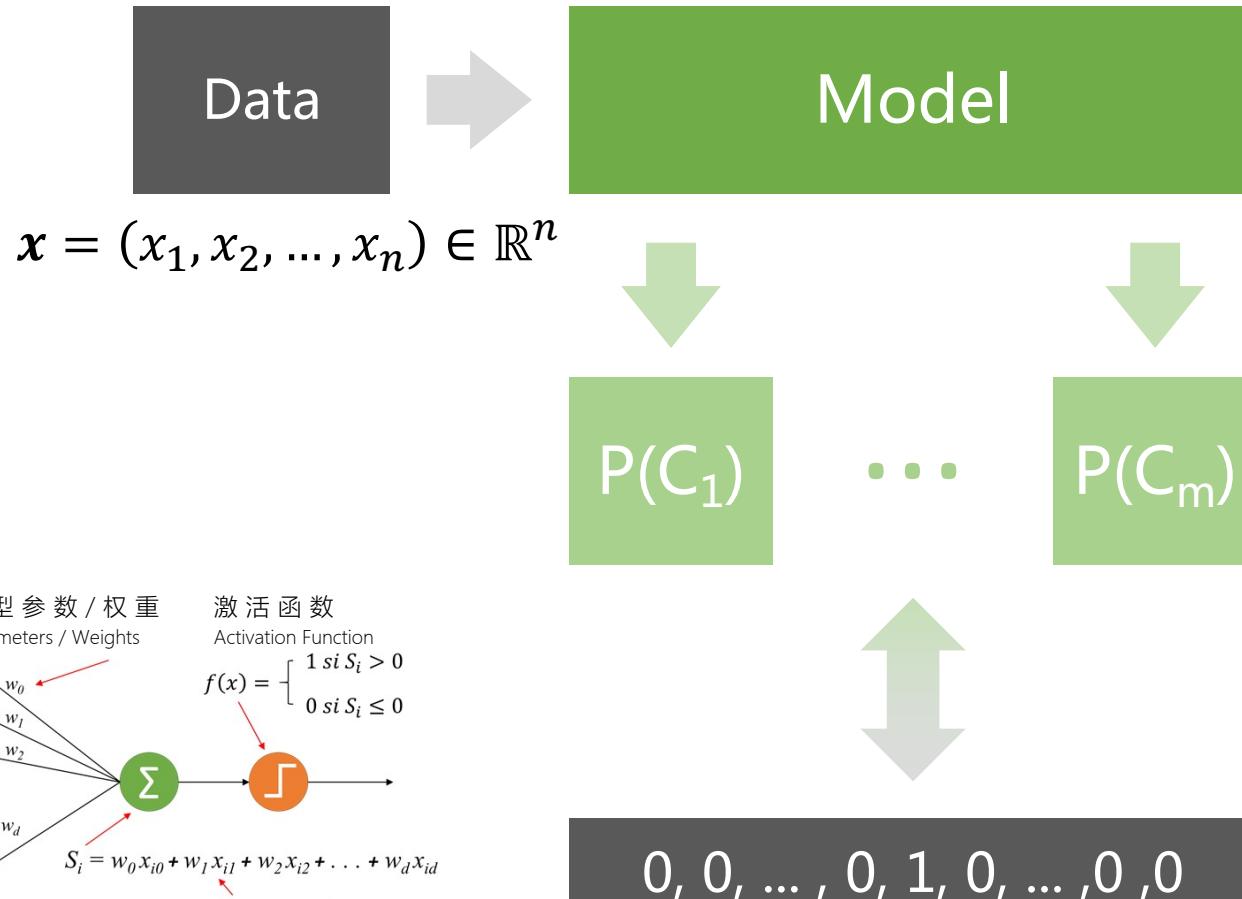
$$f\left(\begin{bmatrix} w_{10} & w_{11} & \dots & w_{1n} \\ \vdots & \vdots & \ddots & w_{in} \\ w_{m0} & w_{m1} & \dots & w_{mn} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix}\right) = \begin{bmatrix} x'_0 \\ x'_1 \\ \vdots \\ x'_m \end{bmatrix} \in \mathbb{R}^m$$



感知机  
Perceptron

# Neural Networks

# Multilayer Perceptron for classification



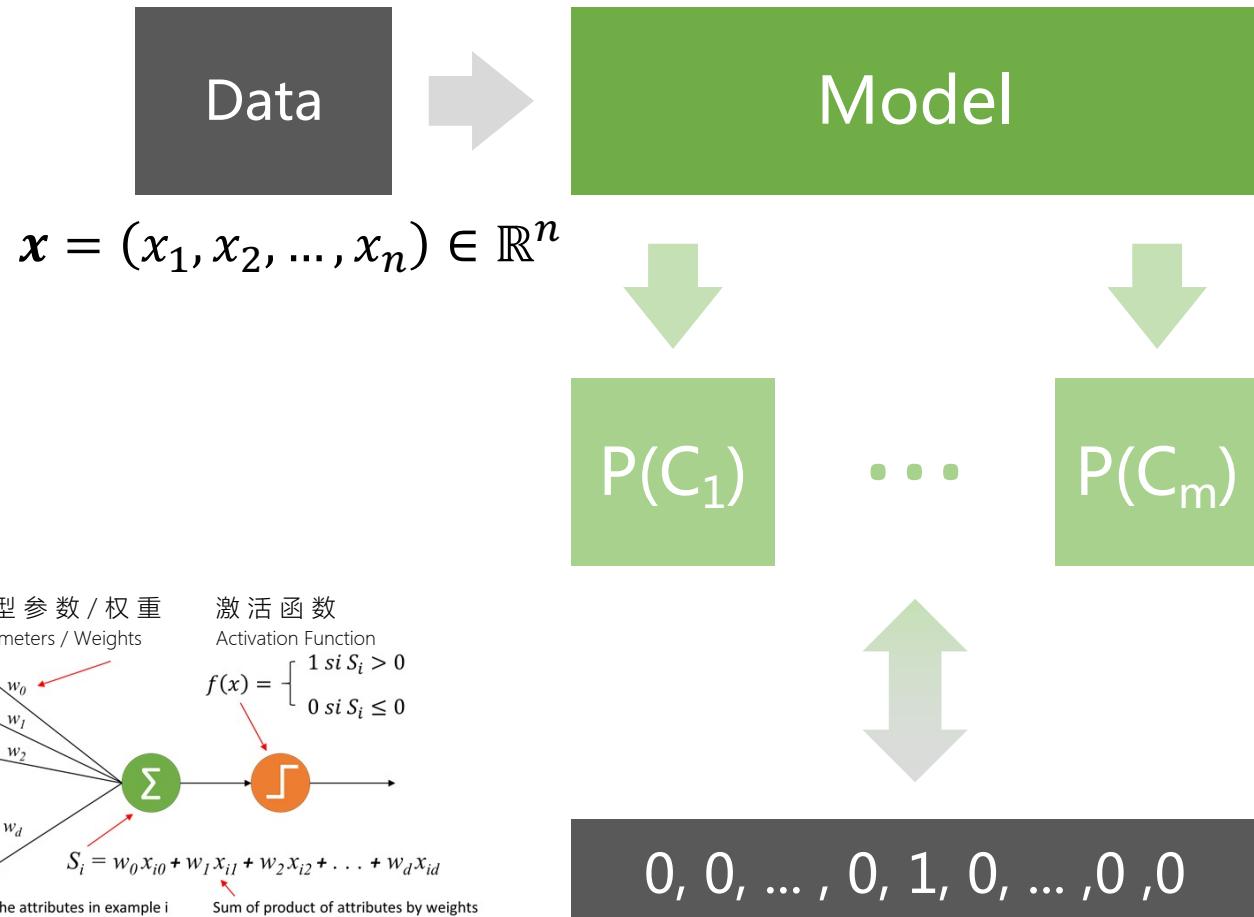
$$\mathcal{L} = \sum (\hat{y}_i - y_i)^2$$

The same for every layer:

$$f\left(\begin{bmatrix} w_{10} & w_{11} & \dots & w_{1n} \\ \vdots & \vdots & \dots & \vdots \\ w_{m0} & w_{m1} & \dots & w_{mn} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix}\right) = \begin{bmatrix} x'_0 \\ x'_1 \\ \vdots \\ x'_m \end{bmatrix} \in \mathbb{R}^m$$

