

AI深度学习之自然语言处理顶级实战课程

七、深度学习之卷积神经网络

讲师：aopu

自我介绍

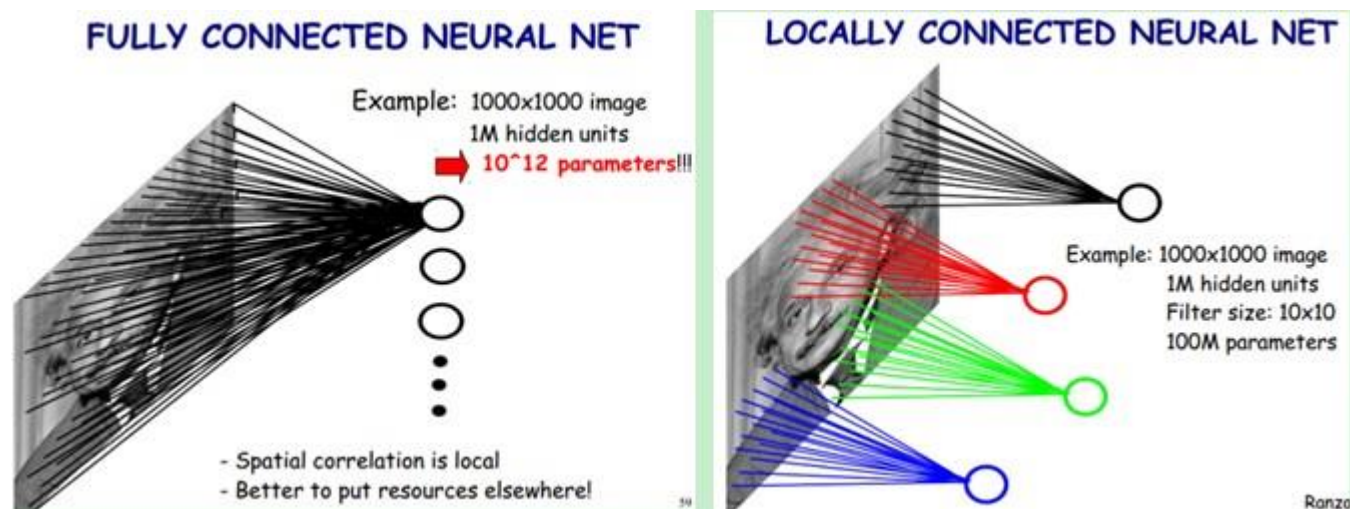
- 天善商业智能和大数据社区[aopu](#) 讲师
- 天善社区ID- [aopu](#)主页
- <https://www.hellobi.com> – 学习过程中有任何相关的问题都可以提到技术社区 人工智能 版块。

7、2 彻底理解深度学习之卷积神经网络

卷积神经网络(**Convolutional Neural Network, CNN**)是深度学习技术中极具代表的网络结构之一。CNN相较于传统的算法避免了对图像复杂的前期预处理过程(提取人工特征等),可以直接输入原始图像。

