

第20讲 深度学习怎样提高智能？ ——神经网络与深度学习

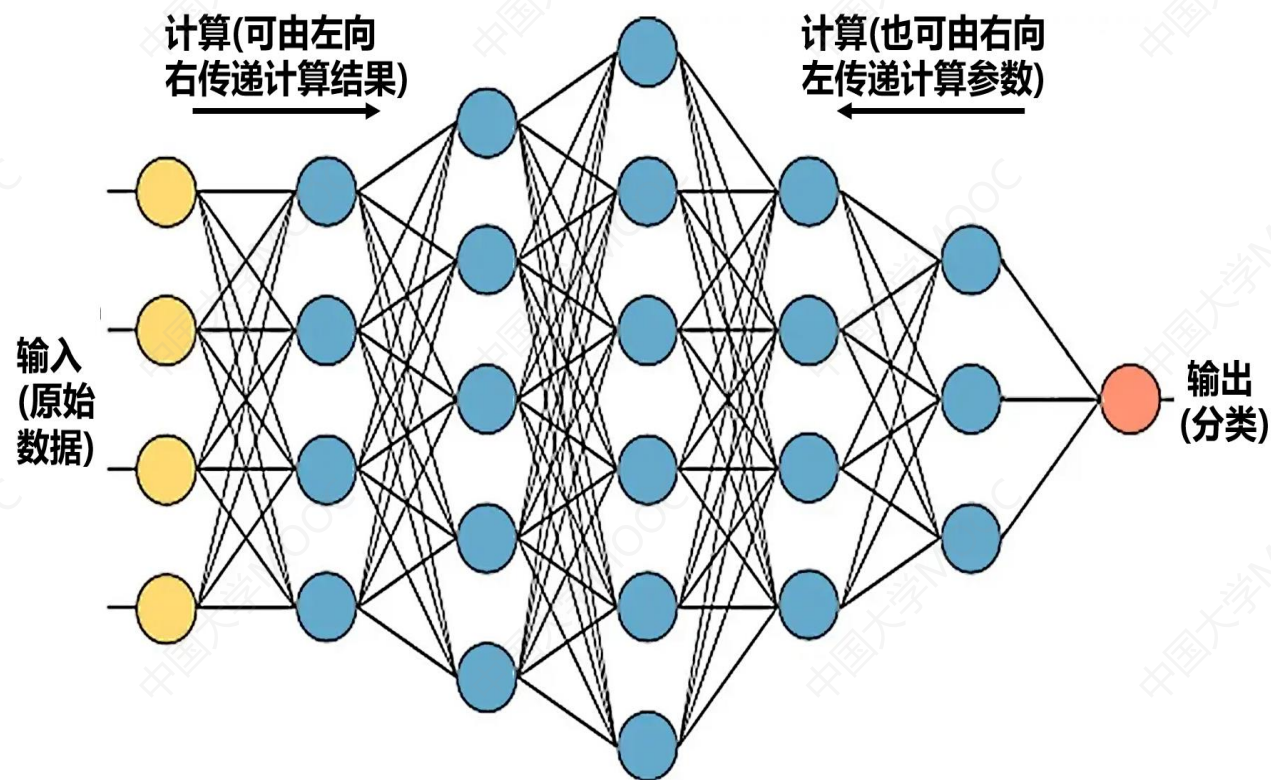
战 德 臣

哈尔滨工业大学计算学部教学委员会主任
国家教学名师

人工智能与深度学习

2

神经网络和深度学习是人工智能研究的一个重要流派



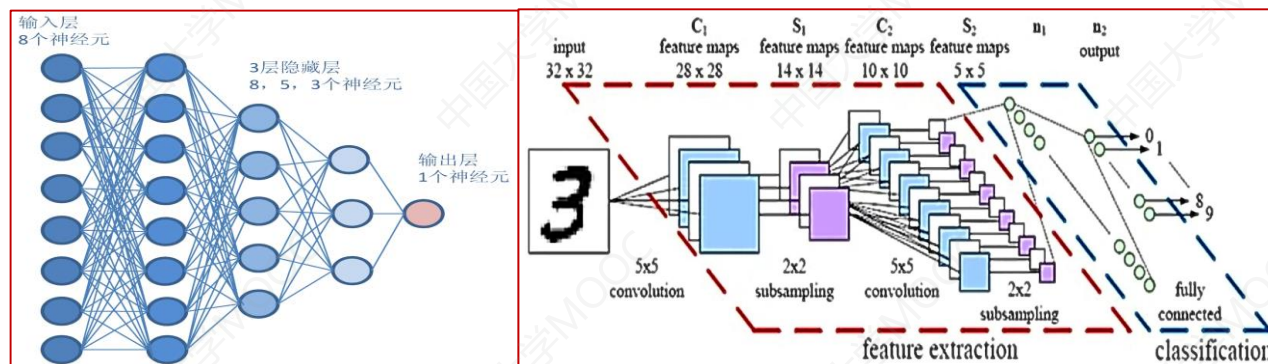
需要理解：1神经网络、2卷积、3深度神经网络

深度学习与神经网络

神经网络发展的几个阶段

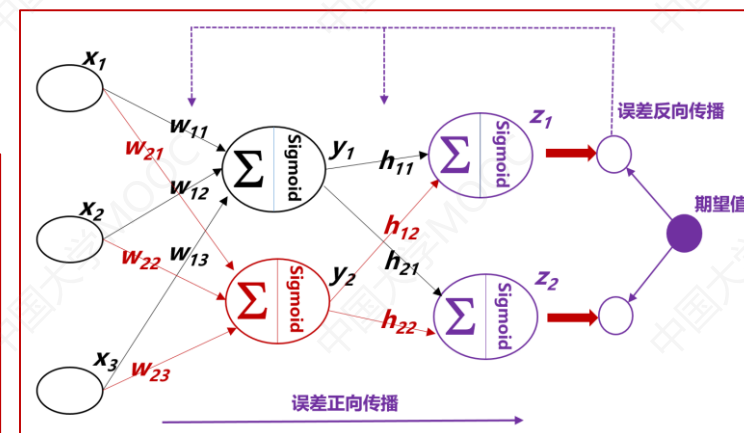
2012

多层神经网络--CNN卷积神经网络
卷积-特征提取，多层（已出现150层甚至更多）



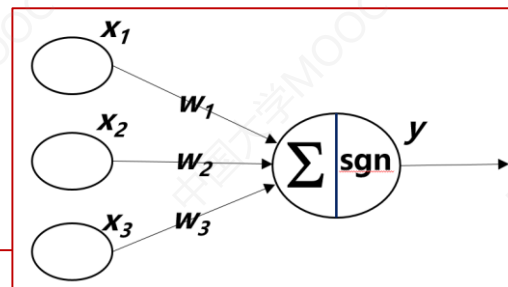
1986

两层神经网络--BP神经网络
误差反向传播



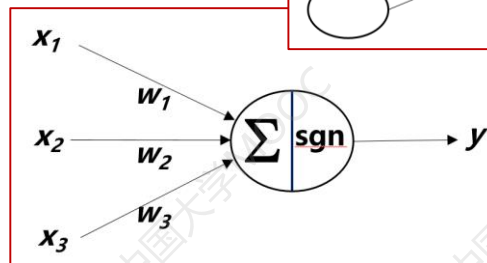
1958