# Neural Networks

# MLP vs. Linear Model



线性回归  
Linear Regression

$$f(x) = \mathbf{w}^T \mathbf{x} + \mathbf{b} \in \mathbb{R}^m$$

$$\mathbf{w} \in \mathbb{R}^{m \times n}, \mathbf{b} \in \mathbb{R}^m$$

多层感知机 /  
前馈神经网络  
Multilayer Perceptron (MLP)  
Feed Forward Network (FFN)

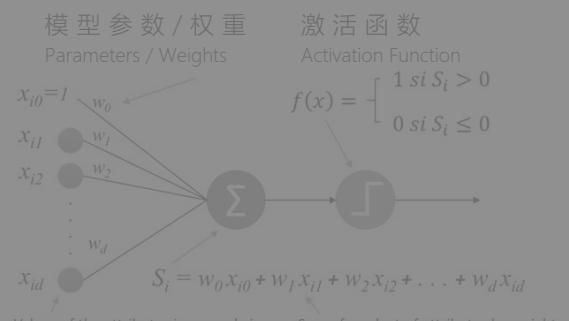
Neural Networks

# MLP vs. Linear Regression

## Perceptron was invented in

$$x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$$

# 1950s

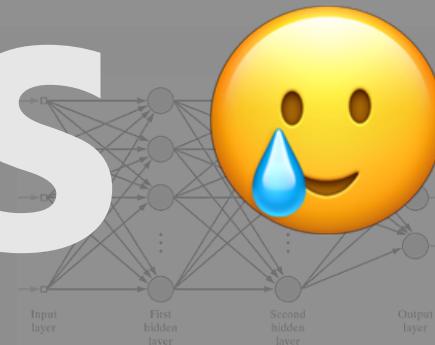


感知机 Perceptron

线性回归

Linear Regression

$$w \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$$



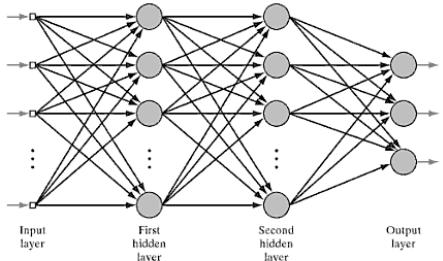
多层感知机 /  
前馈神经网络

Multilayer Perceptron (MLP)  
Feed Forward Network (FFN)

$$f\left(\begin{bmatrix} w_{10} & w_{11} & \dots & w_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m0} & w_{m1} & \dots & w_{mn} \end{bmatrix} \begin{bmatrix} x'_0 \\ x'_1 \\ \vdots \\ x'_n \end{bmatrix}\right) = \begin{bmatrix} x'_0 \\ x'_1 \\ \vdots \\ x'_m \end{bmatrix} \in \mathbb{R}^m$$

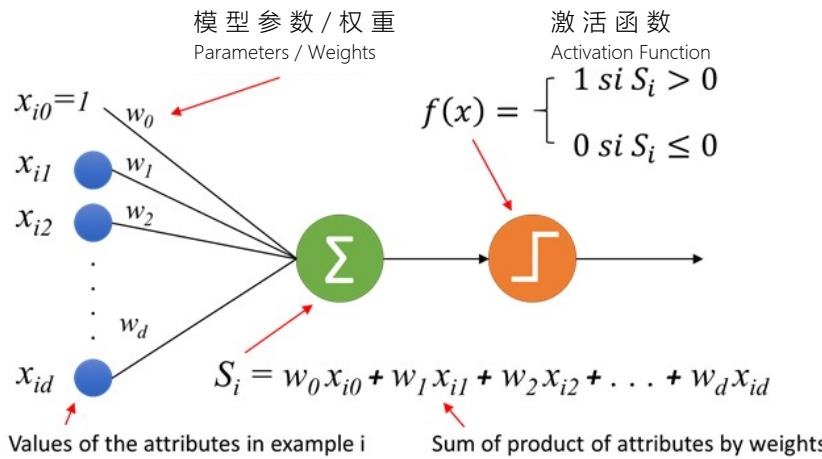
# Neural Networks

# Understanding MLP



多层感知机 /  
前馈神经网络  
Multilayer Perceptron (MLP)  
Feed Forward Network (FFN)

$$f \left( \begin{bmatrix} w_{10} & w_{11} & \dots & w_{1n} \\ \vdots & \vdots & \dots & w_{in} \\ w_{m0} & w_{m1} & \dots & w_{mn} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix} \right) = \begin{bmatrix} x'_0 \\ x'_1 \\ \vdots \\ x'_m \end{bmatrix} \in \mathbb{R}^m$$



感知机 Perceptron

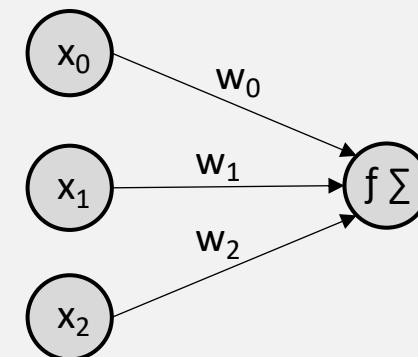
## 如何理解MLP?

- 一个简单的感知机能完成一些逻辑运算

### Task 1

Find  $w_0, w_1, w_2$  for AND, OR, and NOT (for  $x_1$ ).

Note that the bias  $x_0=1$ .

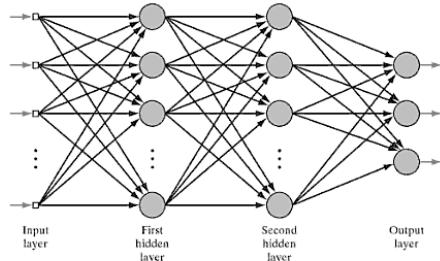


Answer:

AND	[1,1,-2]
OR	[1,1,-0.5]
NOT	[-0.6,0,0.5]

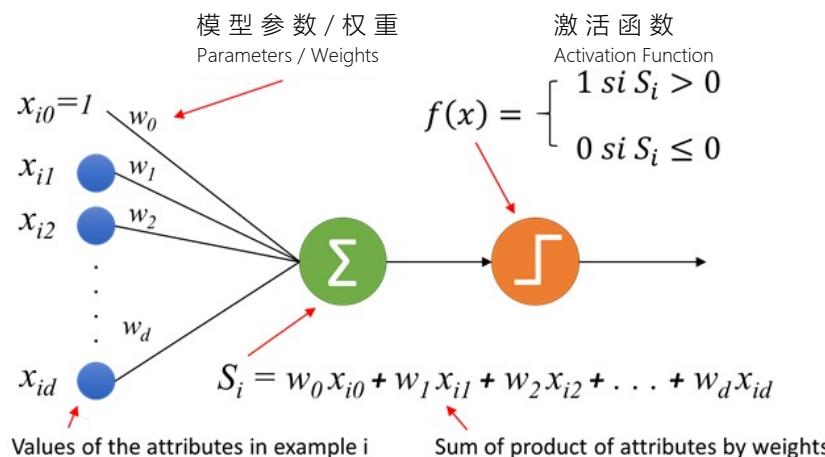
# Neural Networks

# Understanding MLP



多层感知机 /  
前馈神经网络  
Multilayer Perceptron (MLP)  
Feed Forward Network (FFN)

$$f\left(\begin{bmatrix} w_{10} & w_{11} & \dots & w_{1n} \\ \vdots & \vdots & \dots & w_{in} \\ w_{m0} & w_{m1} & \dots & w_{mn} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix}\right) = \begin{bmatrix} x'_0 \\ x'_1 \\ \vdots \\ x'_m \end{bmatrix} \in \mathbb{R}^m$$



感知机 Perceptron

## 如何理解MLP?

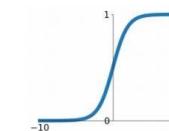
1. 一个简单的感知机能完成一些逻辑运算
2. 线性变换 + 非线性变换去逼近输入到输出

## Task 2

1. Go to <https://playground.tensorflow.org>
2. Set Dataset=2, LR=0.3, Activation=Linear
3. Set Dataset=2, LR=0.4, Activation=Sigmoid
4. Play around and see who can get the highest accuracy on dataset 4!

