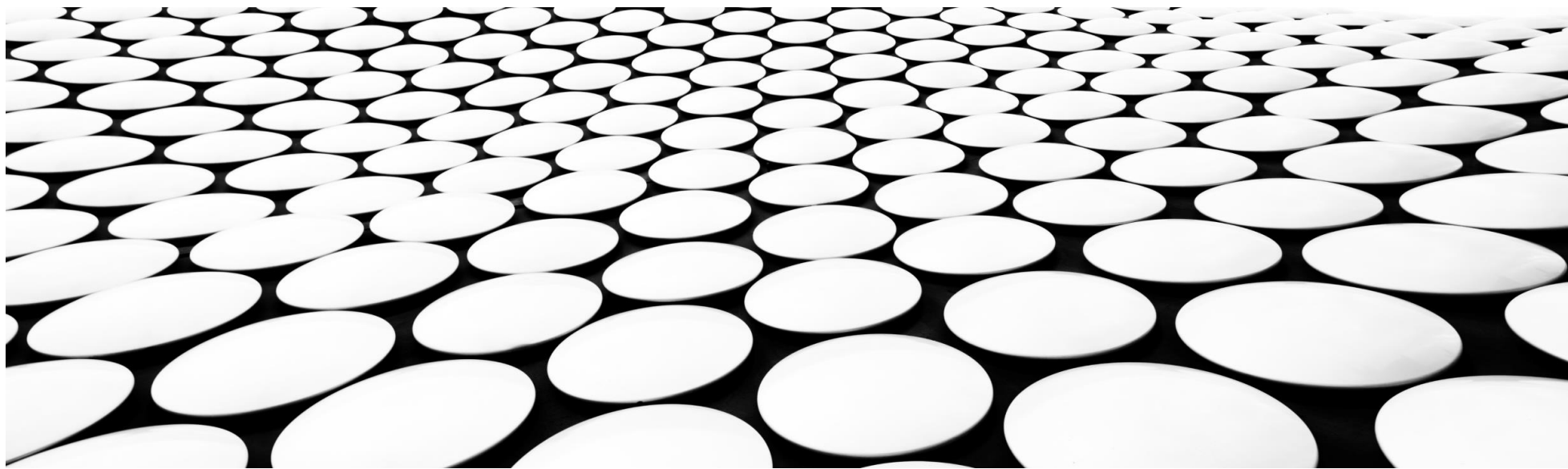


# 深度学习

邱怡轩



---

# 今天的主题

- 卷积神经网络

# 卷积的概念

- 先看一道初中数学题
- 考虑两个多项式,  $p(x) = 3 - 2x + x^2$ ,  
 $q(x) = -5 + 4x$
- 问  $r(x) = p(x) \cdot q(x)$  的展开式是什么?

# 卷积的概念

- 再看一道“大学”算法题
- 一般地，用一个向量表示多项式的系数
- 如  $v = (1, 0, 2)'$  表示  $1 + 2x^2$ ,  $w = (0, 1, 2, 3)'$  表示  $x + 2x^2 + 3x^3$
- 给定两个任意的多项式系数  $v$  和  $w$ , 求多项式乘积的系数向量  $r$

# 卷积的概念

- 先说结论

- R

```
> v = c(3, -2, 1)
> w = c(-5, 4)
> convolve(v, rev(w), type = "open")
[1] -15  22 -13  4
```

- Python

```
In [1]: import numpy as np
v = [3, -2, 1]
w = [-5, 4]
np.convolve(v, w)
```

```
Out[1]: array([-15,  22, -13,  4])
```

# 卷积的概念

- 先说结论

- R

```
> v = c(3, -2, 1)
> w = c(-5, 4)
> convolve(v, rev(w), type = "open")
[1] -15  22 -13   4
```

- Python

```
In [1]: import numpy as np
        v = [3, -2, 1]
        w = [-5, 4]
        np.convolve(v, w)

Out[1]: array([-15,  22, -13,   4])
```

# 卷积的概念

**convolve**



英 [kən'vɒlv]



美 [kən'vɑːlv]



v. 使卷曲，使缠绕；卷积

[ 第三人称单数 convolves 现在分词 convolving 过去式 convolved 过去分词 convolved ]

同近义词

同根词

词根: **convolve**

adj.

**convoluted** 复杂的；费解的；旋绕的

n.

**convolution** [数] 卷积；回旋；盘旋；卷绕

v.