单层神经网络--感知机
基本计算模型，能表示并实现基本逻辑运算如与、或、非等



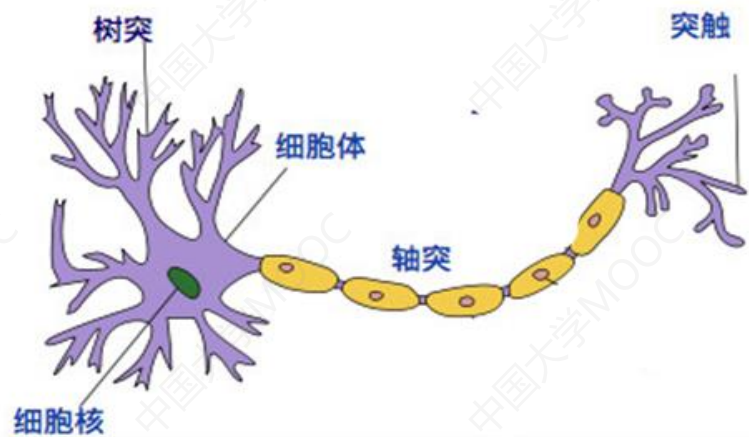
1943

神经元--MP模型
抽象模型

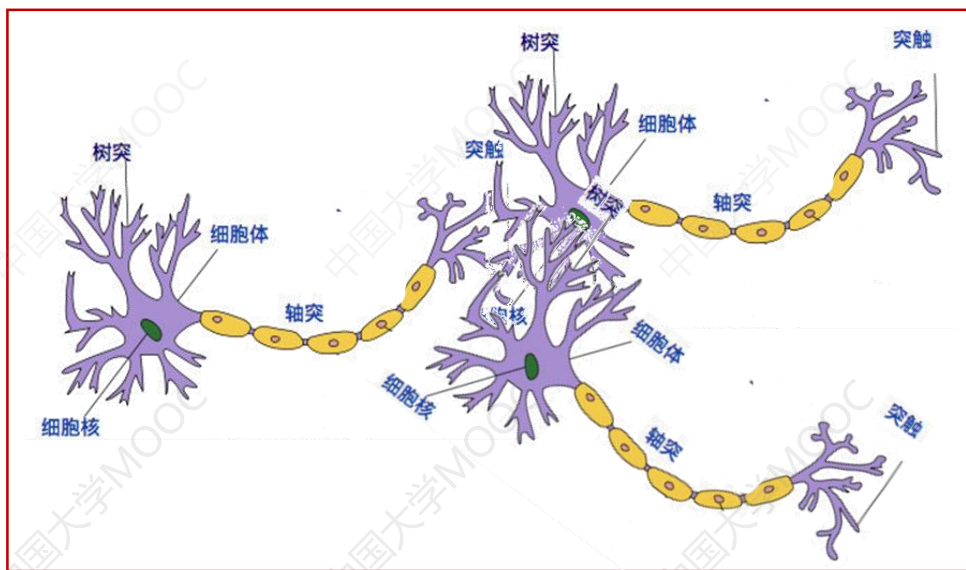


神经网络与生物神经网络

生物神经元与生物神经网络



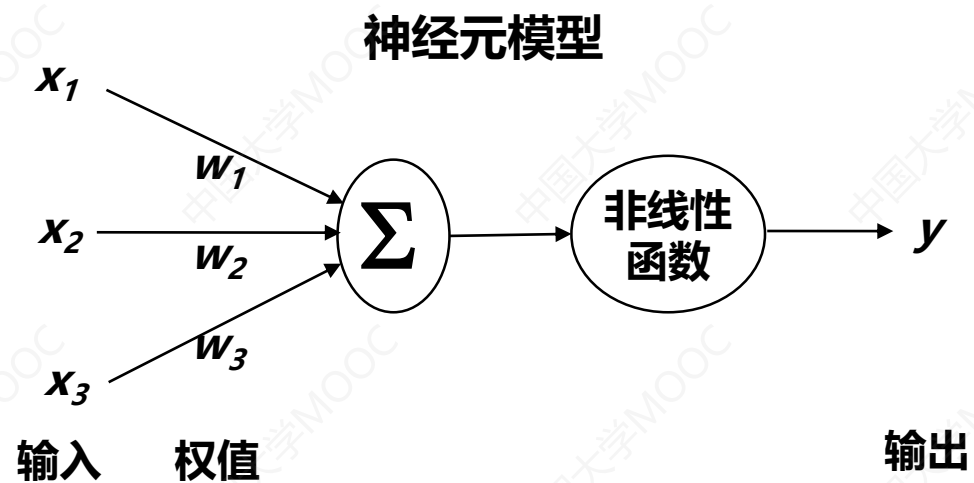
一个神经元：多个树突（输入）+1个轴突+多个轴突末梢（输出），轴突末梢与其他神经元的树突产生连接，从而传递信号。连接位置被称为“突触”。



神经元模型

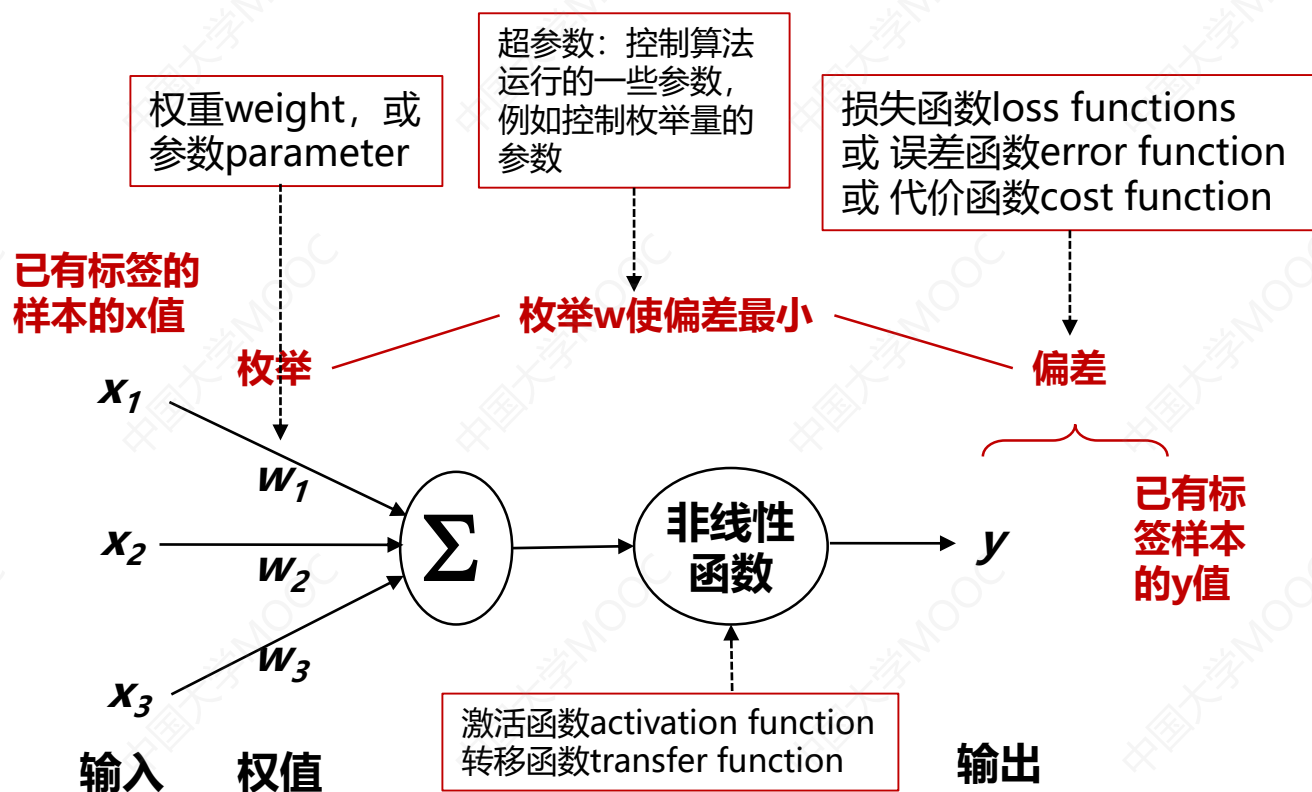
5

基本神经元模型【示例】



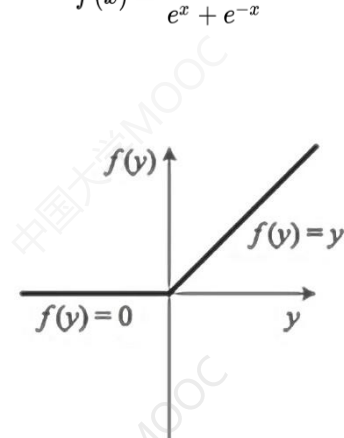
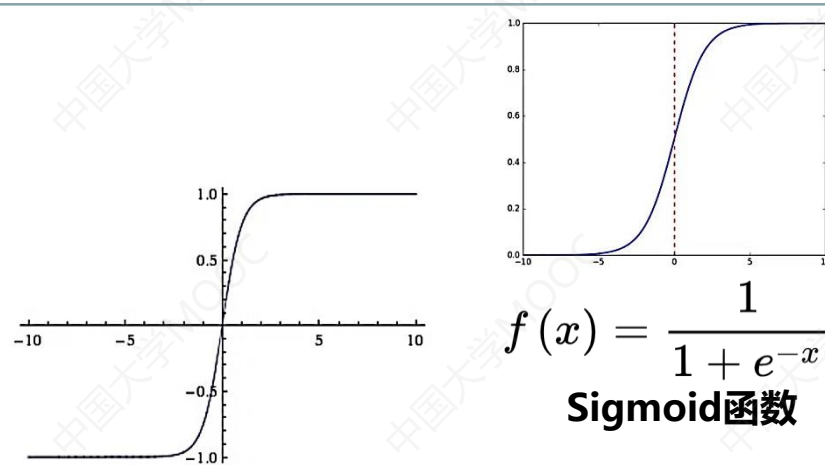
神经元模型