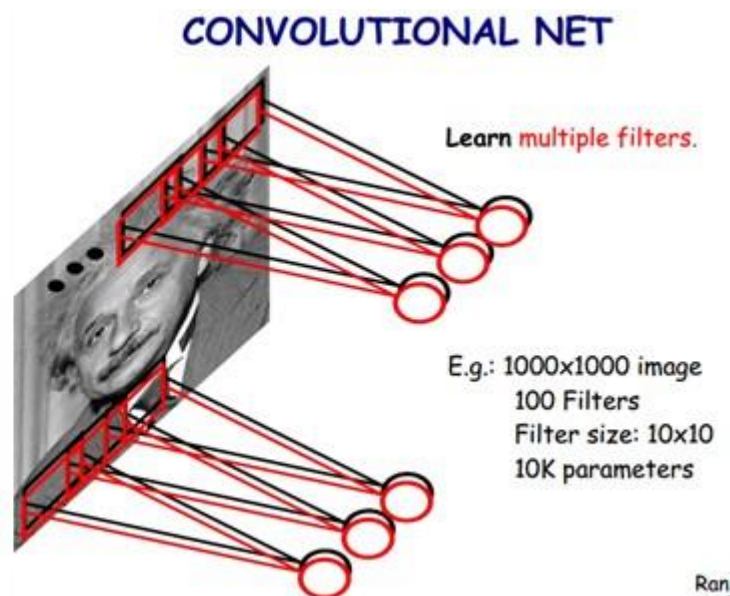
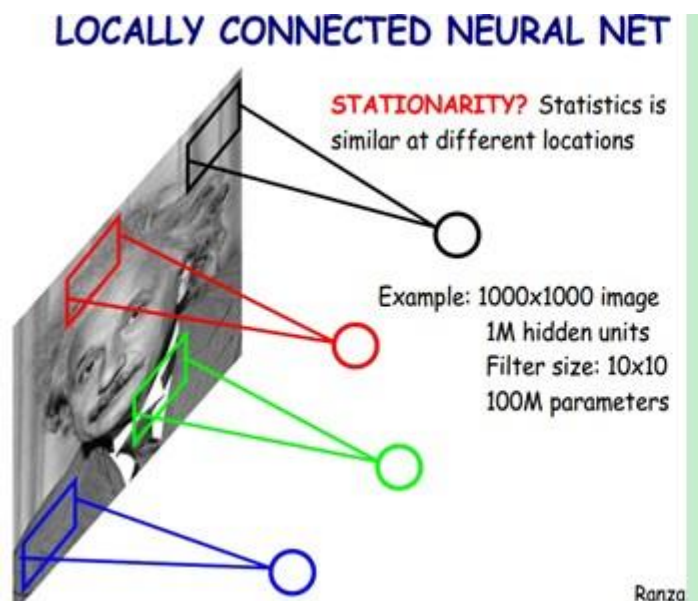
CNN中的**局部连接(Sparse Connectivity)**和**权值共享(Shared Weights)**方法,下图是一个很经典的图示,左边是全连接,右边是局部连接。

对于一个 1000×1000 的输入图像而言,如果下一个隐藏层的神经元数目为 10^6 个,采用全连接则有 $1000 \times 1000 \times 10^6 = 10^{12}$ 个权值参数,如此数目巨大的参数几乎难以训练;而采用局部连接,隐藏层的每个神经元仅与图像中 10×10 的局部图像相连接,那么此时的权值参数数量为 $10 \times 10 \times 10^6 = 10^8$,将直接减少4个数量级。