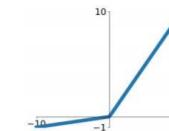
### Sigmoid

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



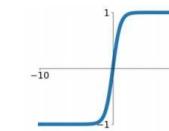
### Leaky ReLU

$$\max(0.1x, x)$$



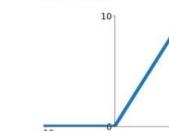
### tanh

$$\tanh(x)$$



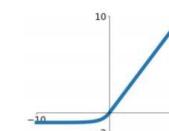
### Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$



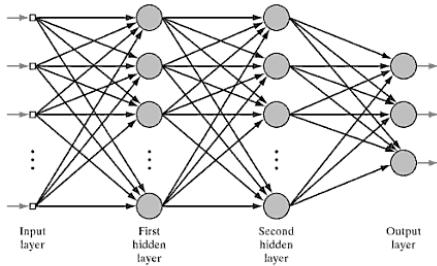
### ReLU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



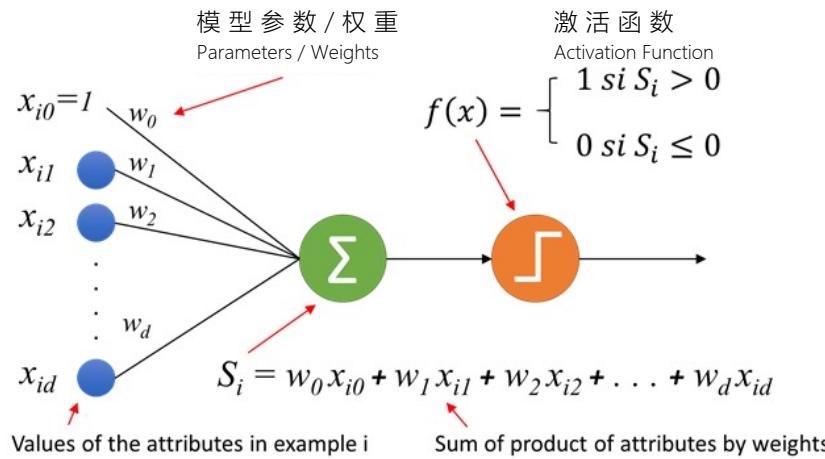
# Neural Networks

# Understanding MLP



多层感知机 /  
前馈神经网络  
Multilayer Perceptron (MLP)  
Feed Forward Network (FFN)

$$f\left(\begin{bmatrix} w_{10} & w_{11} & \dots & w_{1n} \\ \vdots & \vdots & \dots & w_{in} \\ w_{m0} & w_{m1} & \dots & w_{mn} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix}\right) = \begin{bmatrix} x'_0 \\ x'_1 \\ \vdots \\ x'_m \end{bmatrix} \in \mathbb{R}^m$$

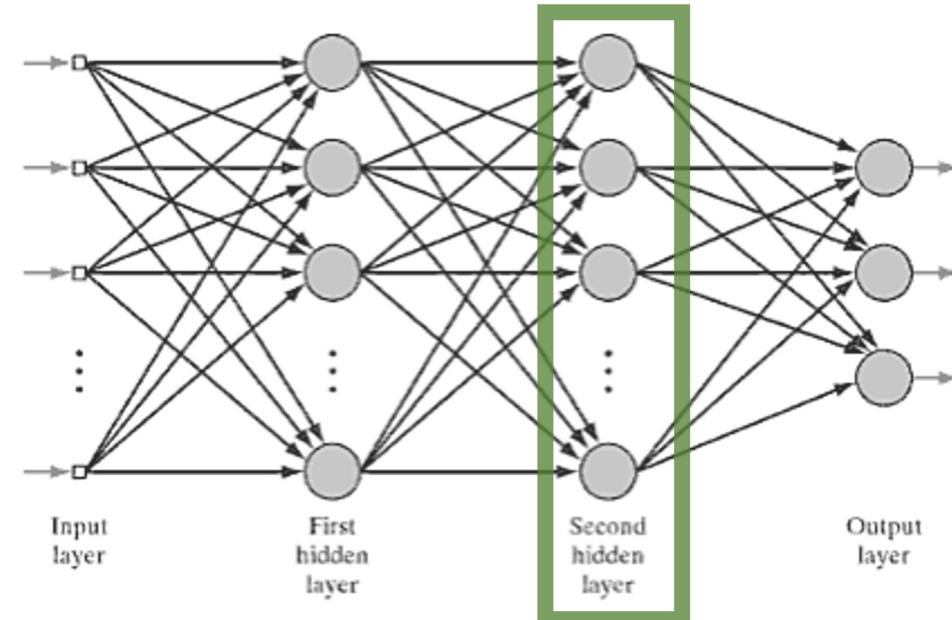


感知机 Perceptron

## 如何理解 MLP ?

1. 一个简单的感知机能完成一些逻辑运算
2. 线性变换 + 非线性变换去逼近输入到输出
3. 隐层形成的表征向量 ( 表示学习 )

人工特征工程 → 表示学习学习特征



# Machine Learning

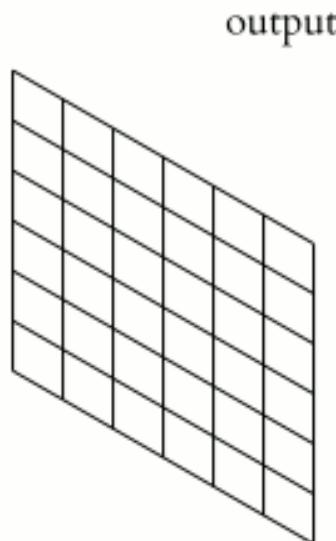
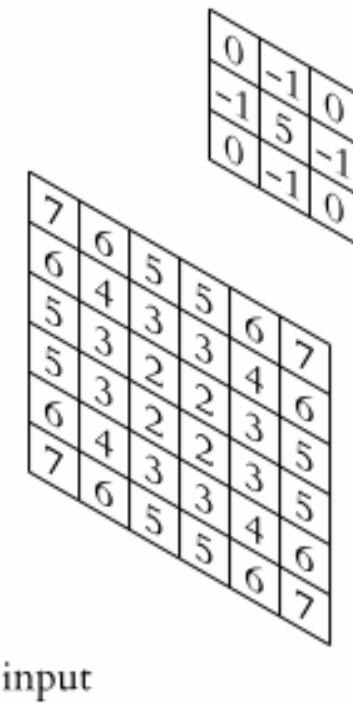


1. Artificial Intelligence and Machine Learning
2. One of the General Machine Learning Paradigms: Supervised Learning
3. Solving a Classification Task with the Linear Model
4. Neural Networks
  - a. Multilayer Perceptron (MLP)
  - b. Convolutional Neural Networks (CNN)
  - c. Recurrent Neural Networks (RNN)
5. Some Ideas and Fancy Models
  - a. Representation Learning
  - b. The Encoder-Decoder Architecture and Autoencoder
  - c. Generative Adversarial Network (GAN)
  - d. Reinforcement Learning
6. Research Areas
7. Limitations
8. Resources