神经元的训练与学习

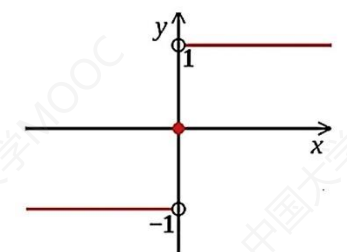


神经元模型

可依据已有标签的样本训练, 获得最优权值



$$\text{ReLU}(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases}$$



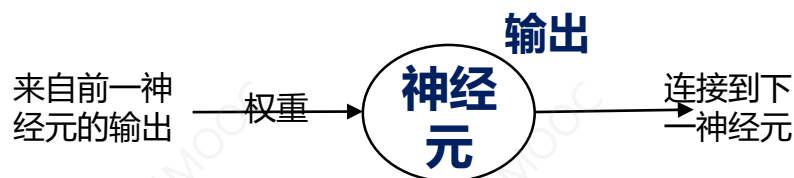
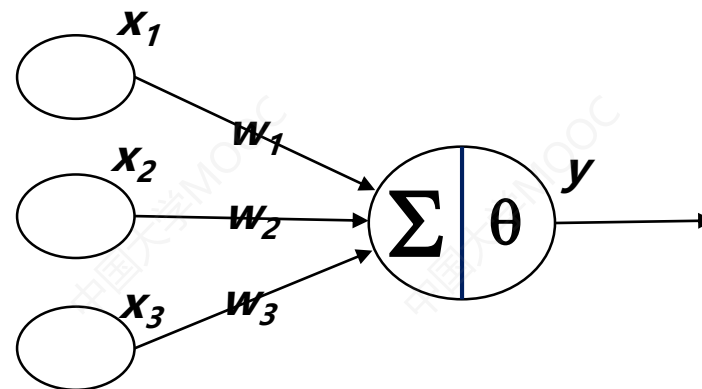
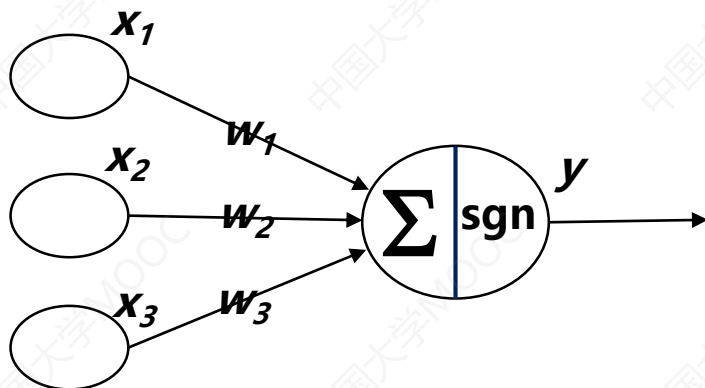
$$\text{sgn}(x) = \begin{cases} 1 & \text{当 } x > 0 \\ 0 & \text{当 } x = 0 \\ -1 & \text{当 } x < 0 \end{cases}$$

典型激活函数示例

单层神经网络—感知机

7

单层感知机



$$y = \text{sgn}(\Sigma(x_i * w_i)) = \begin{cases} 1 & \Sigma(x_i * w_i) > 0 \\ 0 & \Sigma(x_i * w_i) = 0 \\ -1 & \Sigma(x_i * w_i) < 0 \end{cases}$$

$$y = f(\Sigma(x_i * w_i) - \theta) = \begin{cases} 1 & \Sigma(x_i * w_i) - \theta \geq 0 \\ 0 & \Sigma(x_i * w_i) - \theta < 0 \end{cases}$$

θ : 阈值
输出为二值

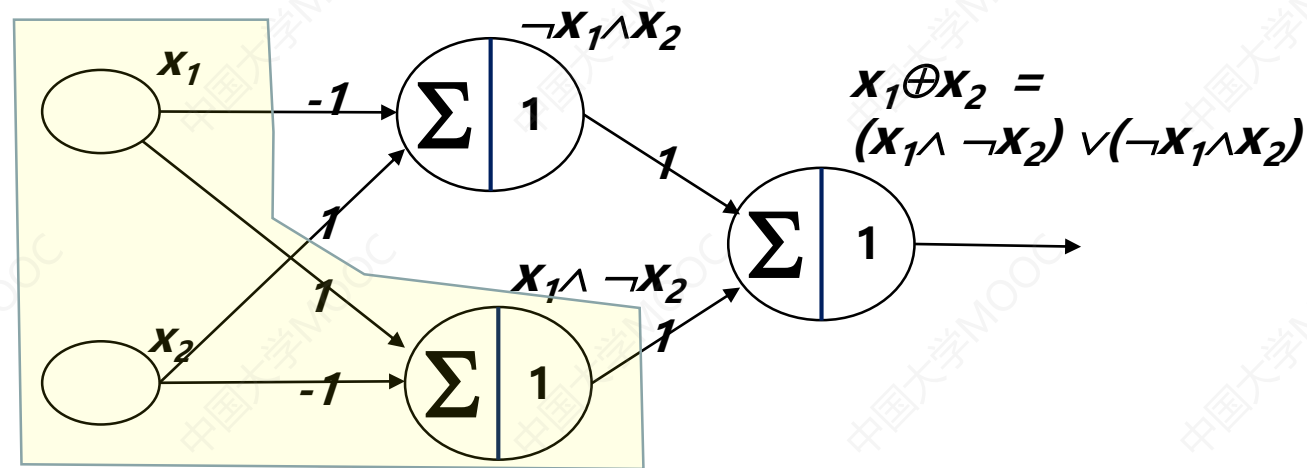
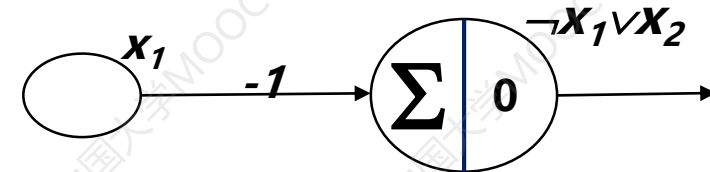
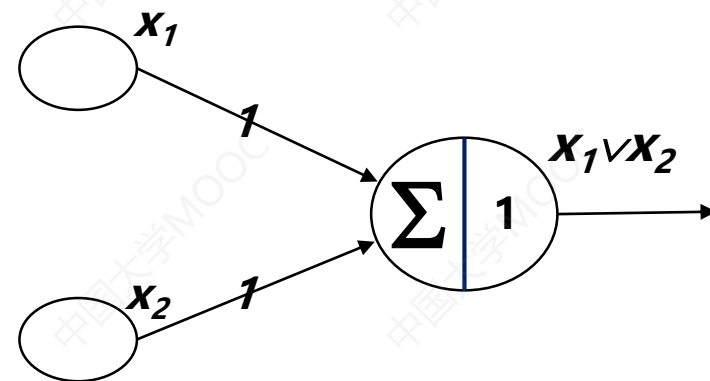
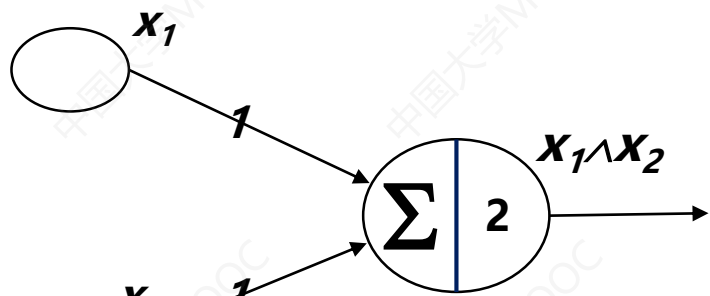
单层感知机模型

可依据有标签的样本训练，获得最优权值

单层神经网络—感知机

8

用单层神经网络表达二值逻辑运算



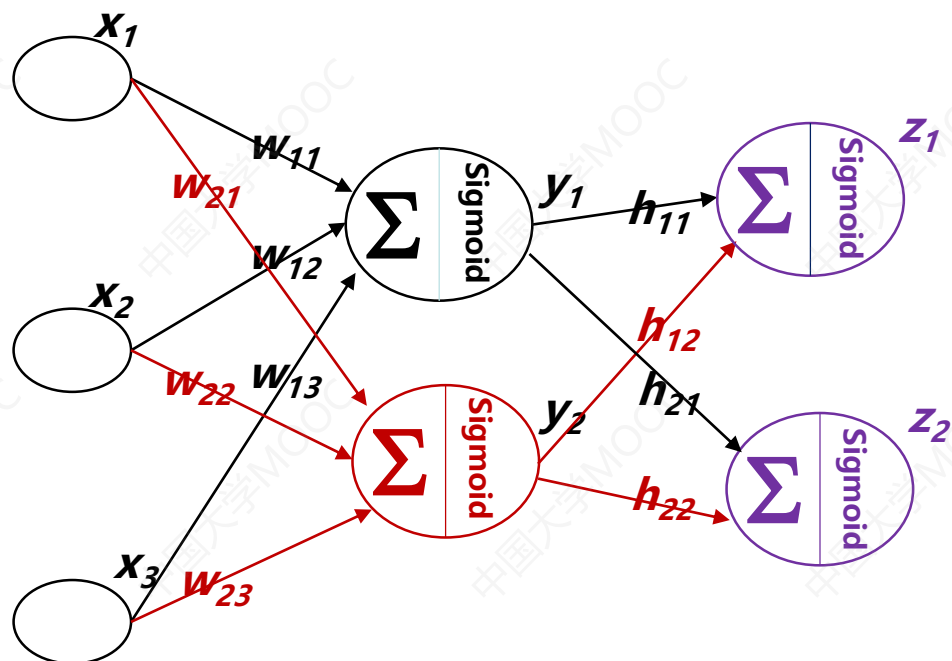
x_1	x_2	$\neg x_2$	$x_1 * 1 + x_2 * (-1)$	输出
0	0	1	0	0
0	1	0	-1	0
1	0	1	1	1
1	1	0	0	0