**convoluted** 盘绕；缠绕 (convolute的过去分词)

# 一维卷积

■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$

$v$

1	$x$	$x^2$
3	-2	1

$w$

1	$x$
-5	4



# 一维卷积

■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$

$v$

1	$x$	$x^2$
3	-2	1

$v$

1	$x$	$x^2$
3	-2	1

$w$

1	$x$
-5	4

$\text{rev}(w)$

$x$	1
4	-5

# 一维卷积

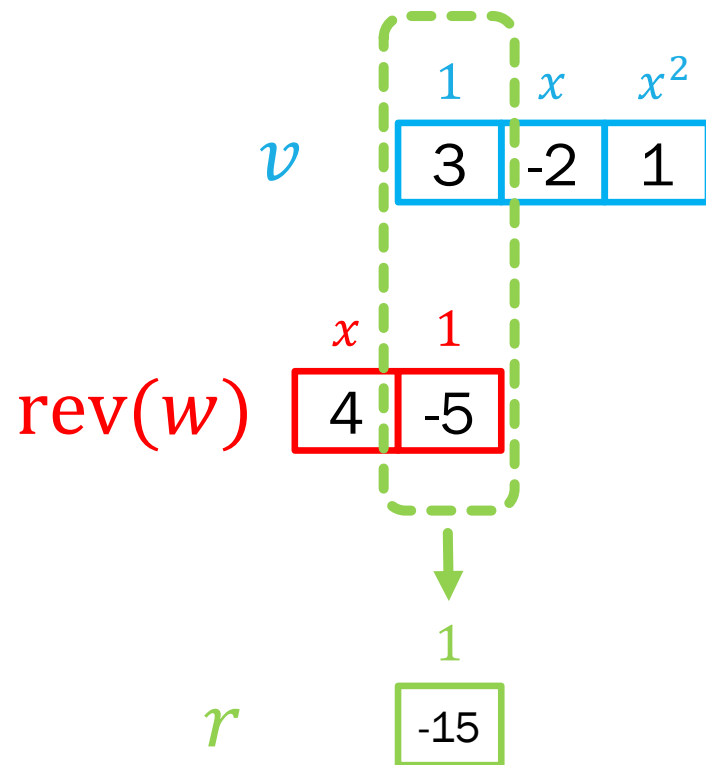
■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$

$$v \quad \begin{array}{|c|c|c|} \hline 1 & x & x^2 \\ \hline 3 & -2 & 1 \\ \hline \end{array}$$

$$\text{rev}(w) \quad \begin{array}{|c|c|} \hline x & 1 \\ \hline 4 & -5 \\ \hline \end{array}$$