*Neural Networks*

# What is convolution?

a convolution matrix is also called a filter matrix

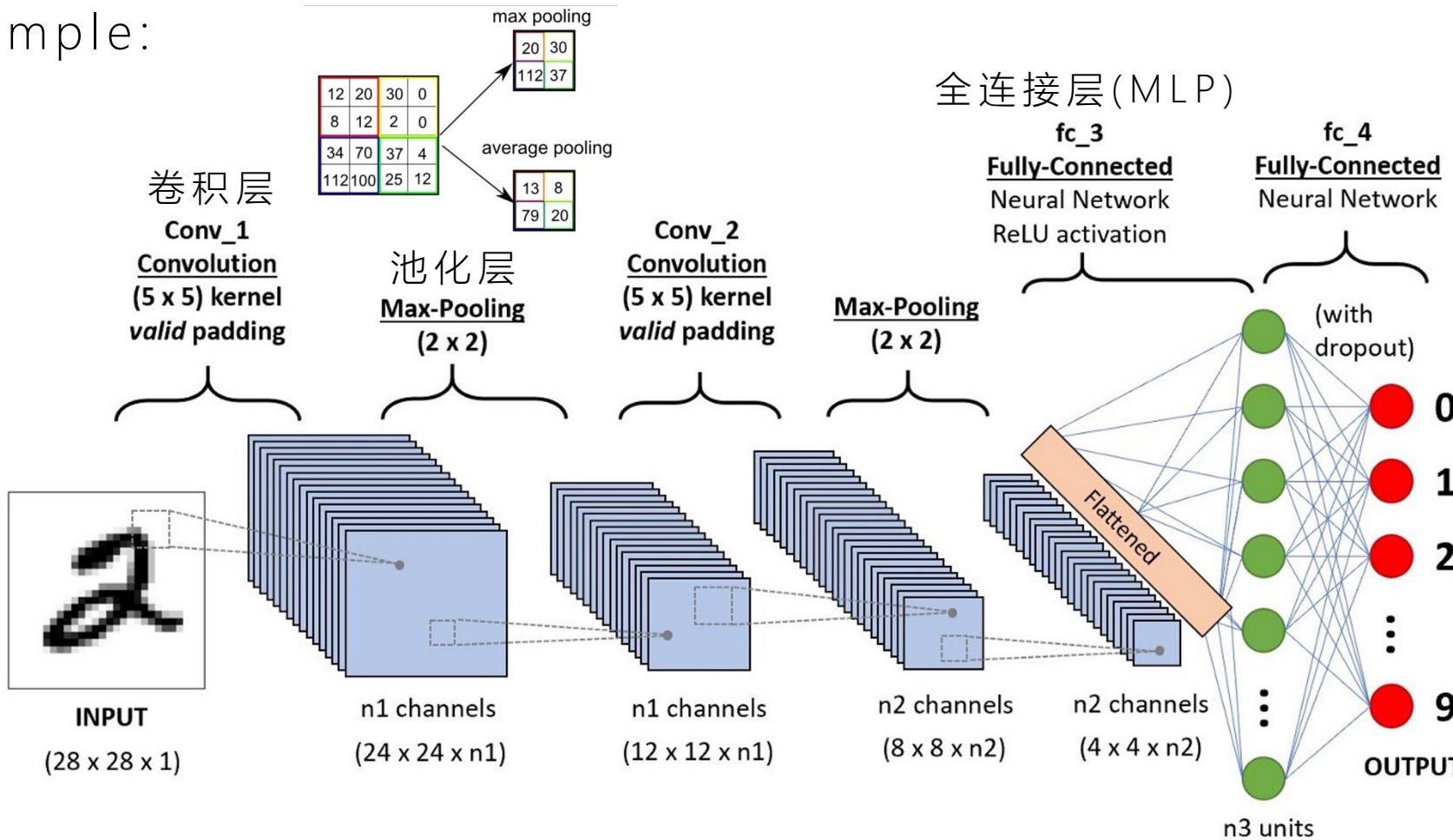


Operation	Kernel $\omega$	Image result $g(x,y)$
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Ridge detection	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 4 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Gaussian blur 3x3 (approximation)	$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	
Gaussian blur 5x5 (approximation)	$\frac{1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$	
Unsharp masking 5x5 Based on Gaussian blur with amount as 1 and threshold as 0 (with no image mask)	$\frac{-1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & -476 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$	

# Neural Networks

# A convolutional neural network

Example:



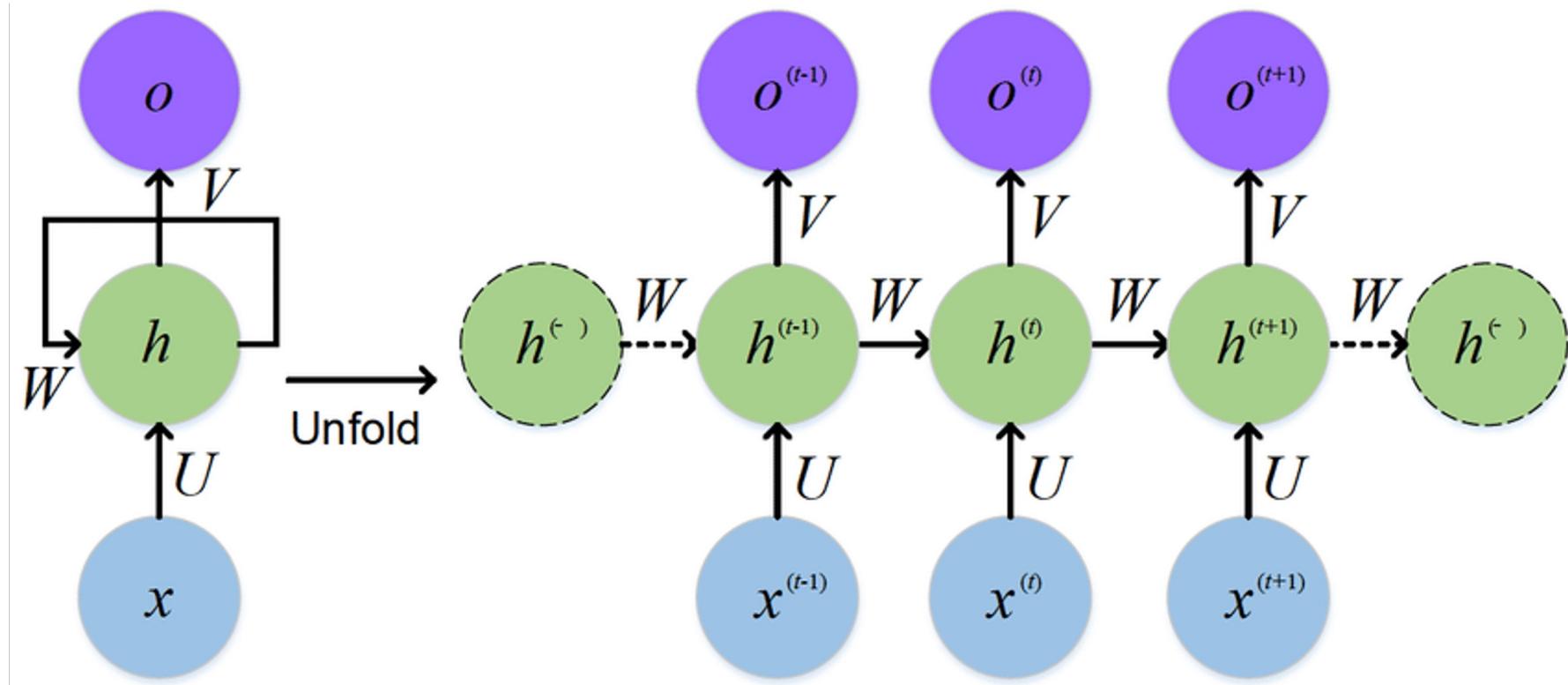
# Machine Learning



1. Artificial Intelligence and Machine Learning
2. One of the General Machine Learning Paradigms: Supervised Learning
3. Solving a Classification Task with the Linear Model
4. Neural Networks
  - a. Multilayer Perceptron (MLP)
  - b. Convolutional Neural Networks (CNN)
  - c. Recurrent Neural Networks (RNN)
5. Some Ideas and Fancy Models
  - a. Representation Learning
  - b. The Encoder-Decoder Architecture and Autoencoder
  - c. Generative Adversarial Network (GAN)
  - d. Reinforcement Learning
6. Research Areas
7. Limitations
8. Resources