单层神经网络能够表达二值逻辑与或非运算，但难以表达异或运算。由此需要两层神经网络，但两层神经网络计算更为复杂。神经网络有近似任何函数的能力。

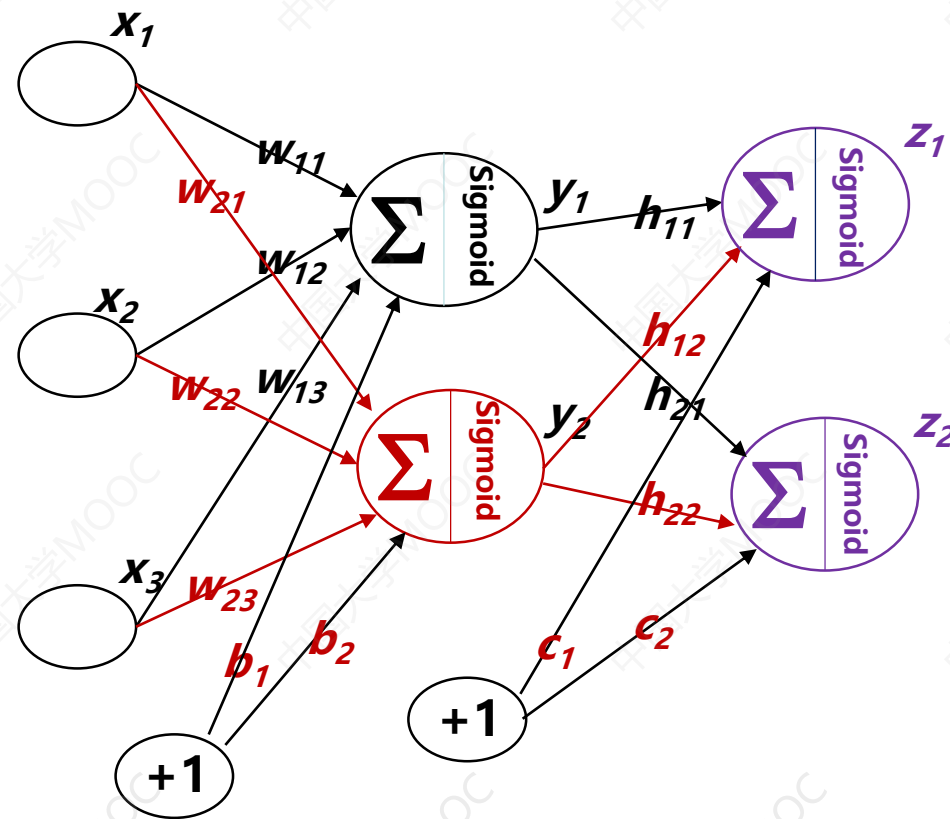
两层神经网络

9

两层神经网络模型



两层神经网络模型

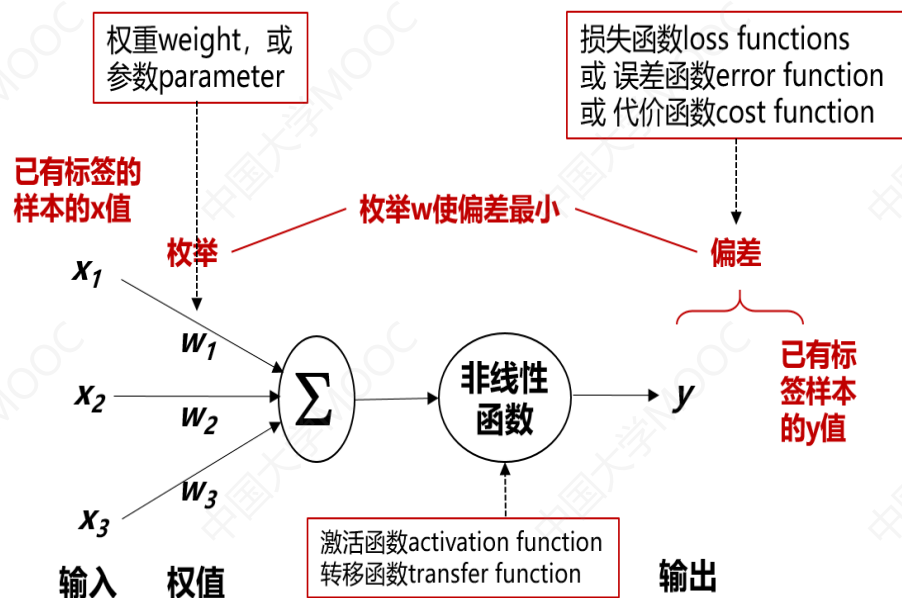


带偏置的两层神经网络模型

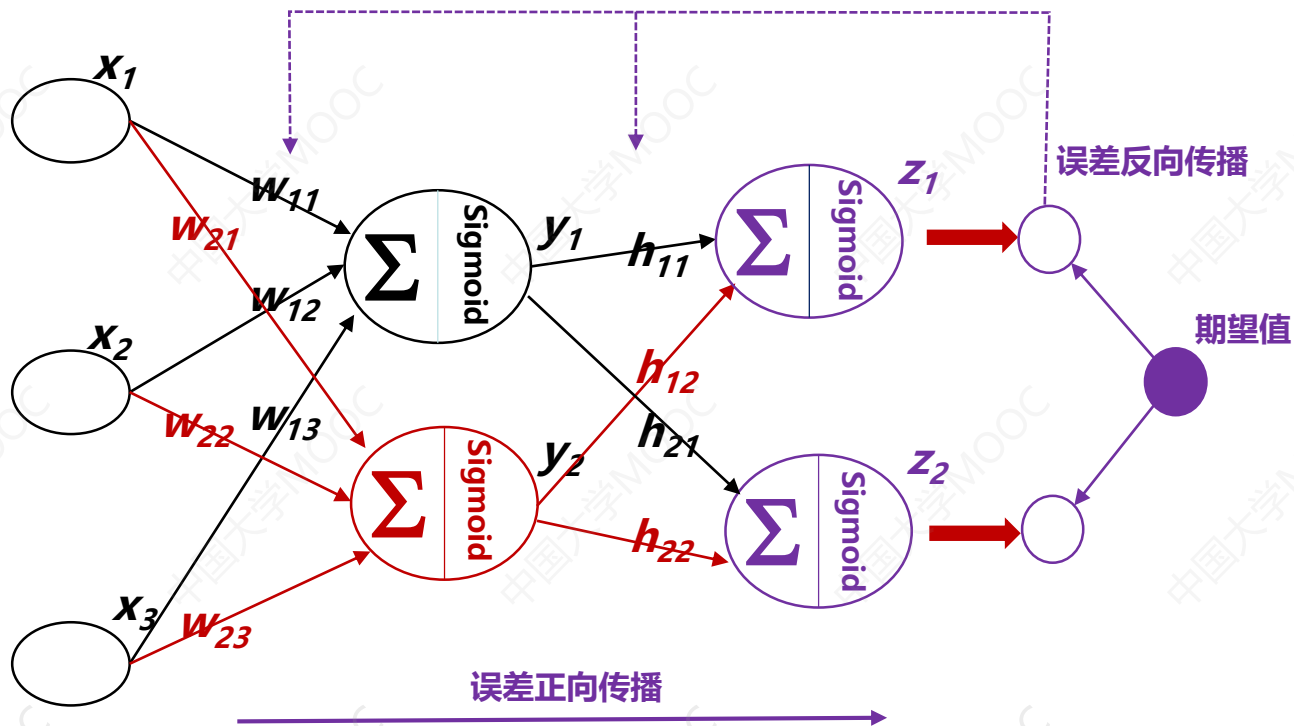
两层神经网络

10

BP神经网络：一种按照误差反向传播算法训练的多层前馈神经网络



神经网络的训练：枚举

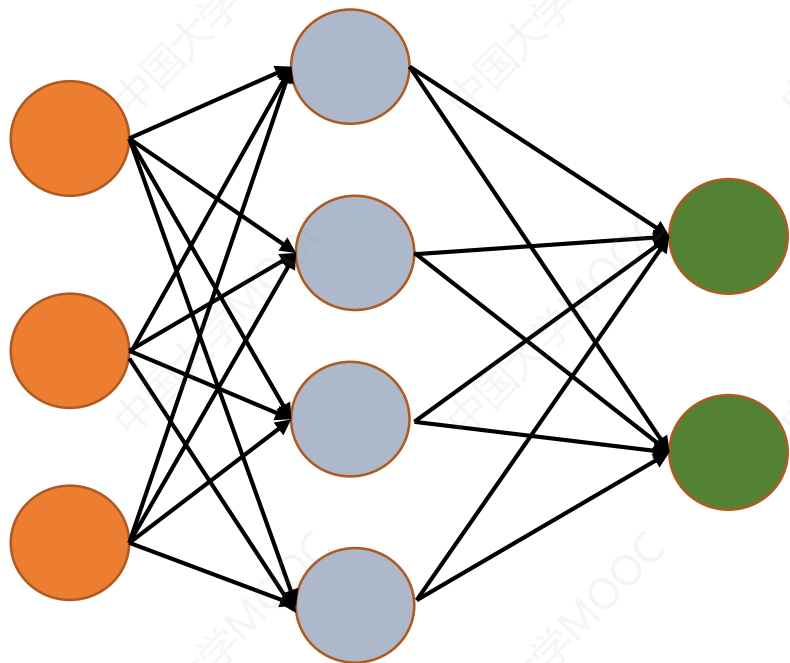


神经网络的训练：枚举（利用误差反向传播机制降低枚举量）

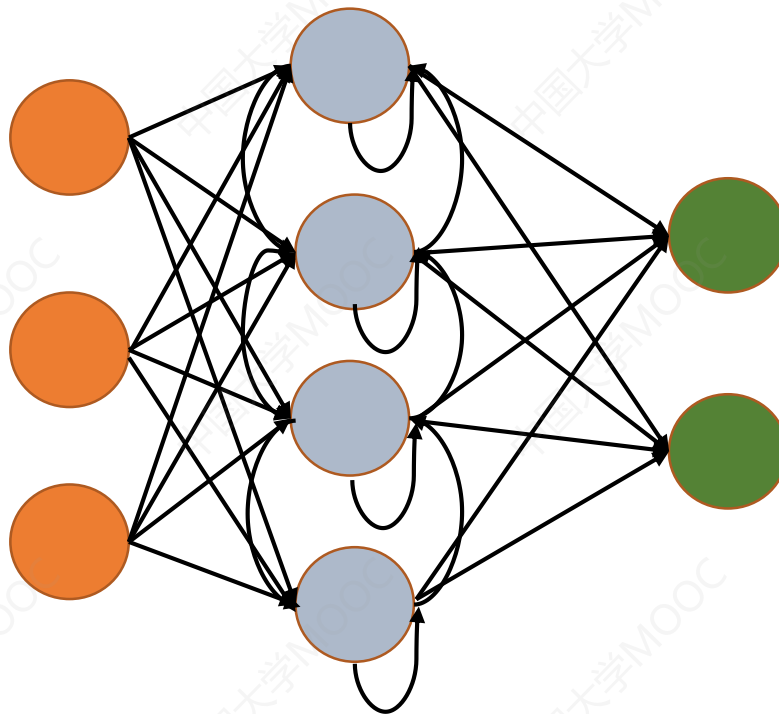
两层神经网络

11

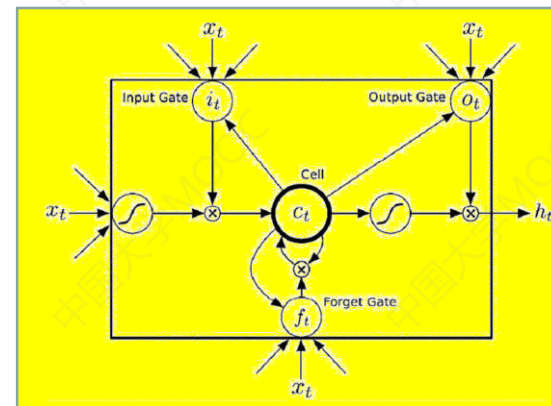
前馈神经网络与循环神经网络



前馈神经网络
仅向下一层传输计算数据



循环神经网络/递归神经网络
向下一层传输计算数据，同时也可能本层循环