# 一维卷积

■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$



# 一维卷积

■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$

$v$ 

$1$	$x$	$x^2$
3	-2	1

$\text{rev}(w)$ 

$x$	$1$
4	-5

→

$r$ 

$1$
-15

# 一维卷积

■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$

$v$ 

1	$x$	$x^2$
3	-2	1

$\text{rev}(w)$ 

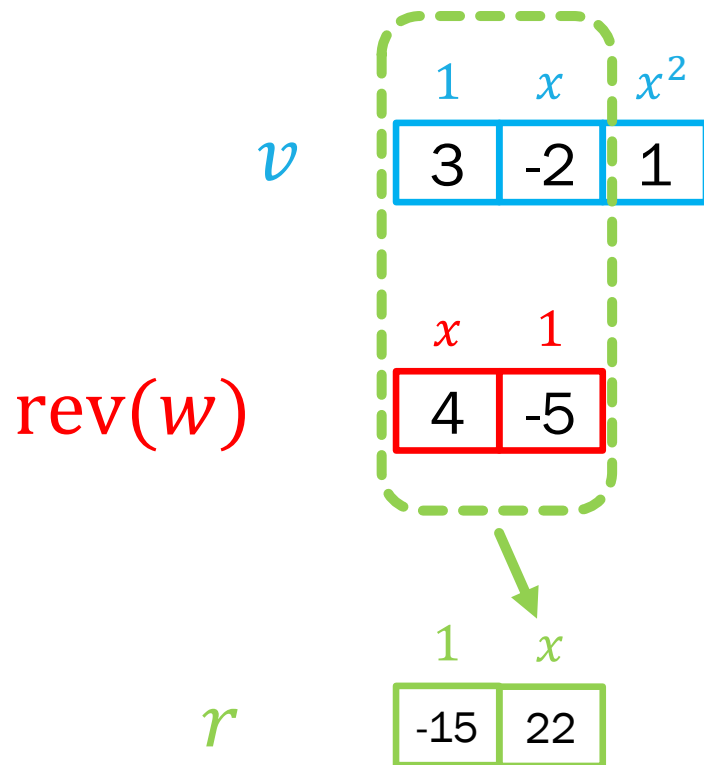
$x$	1
4	-5

$r$ 

1
-15

# 一维卷积

■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$



# 一维卷积

■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$

$v$ 

1	$x$	$x^2$
3	-2	1

$\text{rev}(w)$ 

$x$	1
4	-5

 $\rightarrow$

$r$ 

1	$x$
-15	22

# 一维卷积

■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$

$v$

1	$x$	$x^2$
3	-2	1

$\text{rev}(w)$

$x$	1
4	-5

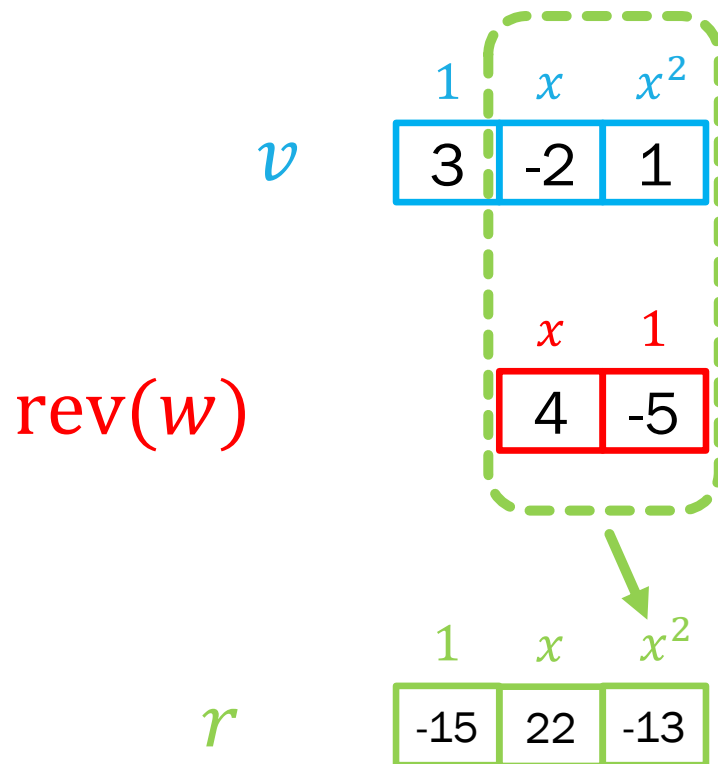
$r$

1	$x$
-15	22



# 一维卷积

■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$



# 一维卷积

■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$

$v$

1	$x$	$x^2$
3	-2	1

$\text{rev}(w)$

$x$	1
4	-5



$r$

1	$x$	$x^2$
-15	22	-13