*Neural Networks*

# Recurrent Neural Network



# Machine Learning

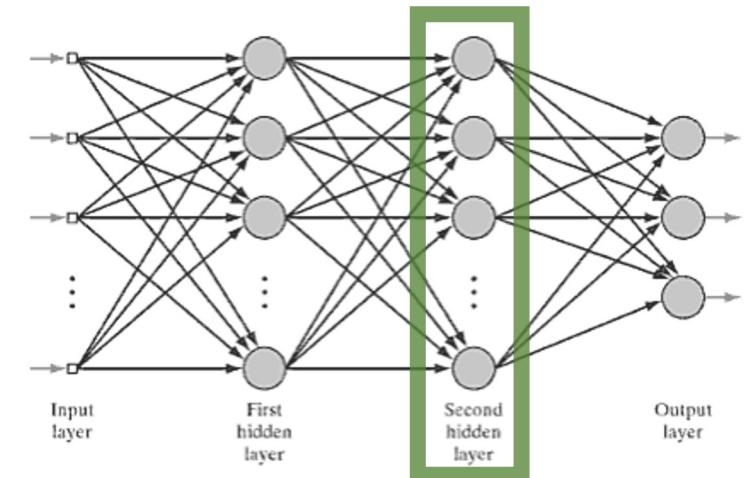
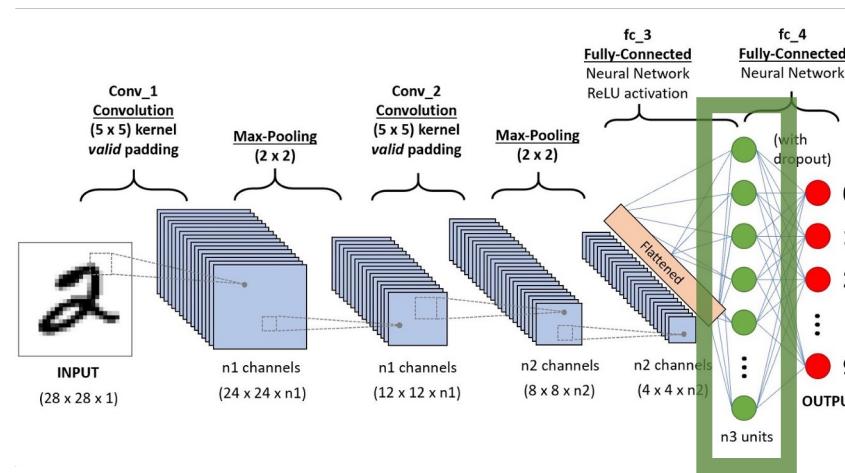
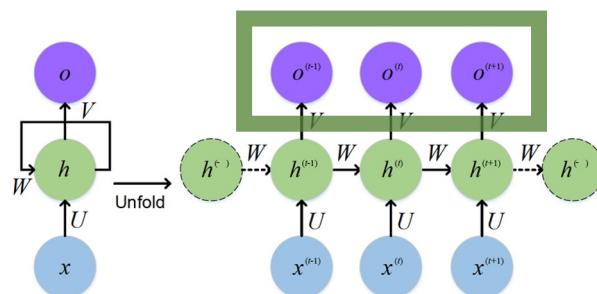


1. Artificial Intelligence and Machine Learning
2. One of the General Machine Learning Paradigms: Supervised Learning
3. Solving a Classification Task with the Linear Model
4. Neural Networks
  - a. Multilayer Perceptron (MLP)
  - b. Convolutional Neural Networks (CNN)
  - c. Recurrent Neural Networks (RNN)
5. Some Ideas and Fancy Models
  - a. Representation Learning
  - b. The Encoder-Decoder Architecture and Autoencoder
  - c. Generative Adversarial Network (GAN)
  - d. Reinforcement Learning
6. Research Areas
7. Limitations
8. Resources

# Neural Networks

# Representation Learning

## 1. What are representations?



*Neural Networks*

# Representation Learning

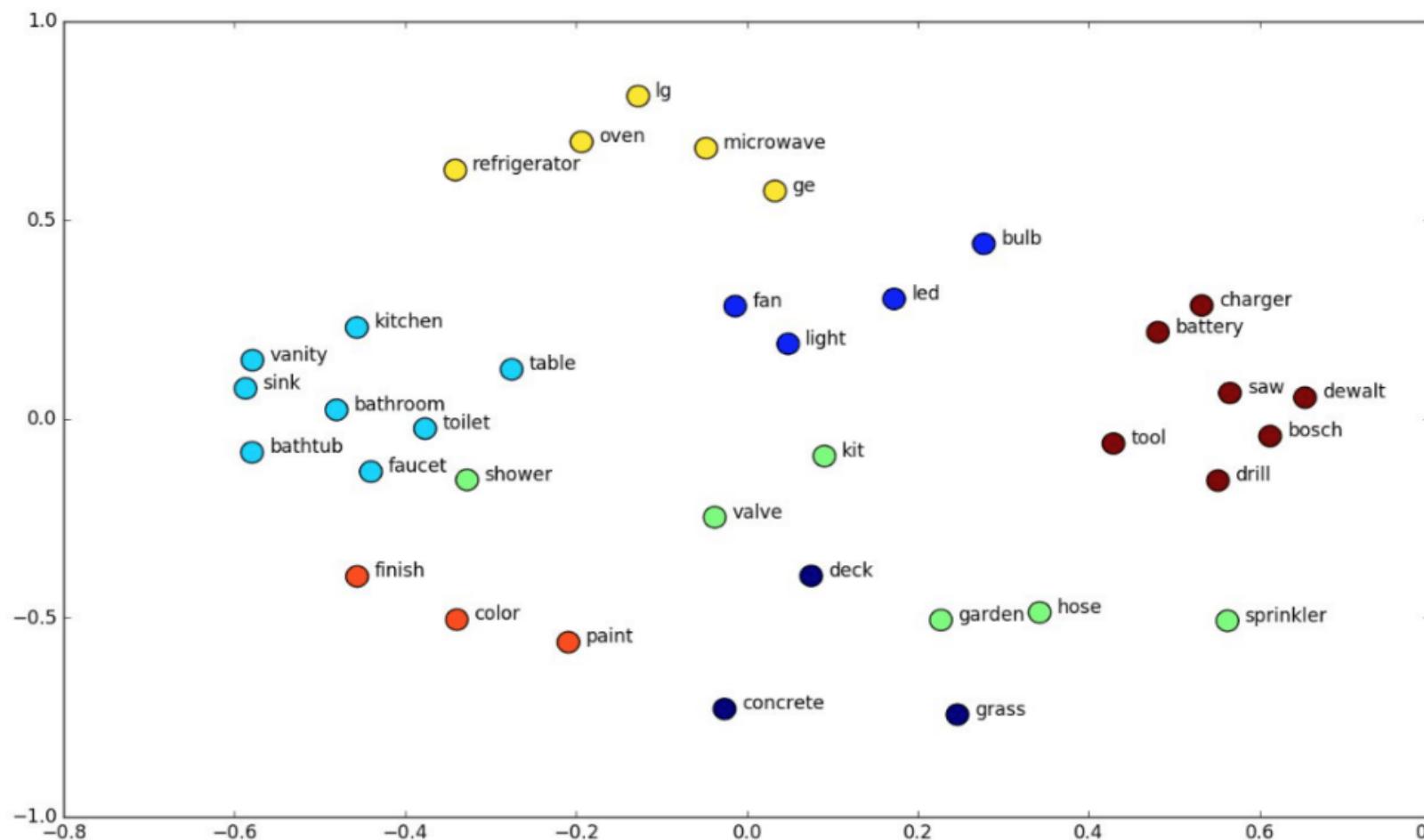
1. What are representations?
2. What is a latent space?