循环神经网络 (RNN: Recurrent Neural Network), 如双向循环神经网络 (Bidirectional RNN) 和长短期记忆网络 (Long Short-Term Memory networks, LSTM), 递归神经网络 (recursive Neural Network)。

多层神经网络

12

将方程组表达成图的形式

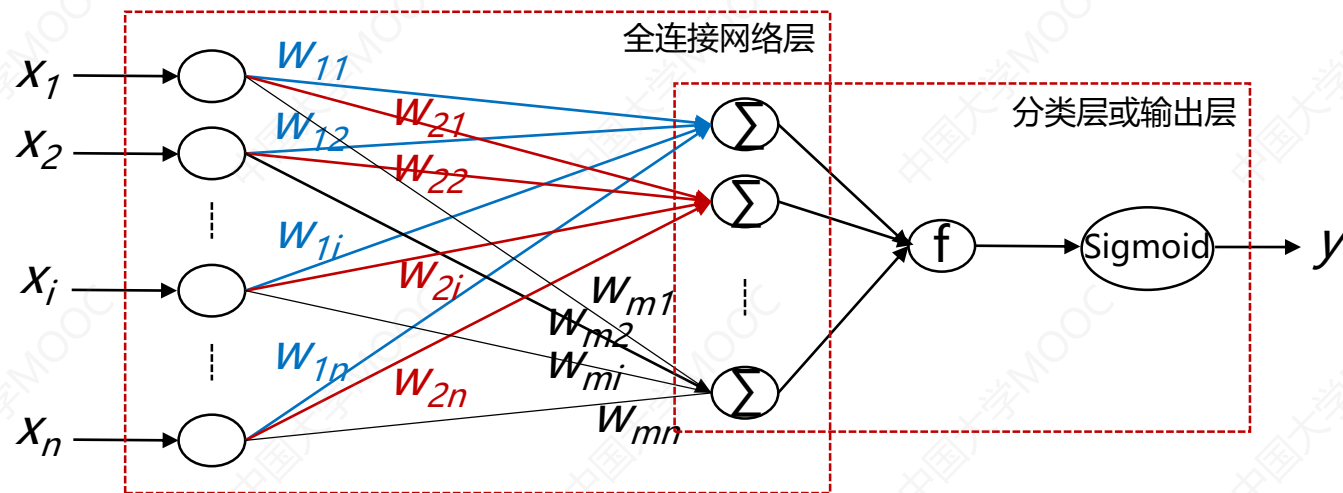
$$y_1 = w_{11}x_1 + w_{12}x_2 + \cdots + w_{1n}x_n$$

$$y_2 = w_{21}x_1 + w_{22}x_2 + \cdots + w_{2n}x_n$$

.....

$$y_m = w_{m1}x_1 + w_{m2}x_2 + \cdots + w_{mn}x_n$$

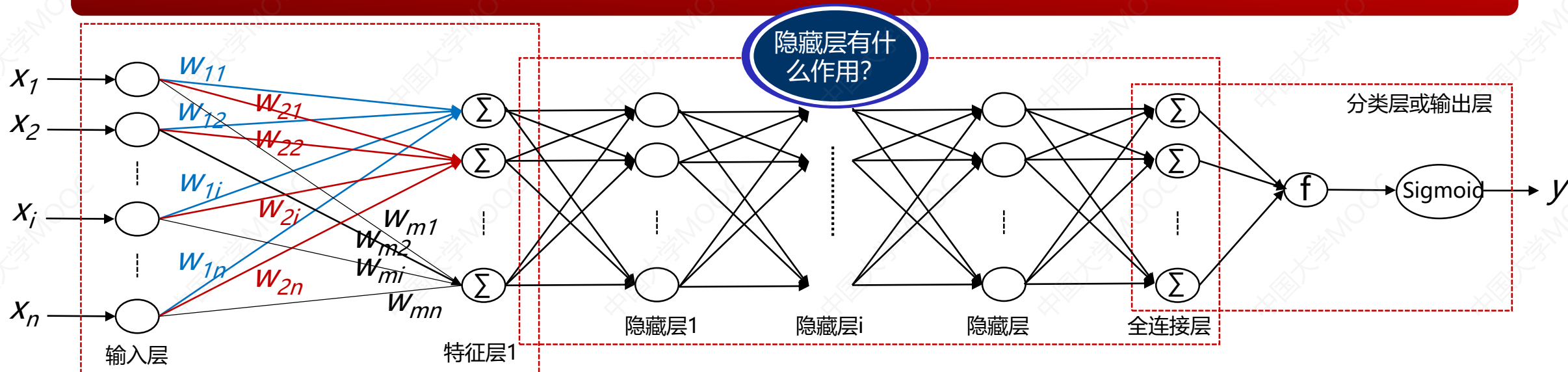
$$y = \text{Sigmoid}(f(y_1, y_2, \dots, y_m))$$



多层神经网络

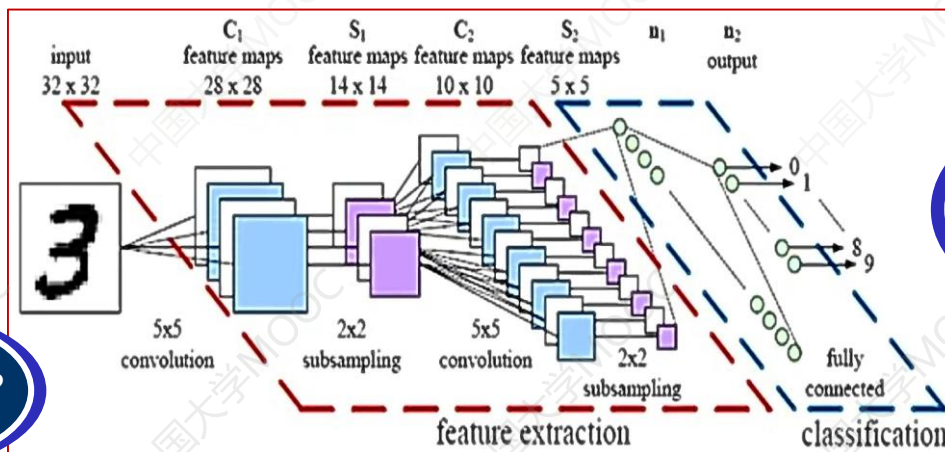
13

典型的多层神经网络：卷积神经网络



卷积神经网络： Convolutional Neural Networks, CNN

什么是卷积?