# 一维卷积

■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$

$v$

1	$x$	$x^2$
3	-2	1

$\text{rev}(w)$

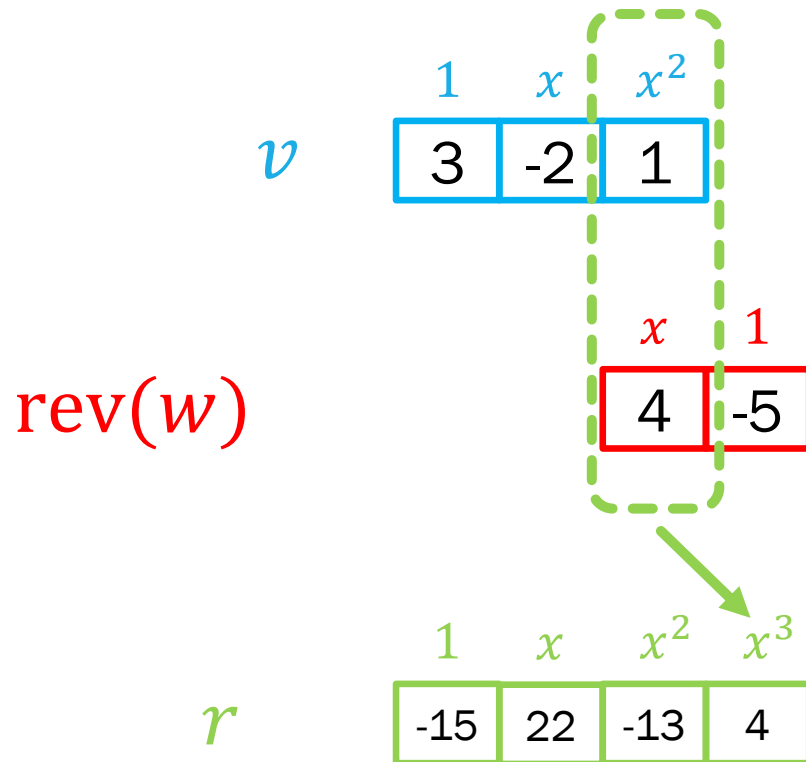
$x$	1
4	-5

$r$

1	$x$	$x^2$
-15	22	-13

# 一维卷积

■  $p(x) = 3 - 2x + x^2$ ,  $q(x) = -5 + 4x$



# 数学定义

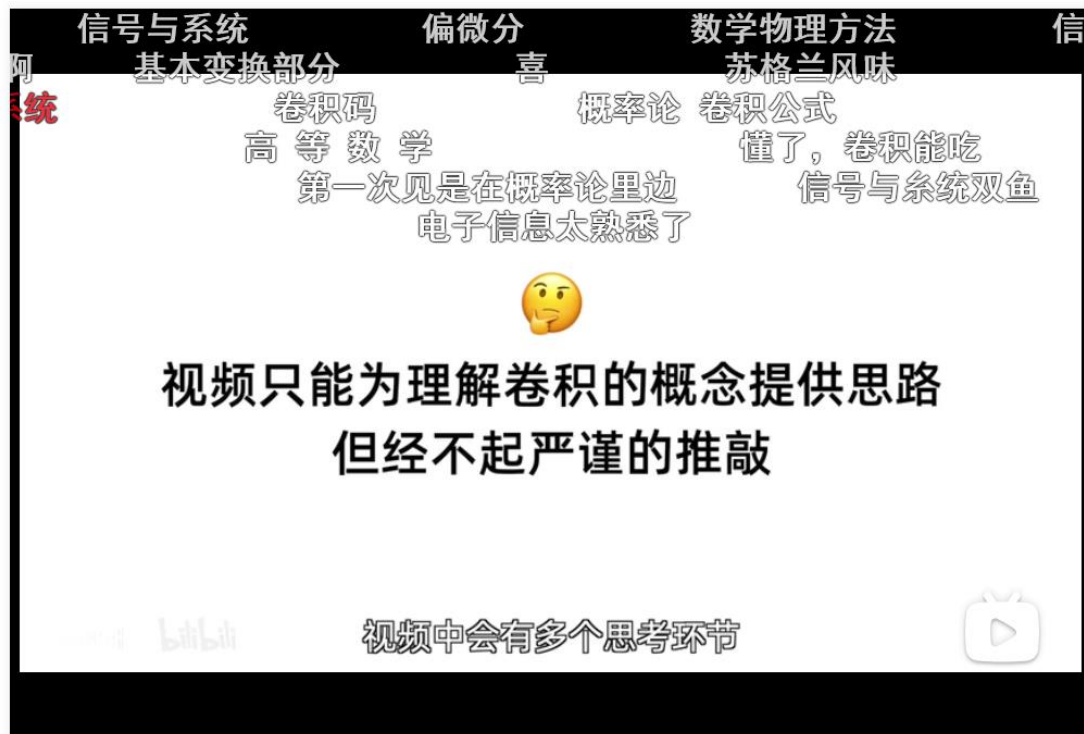
- $v = (v_1, \dots, v_m)', \quad w = (w_1, \dots, w_n)'$
- $r = v * w$
- $r_i = \sum_{k=-\infty}^{\infty} v_k w_{i-k+1}$
- 没有定义的元素用 0 代替

# 延伸学习

- <https://www.bilibili.com/video/BV1JX4y1K7Dr/>

活动作品 卷积究竟卷了啥? ——17分钟了解什么是卷积

112.8万播放 · 3344弹幕 2021-01-22 20:00:14 全站排行榜最高第44名



# 二维卷积

1	2	3	4
5	6	7	8
9	-1	-2	-3
-4	-5	-6	-7

**V**

**W**

0.1	0.2	0.3
0.4	0.5	0.6
0.7	0.8	0.9

卷积核  
(convolutional kernel) /  
滤波器  
(filter)

# 二维卷积

0	0	0	0	0	0
0	1	2	3	4	0
0	5	6	7	8	0
0	9	-1	-2	-3	0
0	-4	-5	-6	-7	0
0	0	0	0	0	0

**V**

**W**

0.1	0.2	0.3
0.4	0.5	0.6
0.7	0.8	0.9

**W'**

0.9	0.8	0.7
0.6	0.5	0.4
0.3	0.2	0.1



# 二维卷积

0	0	0	0	0	0
0.9	0.8	0.7			
0.6	1	2	3	4	0
0.3	5	6	7	8	0
	9	-1	-2	-3	0
	-4	-5	-6	-7	0
0	0	0	0	0	0