*Neural Networks*

# Representation Learning

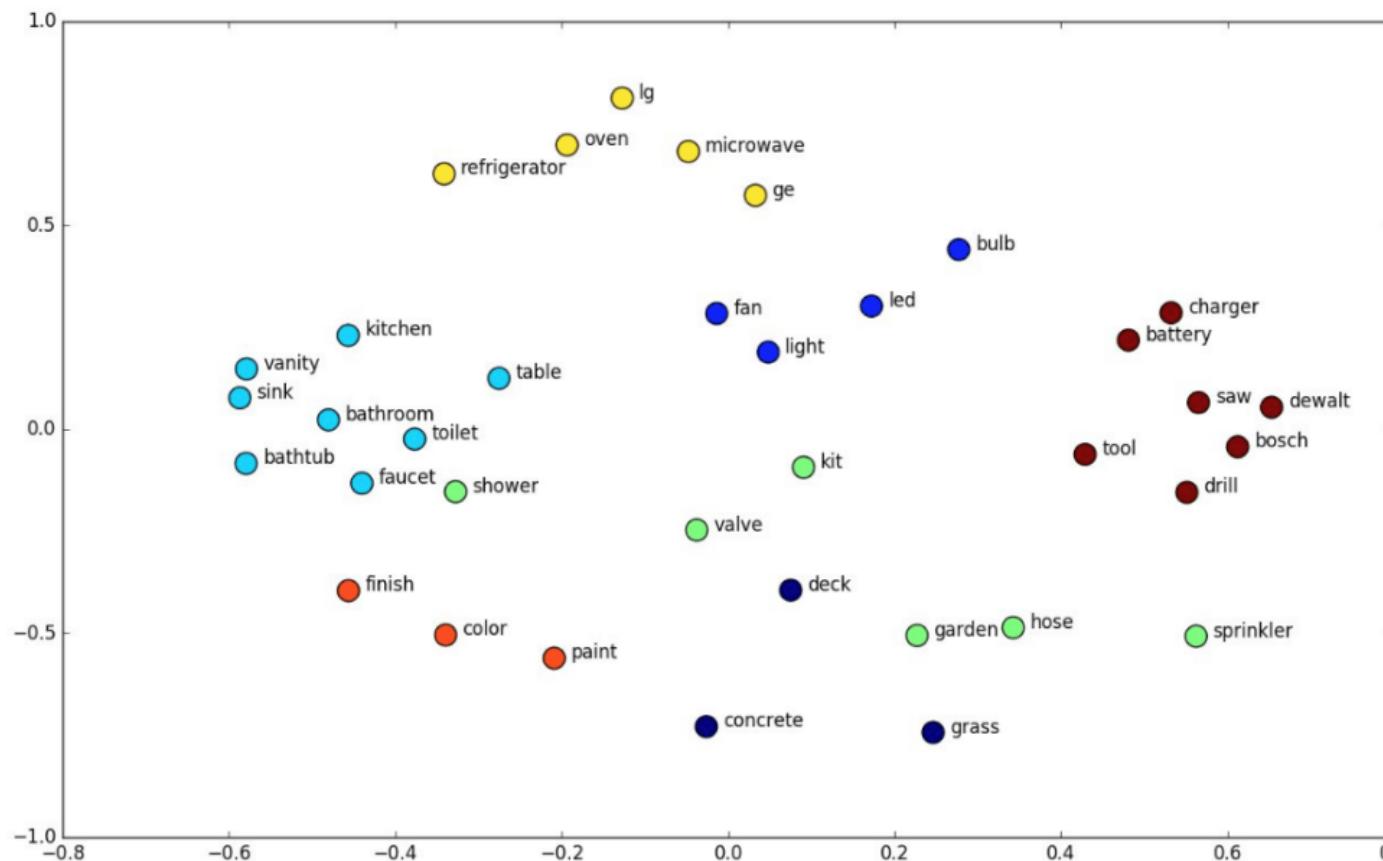
1. What are representations?
2. What is a latent space?



*Neural Networks*

# Representation Learning

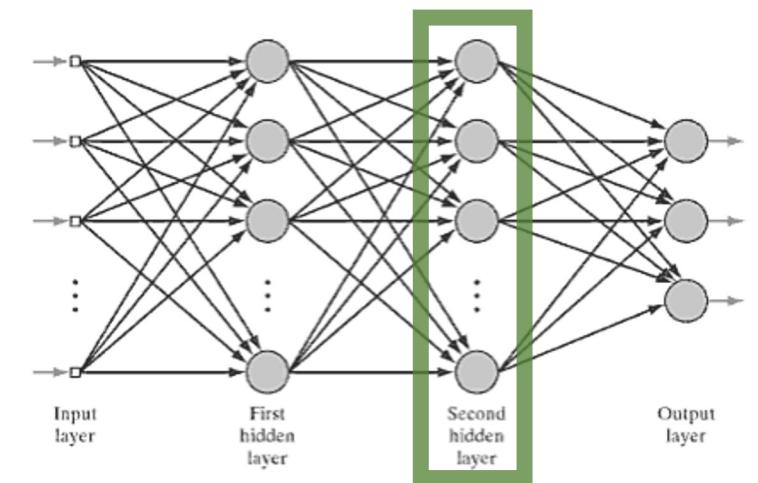
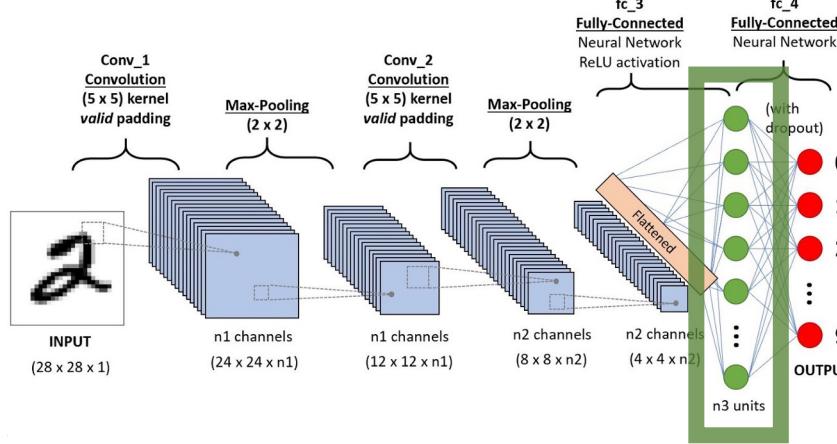
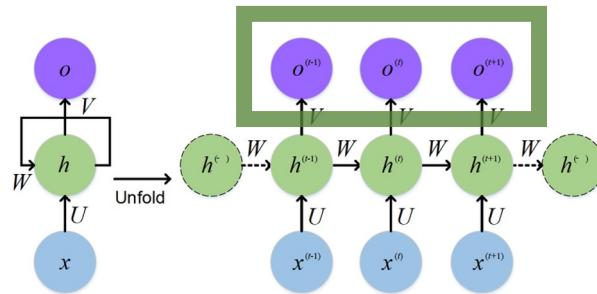
1. What are representations?
2. What is a latent space?
3. Relationship among data samples in the latent space?



# Neural Networks

# Representation Learning

1. What are representations?
2. What is a latent space?
3. Relationship among data samples in the latent space?
4. What are supervision signals for?
5. Using different models on a same dataset?
6. How to get robust and informative representations?