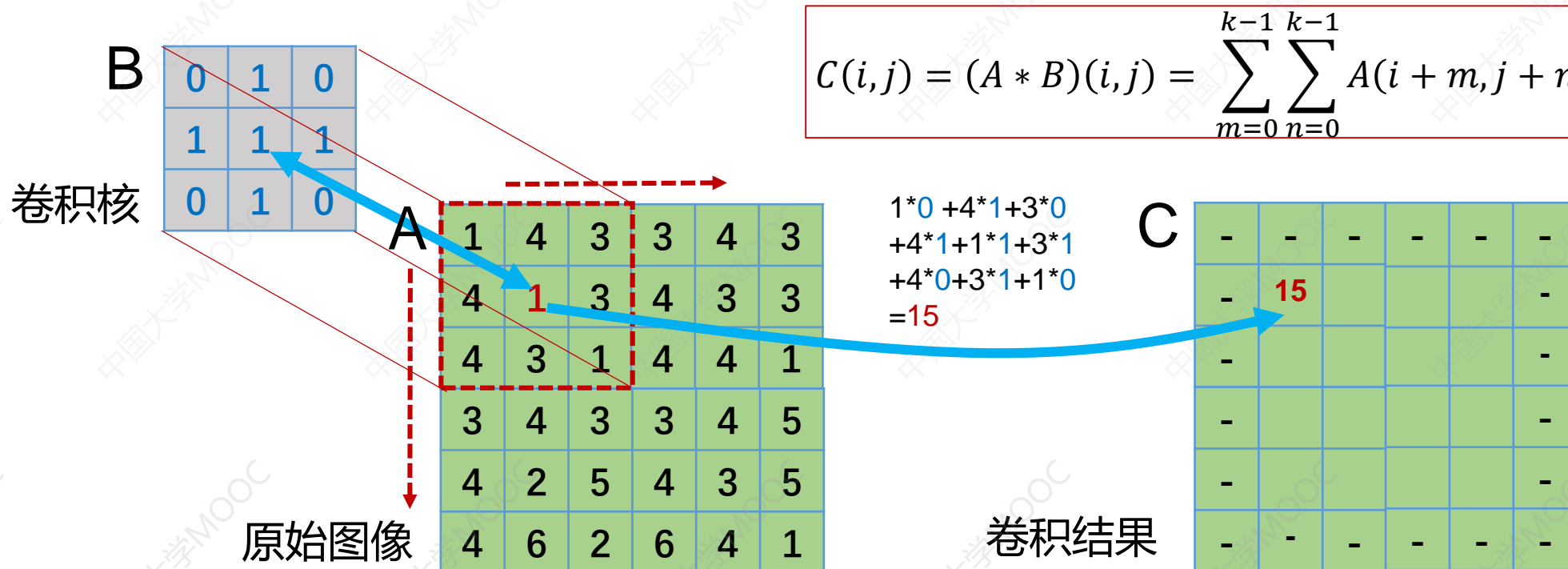


卷积神经网络的基本架构与层次

什么是卷积

【卷积】的概念与计算

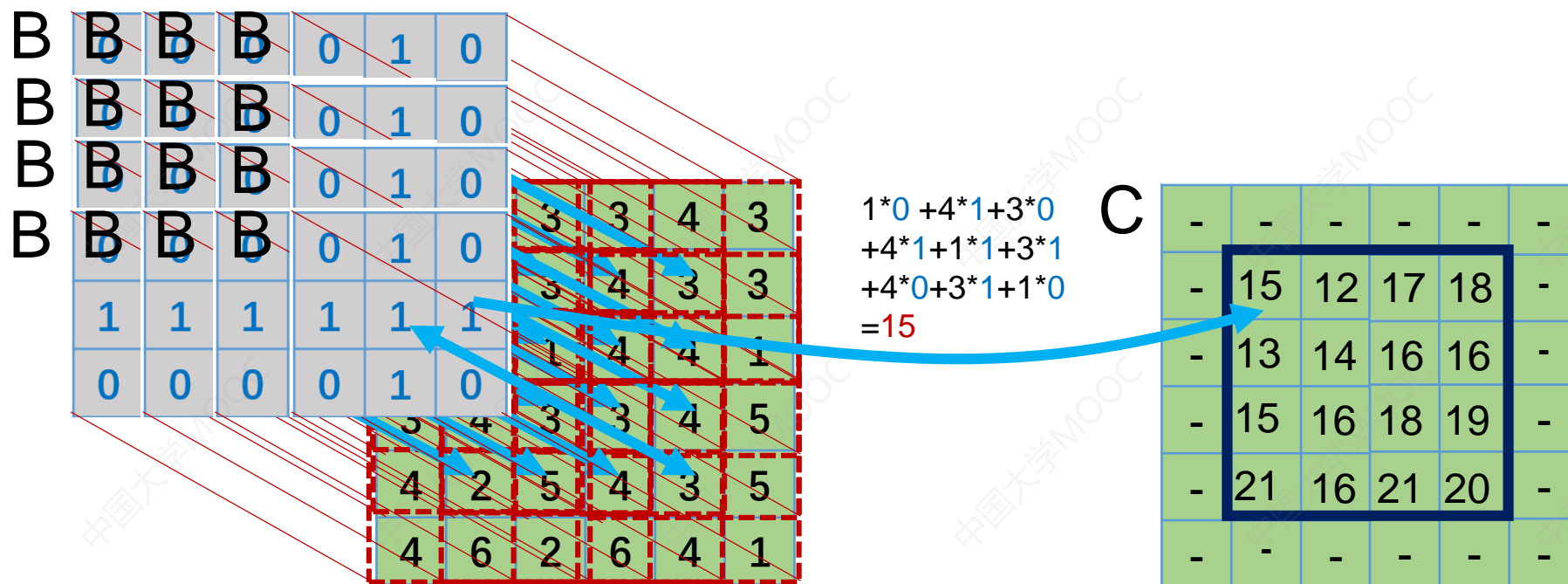
- 【卷积】是一种可作用于每一个元素上的计算。
- 可在一维数据上进行卷积，也可在二维数据上进行卷积，还可在多维数据上进行卷积。
- 有一个【卷积核】，又称为卷积【算子】，可以是 3×3 ， 5×5 ， 7×7 ，...， $k \times k$ 等大小



$$C(i, j) = (A * B)(i, j) = \sum_{m=0}^{k-1} \sum_{n=0}^{k-1} A(i + m, j + n) B(m, n)$$