**V**

**R**


# 二维卷积

0	0	0	0	0	0
	0.9	0.8	0.7		
0	1	2	3	4	0
	0.6	0.5	0.4		
0	5	6	7	8	0
	0.3	0.2	0.1		
0	9	-1	-2	-3	0
	-4	-5	-6	-7	
0	0	0	0	0	0

**V**

**R**


# 二维卷积

0	0	0	0	0	0
		0.9	0.8	0.7	
0	1	2	3	4	0
		0.6	0.5	0.4	
0	5	6	7	8	0
		0.3	0.2	0.1	
0	9	-1	-2	-3	0
0	-4	-5	-6	-7	0
0	0	0	0	0	0

**V**

**R**


# 二维卷积

0	0	0	0	0	0
			0.9	0.8	0.7
0	1	2	3	4	0
			0.6	0.5	0.4
0	5	6	7	8	0
			0.3	0.2	0.1
0	9	-1	-2	-3	0
0	-4	-5	-6	-7	0
0	0	0	0	0	0

V

R


# 二维卷积

0	0	0	0	0	0
0	1	2	3	4	0
0.9	0.8	0.7			0
0.6	0.5	0.4	7	8	0
0.3	0.2	0.1	-2	-3	0
0	-4	-5	-6	-7	0
0	0	0	0	0	0

**V**

**R**


# 二维卷积

0	0	0	0	0	0
0	1	2	3	4	0
0	5	6	7	8	0
0	9	-1	-2	-3	0
0	-4	-5	-6	-7	0
0	0	0	0	0	0

**V**

**R**


# 二维卷积

0	0	0	0	0	0
0	1	2	3	4	0
0	5	6	7	8	0
0	9	-1	-2	-3	0
0	-4	-5	-6	-7	0
0	0	0	0	0	0

**V**

**R**


# 二维卷积

0	0	0	0	0	0
0	1	2	3	4	0
0	5	6	7	8	0
0	9	-1	-2	-3	0
0	-4	-5	-6	-7	0
0	0	0	0	0	0

**V**

**R**




# 二维卷积

0	0	0	0	0	0
0	1	2	3	4	0
0	5	6	7	8	0
0	9	-1	-2	-3	0
0	-4	-5	-6	-7	0
0	0	0	0	0	0

**V**

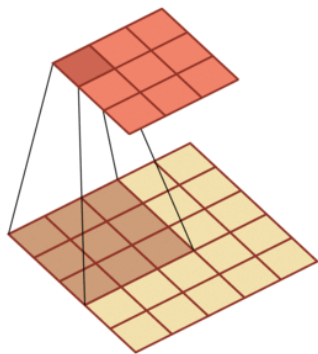
**R**


# 二维卷积

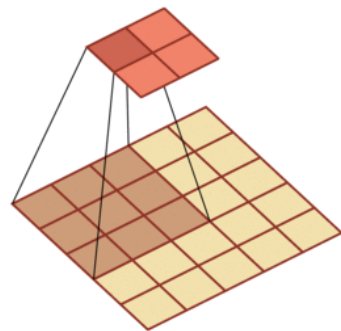
- 二维卷积有不同的类型
- 取决于步长和补零的数量
- 输出大小相应变化

# 二维卷积

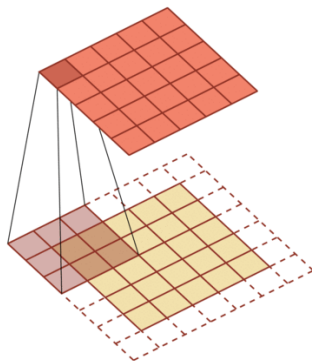
本页图片取自邱锡鹏  
《神经网络与深度学习》



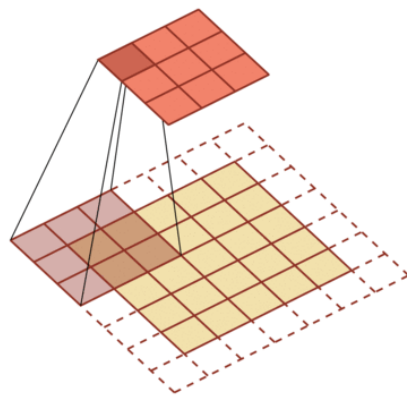
步长1, 零填充0



步长2, 零填充0



步长1, 零填充1



步长2, 零填充1



# 卷积有什么用？

# 卷积的应用

- 与深度学习直接相关
  - 提取图片特征
- 与概率统计直接相关
  - 滑动平均
  - 核密度估计
  - 随机变量之和的分布
- 其他
  - 信号处理
  - 整数乘法、多项式乘法