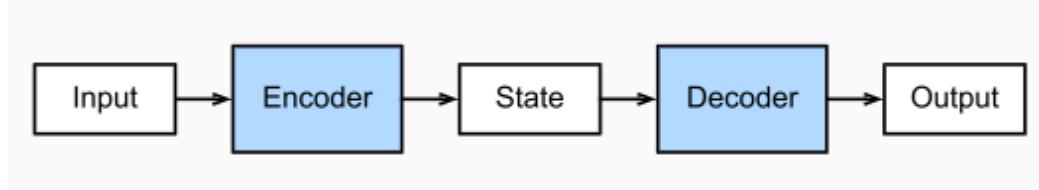
# Machine Learning



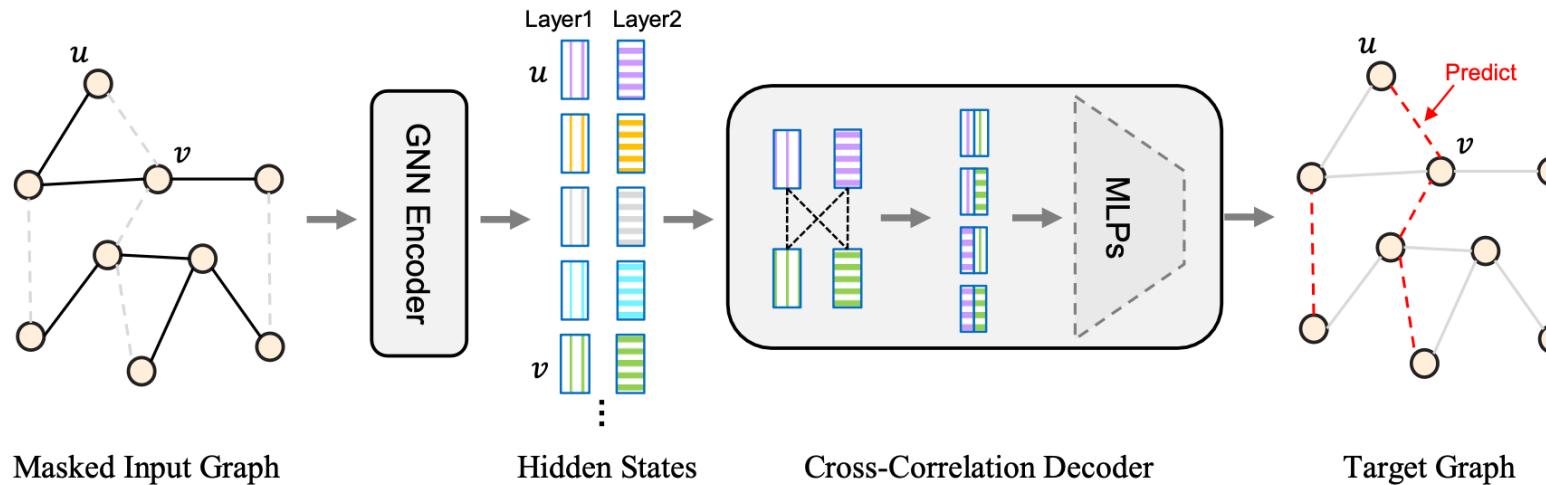
1. Artificial Intelligence and Machine Learning
2. One of the General Machine Learning Paradigms: Supervised Learning
3. Solving a Classification Task with the Linear Model
4. Neural Networks
  - a. Multilayer Perceptron (MLP)
  - b. Convolutional Neural Networks (CNN)
  - c. Recurrent Neural Networks (RNN)
5. Some Ideas and Fancy Models
  - a. Representation Learning
  - b. The Encoder-Decoder Architecture and Autoencoder
  - c. Generative Adversarial Network (GAN)
  - d. Reinforcement Learning
6. Research Areas
7. Limitations
8. Resources

*Neural Networks*

# The Encoder-Decoder Architecture



Encoder-Decoder Architecture



Masked Autoencoder (an example on Graph Neural Networks)

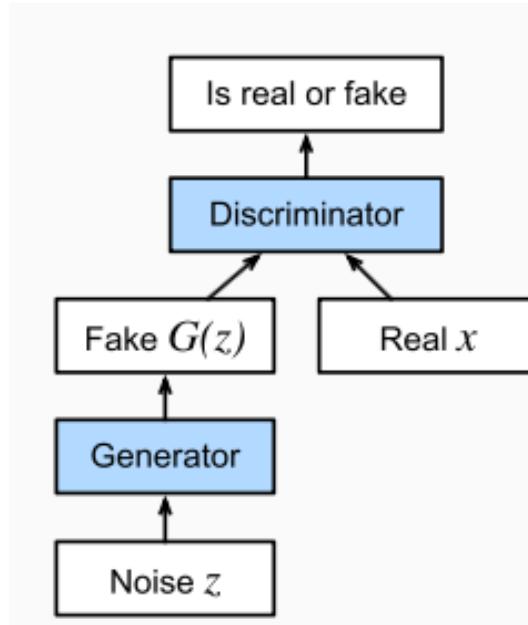
# Machine Learning



1. Artificial Intelligence and Machine Learning
2. One of the General Machine Learning Paradigms: Supervised Learning
3. Solving a Classification Task with the Linear Model
4. Neural Networks
  - a. Multilayer Perceptron (MLP)
  - b. Convolutional Neural Networks (CNN)
  - c. Recurrent Neural Networks (RNN)
5. Some Ideas and Fancy Models
  - a. Representation Learning
  - b. The Encoder-Decoder Architecture and Autoencoder
  - c. Generative Adversarial Network (GAN)
  - d. Reinforcement Learning
6. Research Areas
7. Limitations
8. Resources

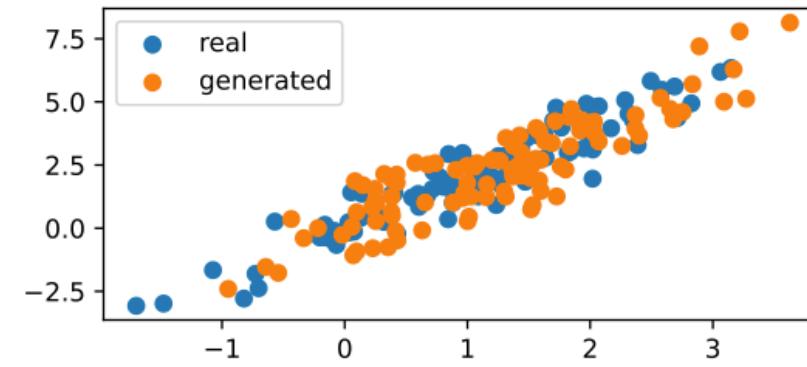
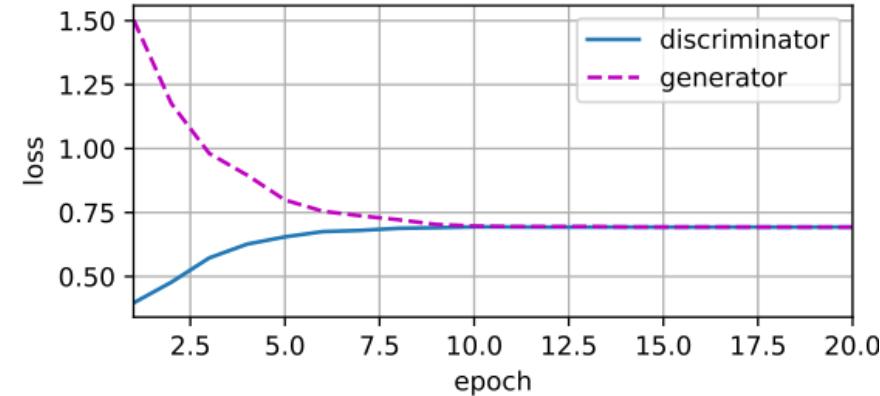
*Neural Networks*

# Generative Adversarial Network



生成式对抗网络

Generative Adversarial Network (GAN)



训练过程例

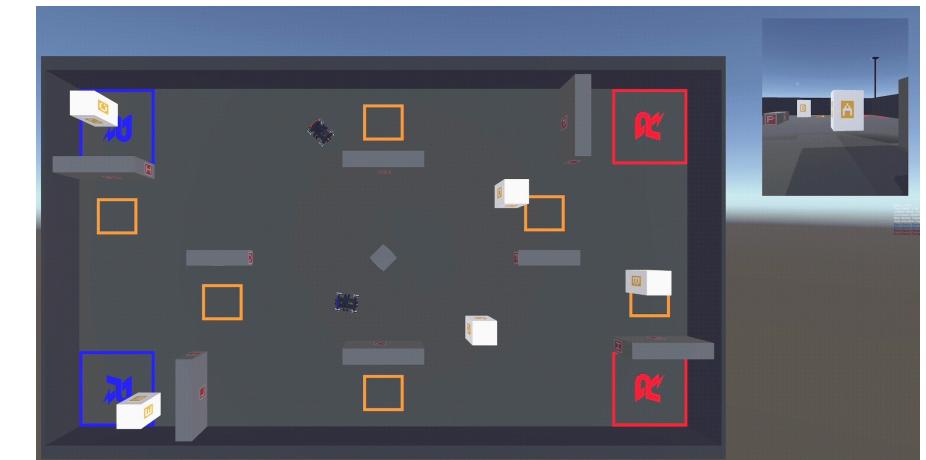
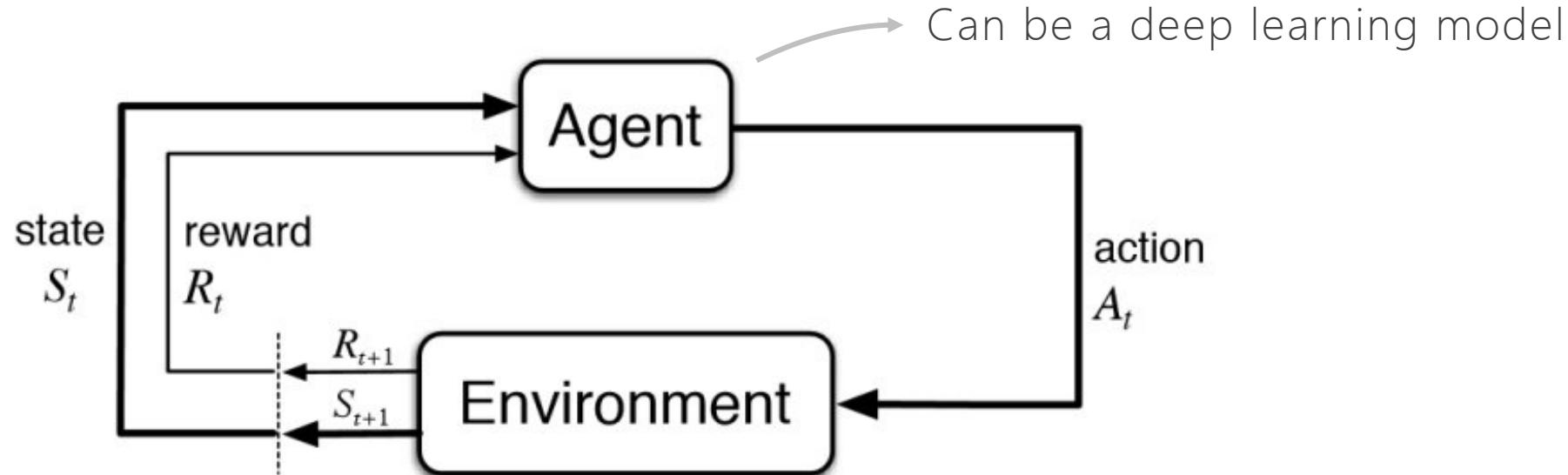
# Machine Learning