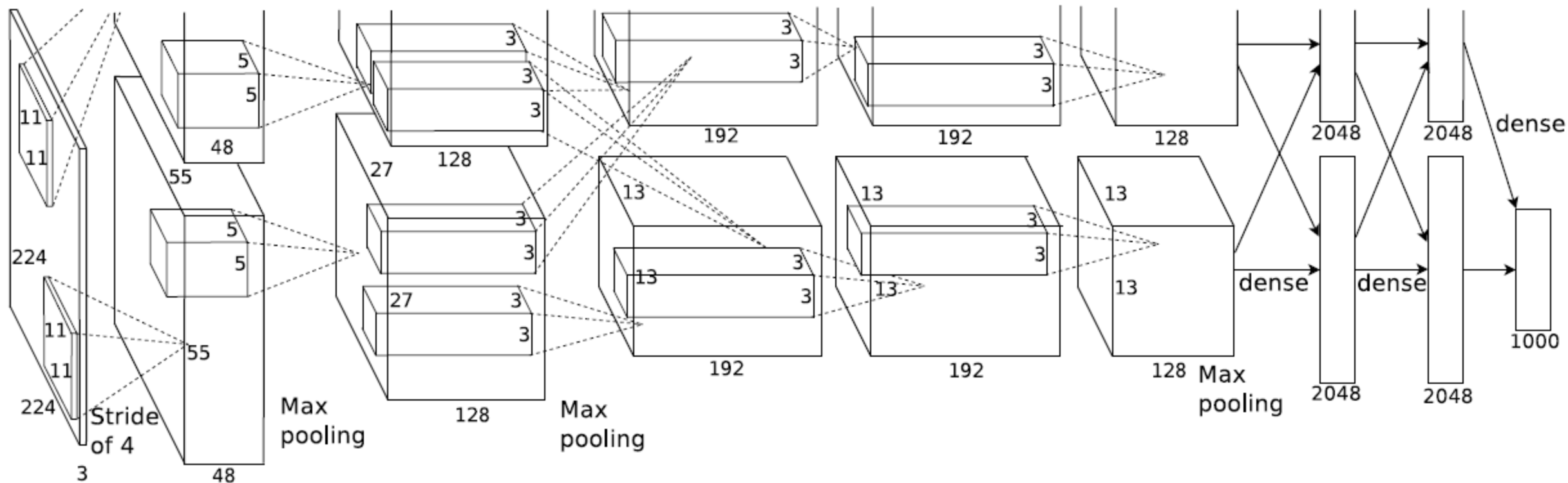
7、2 彻底理解深度学习之卷积神经网络

尽管减少了几数量级，但参数数量依然较多。请看**权值共享继续减少参数**。具体做法是，在局部连接中隐藏层的每一个神经元连接的是一个 10×10 的局部图像，因此有 10×10 个权值参数，将这 10×10 个权值参数共享给剩下的神经元，也就是说隐藏层中 10^6 个神经元的权值参数相同，那么此时不管隐藏层神经元的数目是多少，需要训练的参数就是这 10×10 个权值参数（也就是卷积核(也称滤波器)的大小），如下图。



这大概就是CNN的一个神奇之处，尽管只有这么少的参数，依旧有出色的性能。但是，这样仅提取了图像的一种特征，如果要多提取出一些特征，可以增加多个卷积核，不同的卷积核能够得到图像的不同映射下的特征，称之为**Feature Map**。如果有100个卷积核，最终的权值参数也仅为 $100 \times 100 = 10^4$ 个而已。另外，偏置参数也是共享的，同一种滤波器共享一个。

7、2 彻底理解深度学习之卷积神经网络



卷积神经网络（CNN）由输入层、卷积层、激活函数、池化层、全连接层组成，即INPUT-CONV-RELU-POOL-FC

7、2 彻底理解深度学习之卷积神经网络

卷积

卷积层是卷积核在上一级输入层上通过逐一滑动窗口计算而得，卷积核中的每一个参数都相当于传统神经网络中的权值参数，与对应的局部像素相连接，将卷积核的各个参数与对应的局部像素值相乘之和，（通常还要再加上一个偏置参数），得到卷积层上的结果。如下图所示。

1 _{x1}	1 _{x0}	1 _{x1}	0	0
0 _{x0}	1 _{x1}	1 _{x0}	1	0
0 _{x1}	0 _{x0}	1 _{x1}	1	1
0	0	1	1	0
0	1	1	0	0