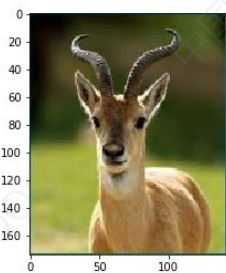
什么是卷积

【卷积】计算示例



不同的卷积核，不同的作用

【示例】卷积的作用：图像增强与特征提取



1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9



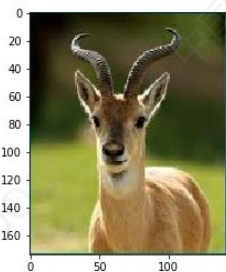
模糊



0	0.2	0
0.2	0.2	0.2
0	0.2	0

*

=



0	-1	0
-1	5	-1
0	-1	0



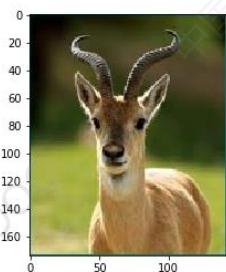
锐化



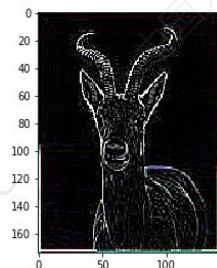
0	-1	0
-1	5	-1
0	-1	0

*

=



-1	-1	-1
-1	8	-1
-1	-1	-1



提取边缘

卷积与特征提取

17

利用【卷积】提取边缘特征：已知卷积核，进行卷积计算

-1	0	1
-2	0	2
-1	0	1

-1	-2	-1
0	0	0
1	2	1

0	1	0
1	-4	1
0	1	0

1	1	1
1	-8	1
1	1	1

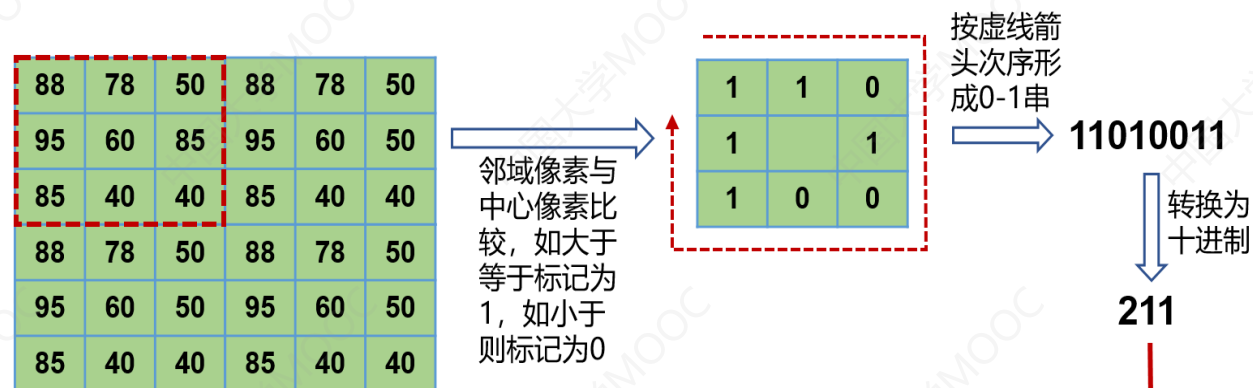
。



卷积与特征提取

改变卷积的计算方式--提取纹理特征

LBP (Local Binary Pattern, 局部二值模式) 是一种用来描述图像局部纹理特征的算子, 具有旋转不变性和灰度不变性等优点



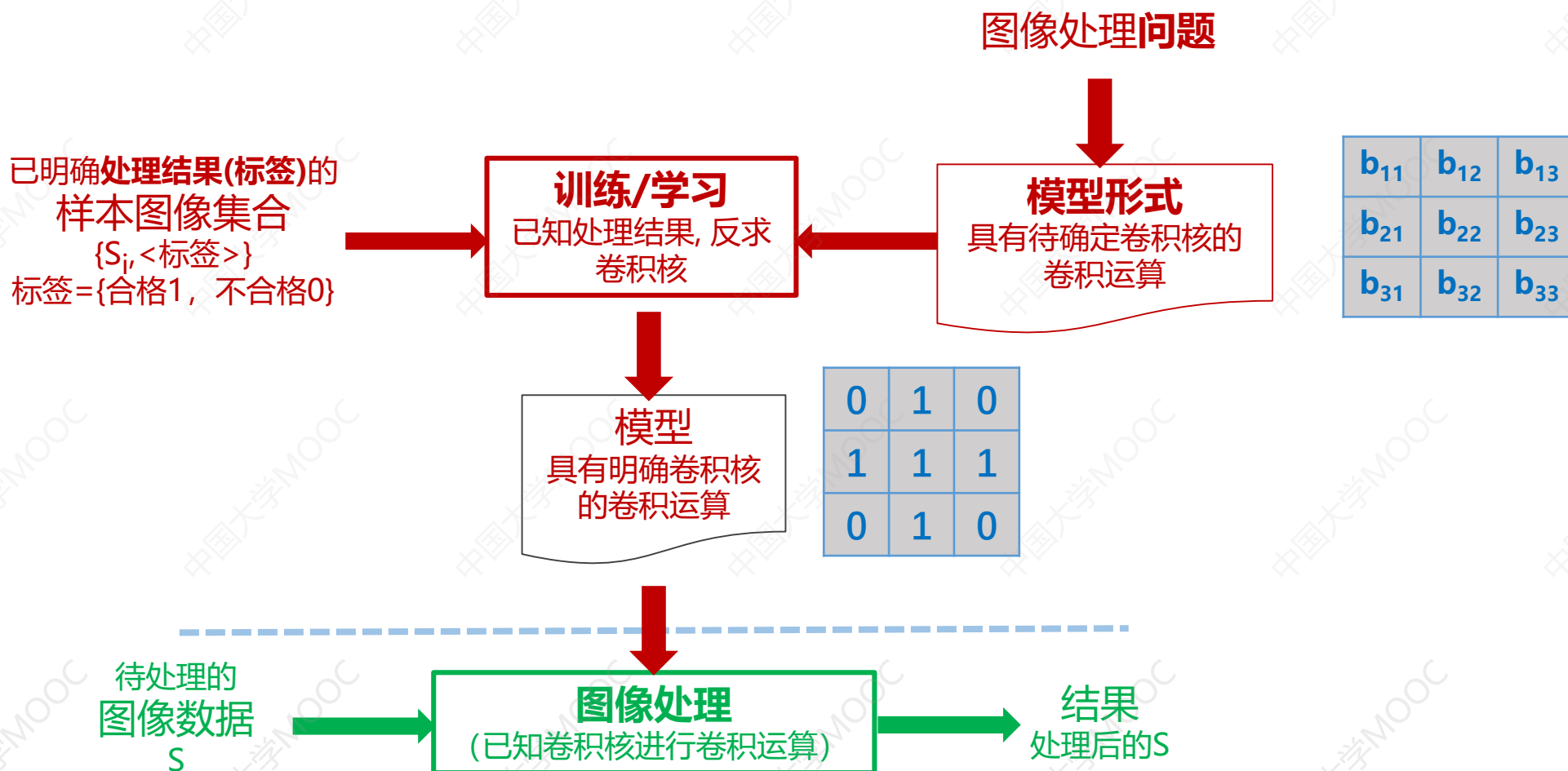
原始图像

纹理特征图像



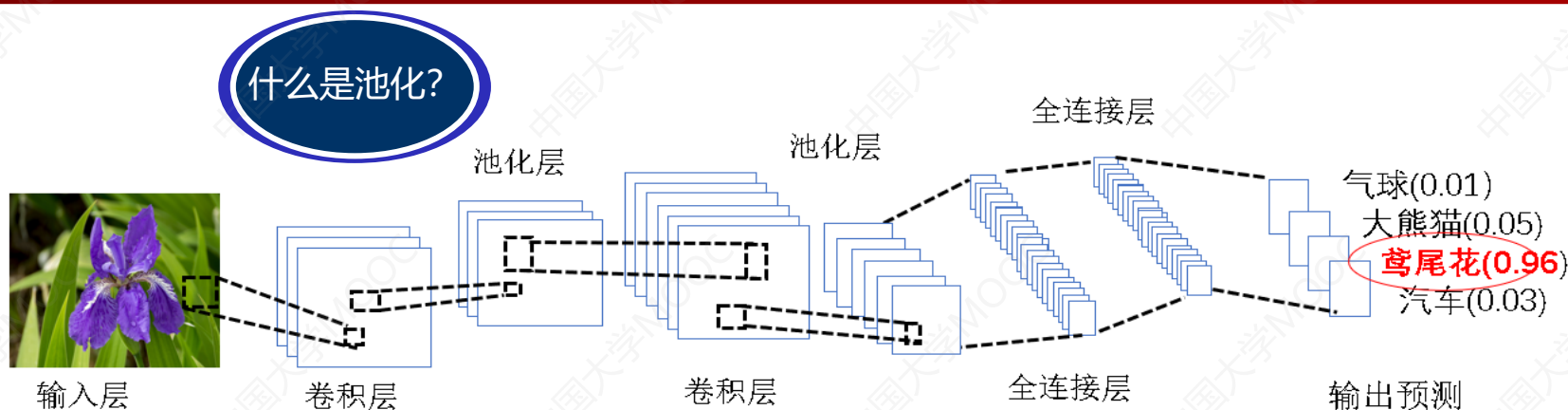
机器学习中的卷积

依据有标签的样本数据，反求【卷积核】

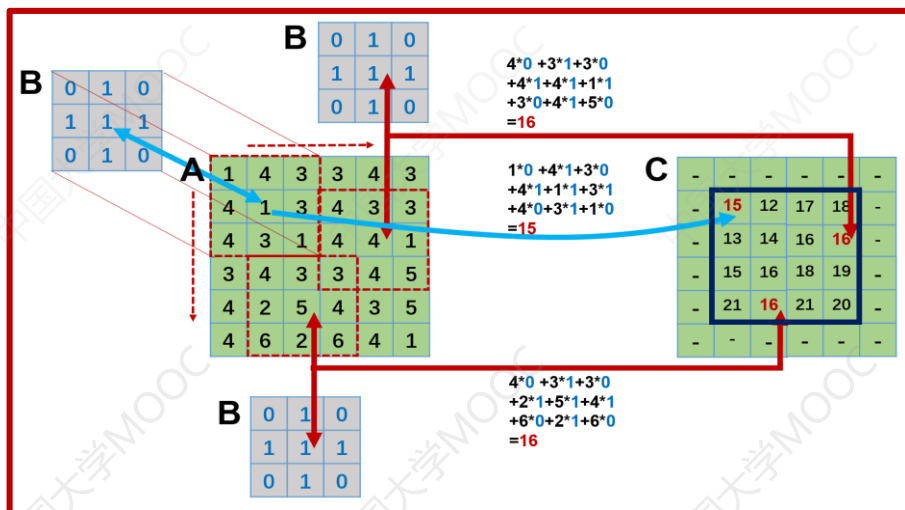


深度学习--深度神经网络

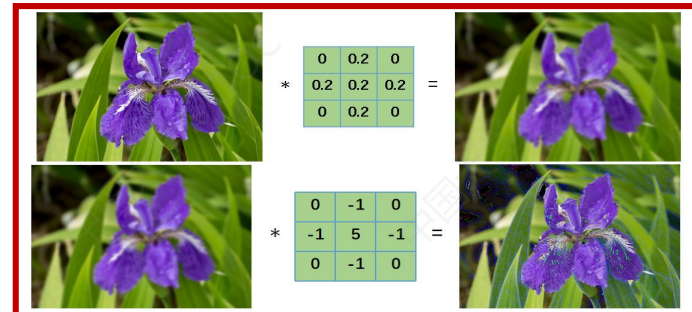
卷积神经网络：卷积层—特征提取