# 卷积与 图像处理

- 卷积是提取图形特征的一种工具
- 参见 [lec6-conv2d.ipynb](#)



# 卷积神经网络

# 回顾

- 前馈神经网络
- 实现简单
- 通用近似
- 计算高效
- 为什么还不够好？



# 前馈神经网络

- 数据维度高时参数很多
- 对于图片的缩放、平移、旋转等操作较敏感
- 难以提取图片的局部信息

# 卷积神经网络

- Convolutional neural network
- CNN/ConvNet

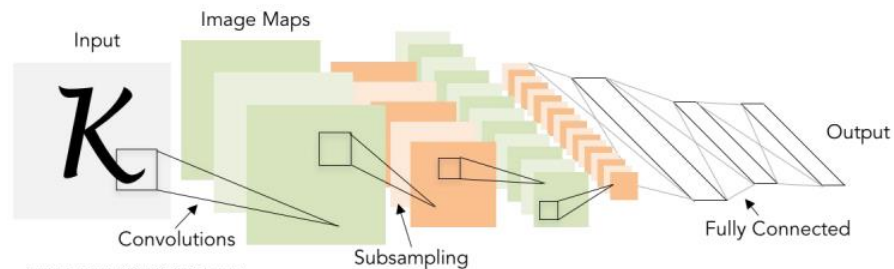
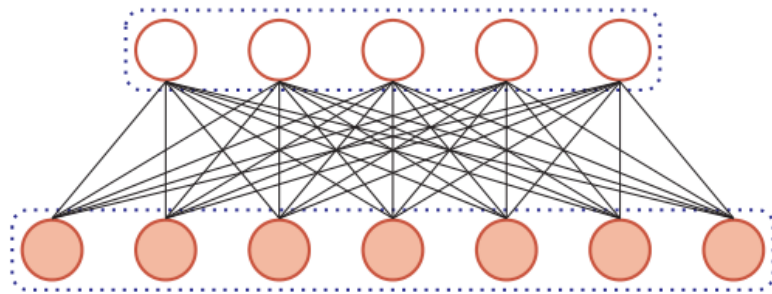