Image

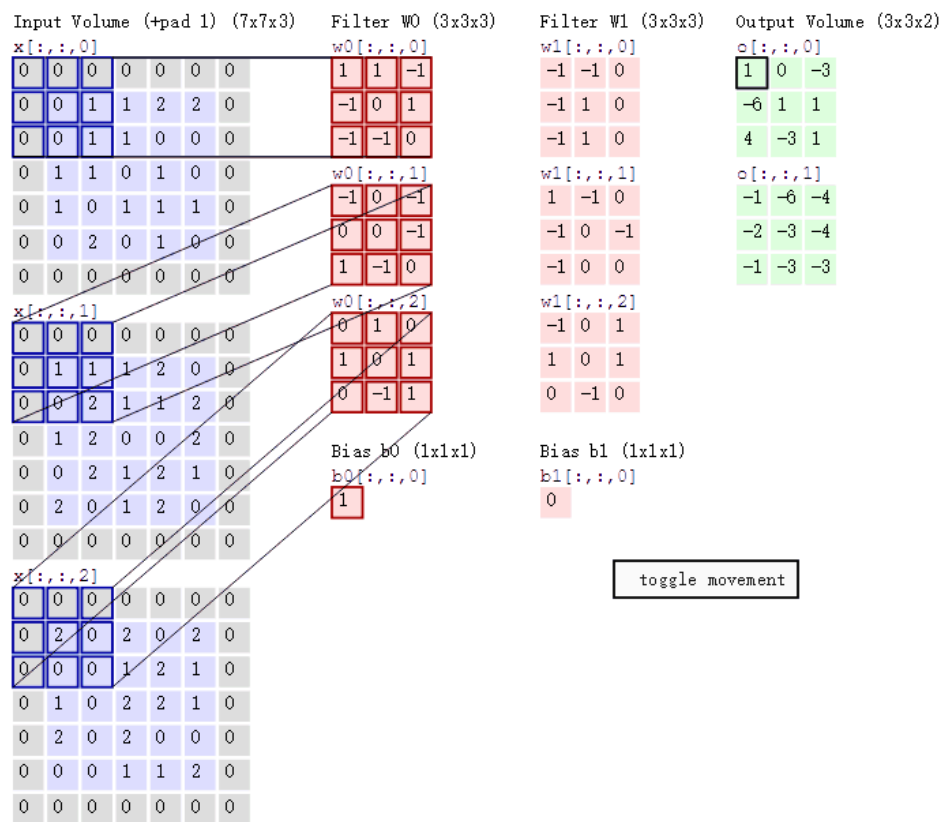
4		

Convolved
Feature

7、2 彻底理解深度学习之卷积神经网络

高维度卷积层

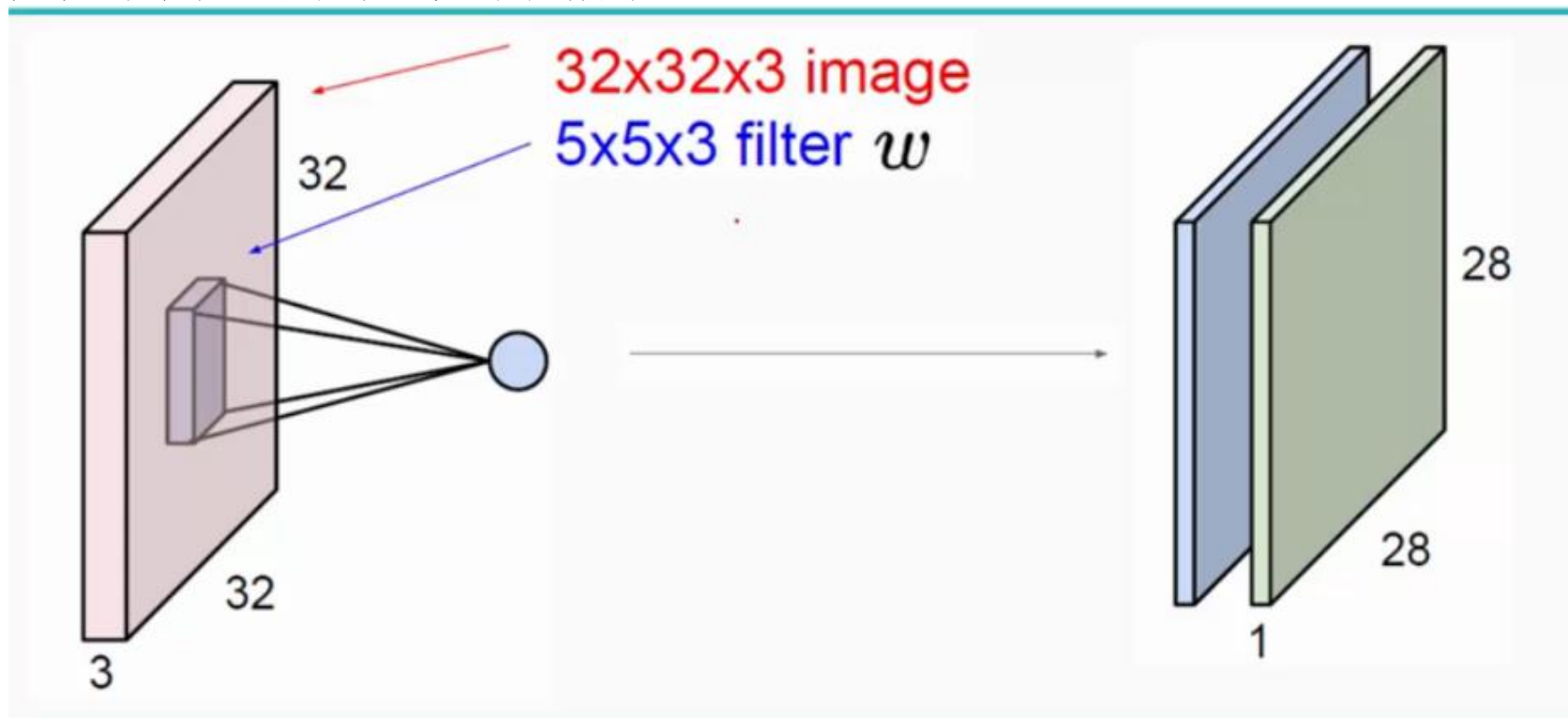
输入图像和卷积核的对应位置元素相乘再求和，最后再加上b,得到特征图。 w_0 的第一层深度和输入图像的蓝色方框中对应元素相乘再求和得到0，其他两个深度得到2，0，则有 $0+2+0+1=3$ 即图中右边特征图的第一个元素3.，卷积过后输入图像的蓝色方框再滑动， $\text{stride}=2$ ，