卷积



提取特征



深度学习--深度神经网络

卷积神经网络：池化层—降维处理

【示例】通过池化层，将 6×6 的特征像素矩阵，降低为 2×2 的池化特征像素矩阵

1	4	3	3	4	3
4	1	3	4	3	3
4	3	1	4	4	1
3	4	3	3	4	5
4	2	5	4	3	5
4	6	2	6	4	1

将左侧像素矩阵划分为四个区域，可以区域内像素的最大值作为区域像素值——最大池化

4	4
6	6

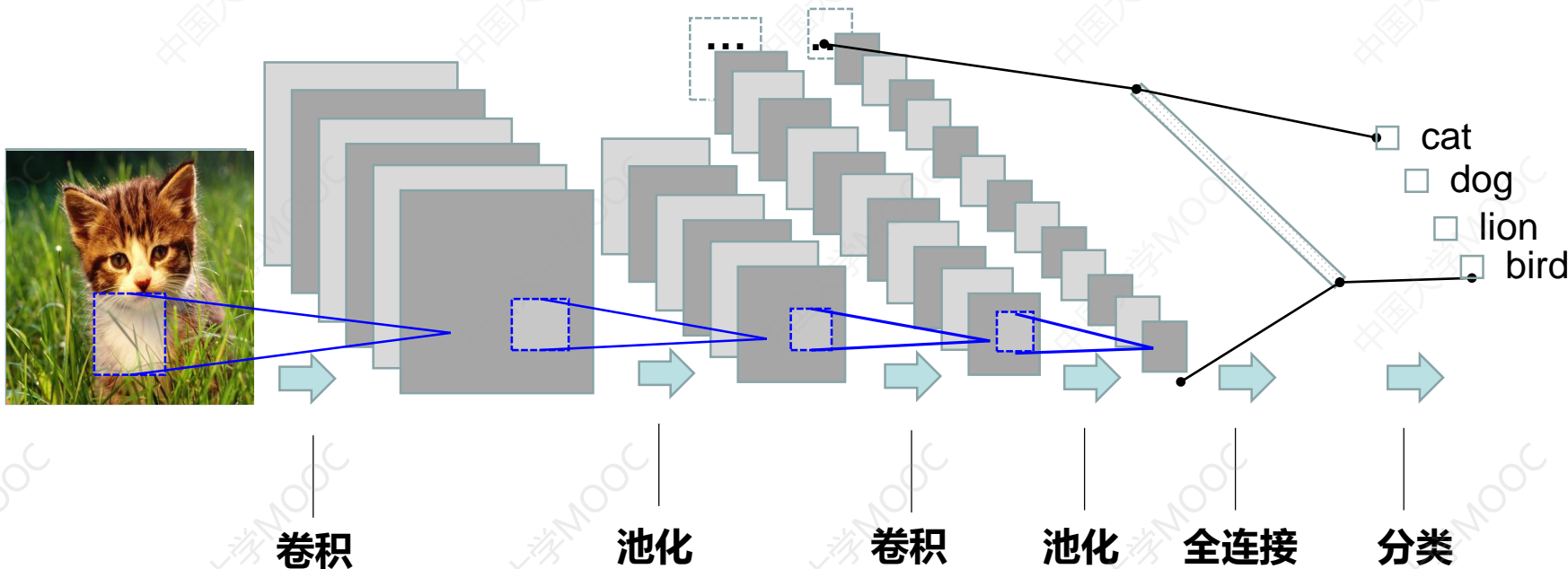
将左侧像素矩阵划分为四个区域，也可以区域内像素的平均值作为区域像素值——平均池化

2.67	3.22
3.67	3.89

深度学习--深度神经网络

计算机如何使用卷积神经网络精准识别图中的动物：深度神经网络

以卷积、激活函数和池化层为基本结构，多次重复构成深度神经网络



- 卷积神经网络通过逐层处理，提取出对图像分类起关键作用的特征。
- 包括如下处理类型：
 - 卷积；
 - 降采样/池化；
 - 等等

神经网络与深度学习

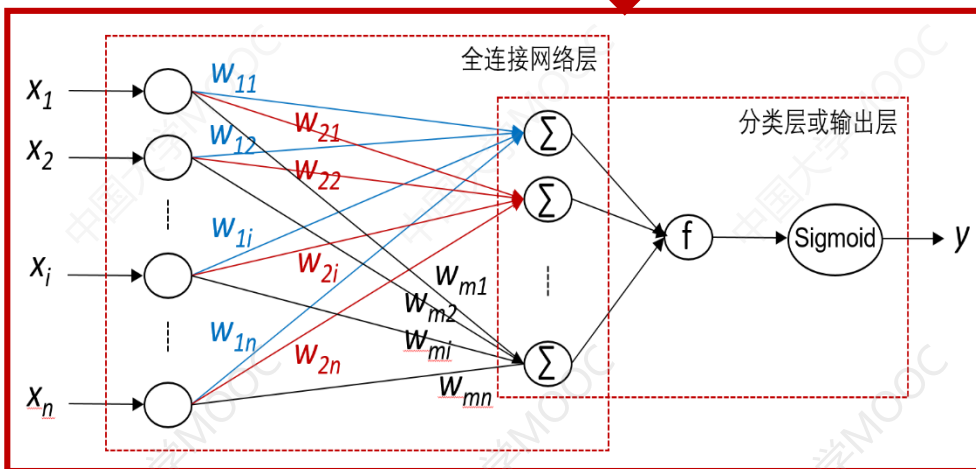
23

小结

分类方程组

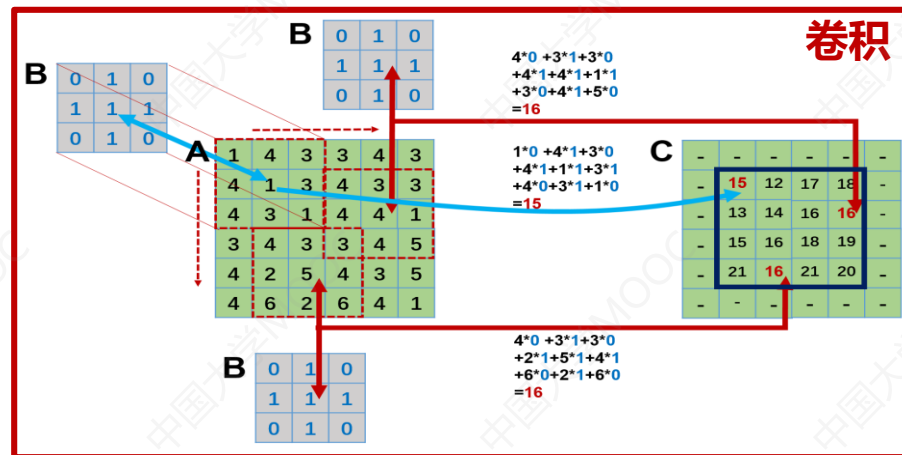
$$\begin{aligned}y_1 &= w_{11}x_1 + w_{12}x_2 + \dots + w_{1n}x_n \\y_2 &= w_{21}x_1 + w_{22}x_2 + \dots + w_{2n}x_n \\&\dots\dots\dots \\y_m &= w_{m1}x_1 + w_{m2}x_2 + \dots + w_{mn}x_n\end{aligned}$$
$$y = \text{Sigmoid}(f(y_1, y_2, \dots, y_m))$$

神经网络

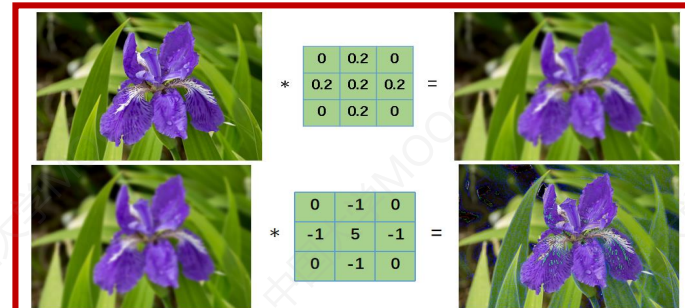


深度学习，增加神经网络的层数

卷积



已知卷积核，
提取特征



已知分类结果，反求
卷积核—机器学习