Illustration of LeCun et al. 1998 from CS231n 2017 Lecture 1

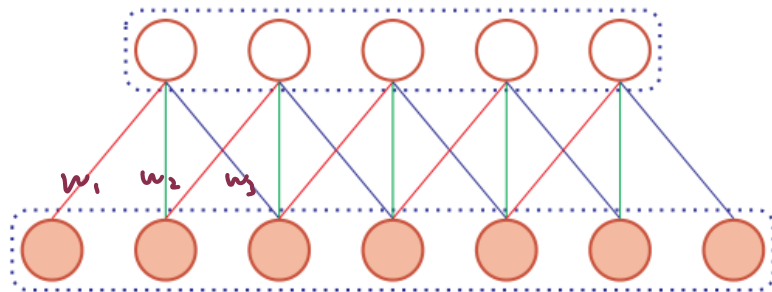
# 卷积神经网络

本页图片取自邱锡鹏  
《神经网络与深度学习》

- 简而言之，CNN 是利用了卷积操作的神经网络结构
- 用卷积层替换全连接层



(a) 全连接层



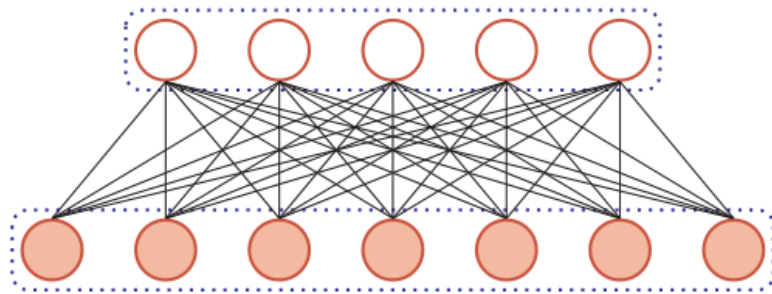
3个参数

(b) 卷积层

# 卷积层

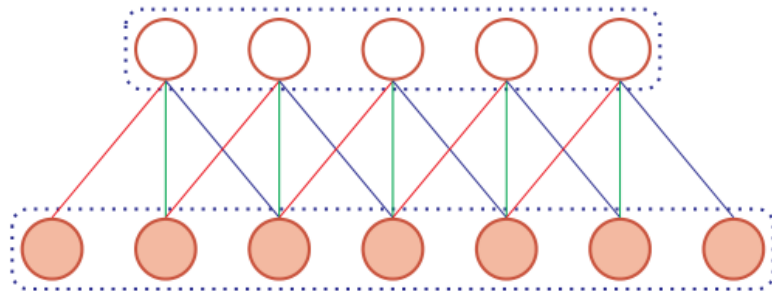
本页图片取自邱锡鹏  
《神经网络与深度学习》

- 局部连接
- 权重共享



(a) 全连接层

? 个参数



(b) 卷积层

? 个参数

# 卷积层

选定的矩阵

- 卷积核作为待估参数 (可学习参数)
- 输入: 图片数据  $[N \times H_{in} \times W_{in} \times C_{in}]$
- 输出: 特征映射 (feature map)  
 $[N \times H_{out} \times W_{out} \times C_{out}]$
- 某一层输出的特征映射 (经过激活函数作用后) 可以作为下一层的输入

## 注意事项

- 对于原始图片,  $C_{in}$  通常有明确的实际意义
- 如灰度图片  $C_{in} = 1$  (单个通道)
- 彩色图片有红绿蓝 (RGB) 三个通道,  $C_{in} = 3$
- 经过处理后的特征映射  $C_{out}$  可以任意指定, 只具有抽象的意义

# 注意事项

- 不同的软件框架可能采用不同的数据格式
- Tensorflow 采用  $[N \times H \times W \times C]$
- PyTorch 采用  $[N \times C \times H \times W]$
- MXNet 可以设定参数在两种格式中选择