7、2 彻底理解深度学习之卷积神经网络

高维度卷积层

卷积层是卷积核在上一级输入层上通过逐一滑动窗口计算而得，卷积核中的每一个参数都相当于传统神经网络中的权值参数，与对应的局部像素相连接，将卷积核的各个参数与对应的局部像素值相乘之和，（通常还要再加上一个偏置参数），得到卷积层上的结果。如下图所示。



输入图像是 $32 \times 32 \times 3$ ，3是它的深度（即R、G、B），卷积层是一个 $5 \times 5 \times 3$ 的filter(感受野)，这里注意：感受野的深度必须和输入图像的深度相同。通过一个filter与输入图像的卷积可以得到一个 $28 \times 28 \times 1$ 的特征图，上图是用了两个filter得到了两个特征图；

7、2 彻底理解深度学习之卷积神经网络

激活层

激活函数如下：

Sigmoid(S形函数)

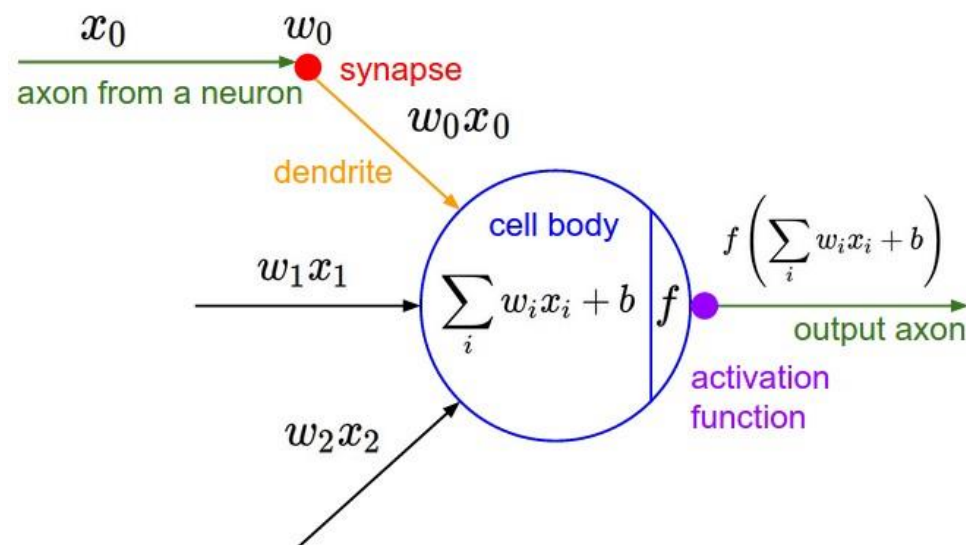
Tanh(双曲正切,双S形函数)