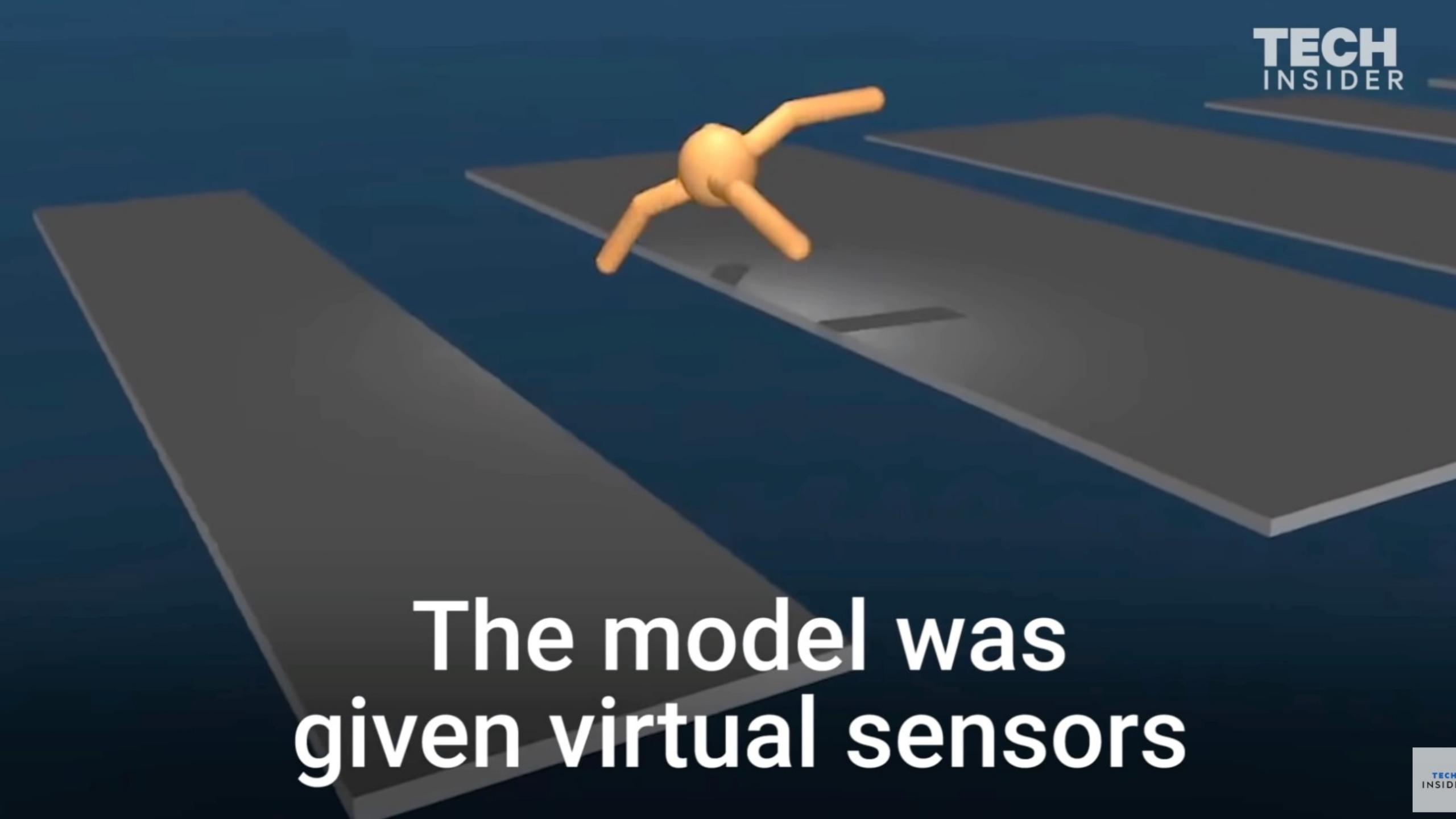


1. Artificial Intelligence and Machine Learning
2. One of the General Machine Learning Paradigms: Supervised Learning
3. Solving a Classification Task with the Linear Model
4. Neural Networks
  - a. Multilayer Perceptron (MLP)
  - b. Convolutional Neural Networks (CNN)
  - c. Recurrent Neural Networks (RNN)
5. Some Ideas and Fancy Models
  - a. Representation Learning
  - b. The Encoder-Decoder Architecture and Autoencoder
  - c. Generative Adversarial Network (GAN)
  - d. Reinforcement Learning
6. Research Areas
7. Limitations
8. Resources

*Neural Networks*

# Reinforcement Learning





The model was  
given virtual sensors

# Machine Learning



1. Artificial Intelligence and Machine Learning
2. One of the General Machine Learning Paradigms: Supervised Learning
3. Solving a Classification Task with the Linear Model
4. Neural Networks
  - a. Multilayer Perceptron (MLP)
  - b. Convolutional Neural Networks (CNN)
  - c. Recurrent Neural Networks (RNN)
5. Some Ideas and Fancy Models
  - a. Representation Learning
  - b. The Encoder-Decoder Architecture and Autoencoder
  - c. Generative Adversarial Network (GAN)
  - d. Reinforcement Learning
6. Research Areas
7. Limitations
8. Resources

# *Astar's Introduction to* **Some Research Areas \***

Recommender Systems

Computer Vision

Natural Language Processing

Medicine

Robotics

Internet of Things

Data Mining

...

Theoretical Analysis

Self-Supervised Learning

Reinforcement Learning

Attention Mechanism

Contrastive Learning

Diffusion Models

Transformers

...

\* Due to limited knowledge, the lists might be incomplete and important areas of research might be missing

*Astar's Introduction to*

# *Limitations of Deep Learning*

- Explainability: When reliability is vital, DL methods might not be safe!
- Data Demanding: Training a DL model might require tremendous amount of data, which is usually hard to obtain.
- Computational Resources Demanding: Very often, larger model make things easier, but them might take months to train.



*Astar's Introduction to*

# *Resources for you*

- 机器学习书籍 : 《机器学习》周志华 (人称西瓜书)
- 机器学习课程 : 吴恩达机器学习 bilibili
- 深度学习课程 : 跟李沐学AI bilibili
- 常用代码框架 : PyTorch (推荐) , TensorFlow 等
- 计算资源获取 : 可以使用社团的 GPU (仅有12G) 或者 CS GPU Farm (CS专业可以用到两张 3090 卡) 或者 Google Colab

*End of  
Astar's  
Rough Introduction to  
Machine Learning*