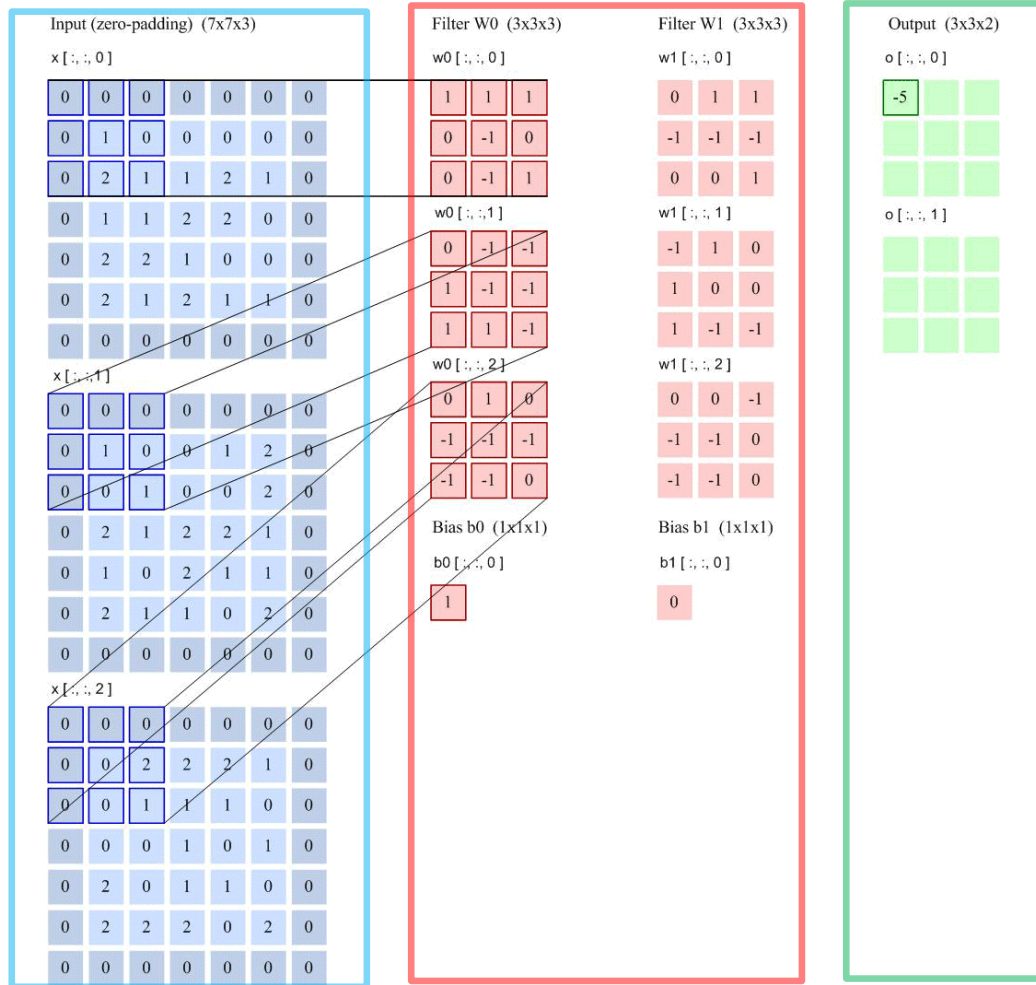
## 注意事项

- 将  $[N \times H_{in} \times W_{in} \times C_{in}]$  的输入变换为  $[N \times H_{out} \times W_{out} \times C_{out}]$  的输出，共需要  $C_{in} \times C_{out}$  个卷积核
- 卷积核的类型和大小决定了  $H_{in} \times W_{in}$  和  $H_{out} \times W_{out}$  之间的对应关系



# 一般形式

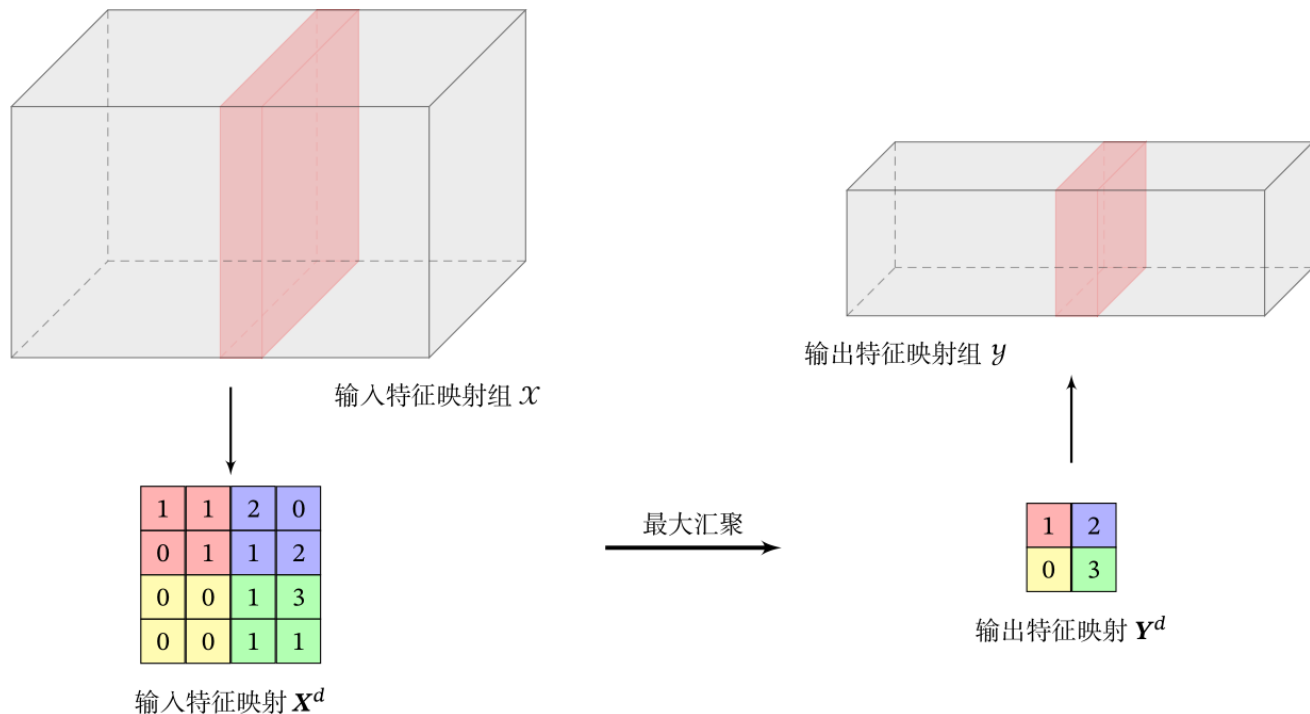
本页图片取自邱锡鹏  
《神经网络与深度学习》



# 汇聚层

本页图片取自邱锡鹏  
《神经网络与深度学习》

- 早期的 CNN 中通常还会加入一些汇聚层 (pooling layer, 或称池化层)
- 用来快速减少神经元的数目



# 汇聚层

- 当前，汇聚层有被步长 $>1$ 的卷积层替代的趋势

# LeNet-5

- LeNet-5 是一个早期的 CNN 架构
- 结合了卷积层、汇聚层和全连接层
- 结合手写识别数据MNIST，成为了经典的 CNN 入门模型