ReLU

Leaky ReLU

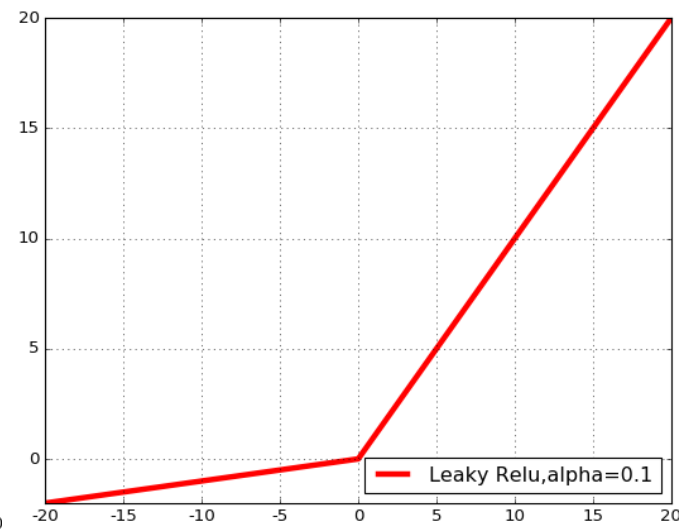
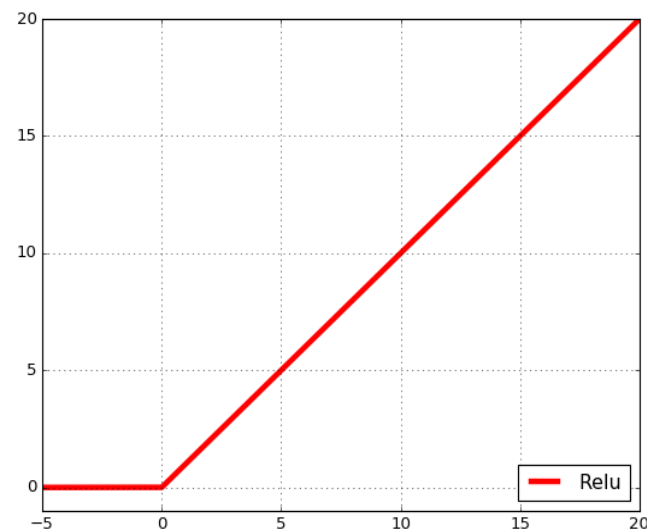
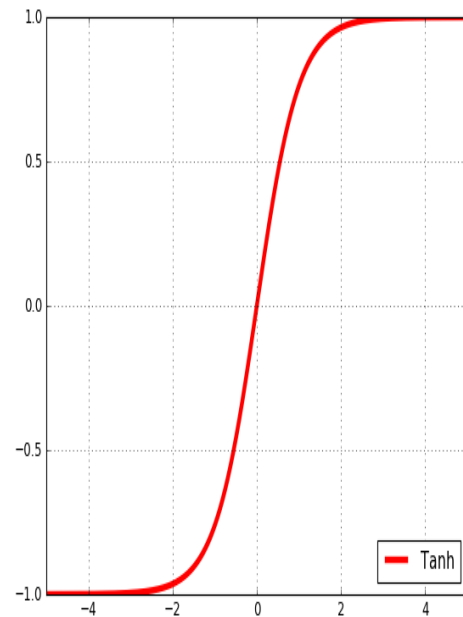
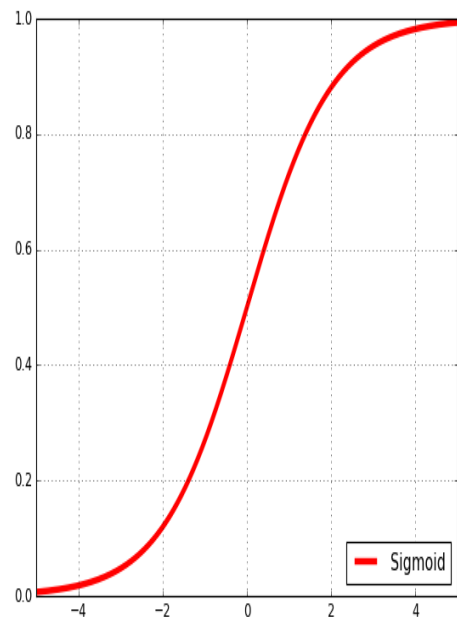
ELU

Maxout



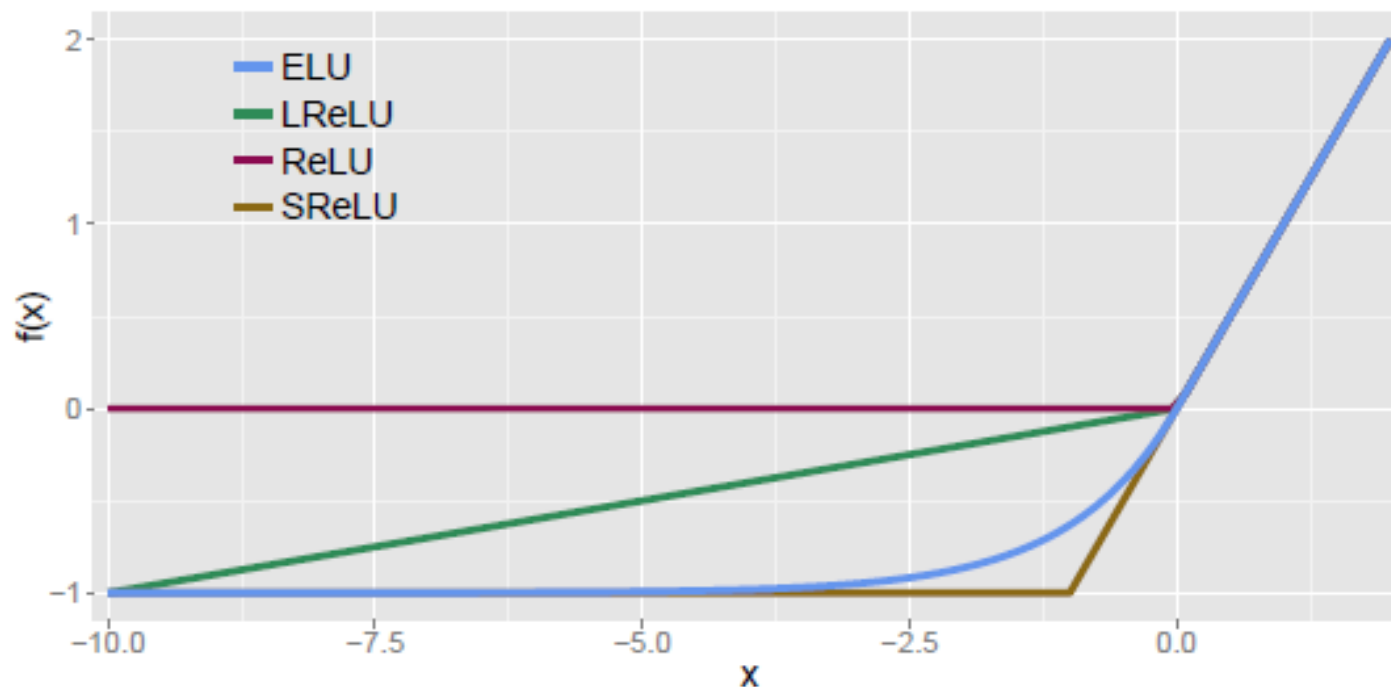
7、2 彻底理解深度学习之卷积神经网络

激活层



7、2 彻底理解深度学习之卷积神经网络

激活层



The *exponential linear unit* (ELU) with $0 < \alpha$ is

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ \alpha (\exp(x) - 1) & \text{if } x \leq 0 \end{cases}, \quad f'(x) = \begin{cases} 1 & \text{if } x > 0 \\ f(x) + \alpha & \text{if } x \leq 0 \end{cases}.$$

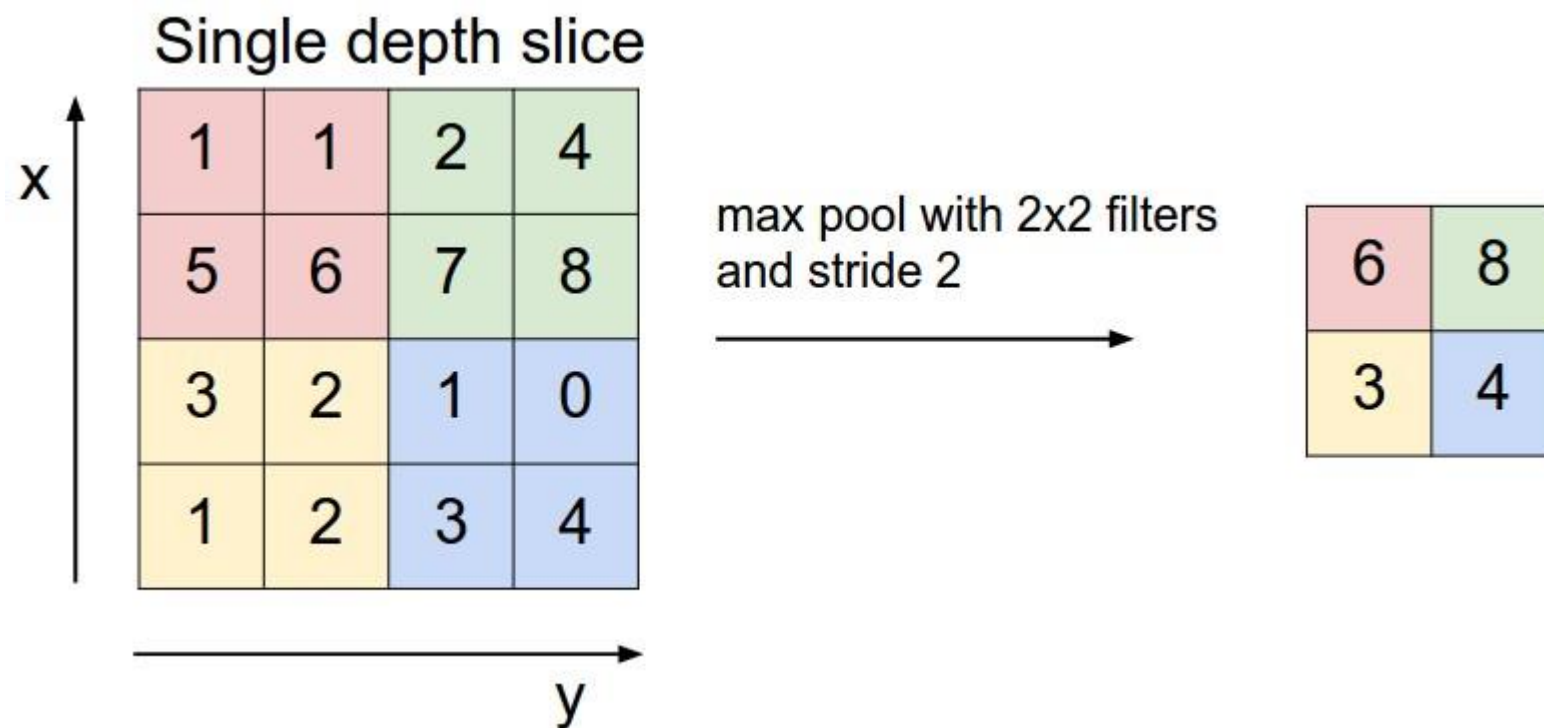
7、2 彻底理解深度学习之卷积神经网络

池化层

在池化层中，进行压缩减少特征数量的时候一般采用两种策略：

Max Pooling: 最大池化，一般采用该方式

Average Pooling: 平均池化



7、2 彻底理解深度学习之卷积神经网络

正则化和dropout

Regularization: 正则化，通过降低模型的复杂度，通过在cost函数上添加一个正则项的方式来降低overfitting，主要有L1和L2两种方式。

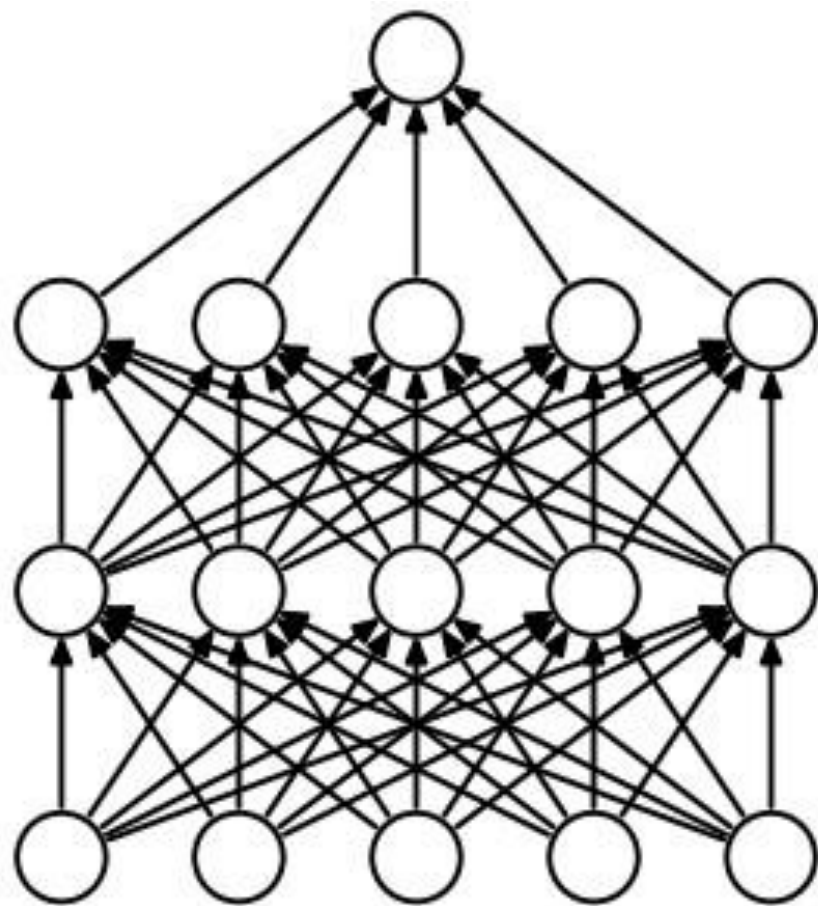
Dropout: 通过随机删除神经网络中的神经元来解决overfitting问题，在每次迭代的时候，只使用部分神经元训练模型获取W和b的值。

对于同一组训练数据，利用不同的神经网络训练之后，求其输出的平均值可以减少过拟合。Dropout就是利用这个原理，每次丢掉一半左右的隐藏层神经元，相当于在不同的神经网络上进行训练，这样就减少了神经元之间的依赖性，即每个神经元不能依赖于某几个其它的神经元（指层与层之间相连接的神经元），使神经网络更加能学习到与其它神经元之间的更加健壮robust（鲁棒性）的特征。另外Dropout不仅减少过拟合，还能提高准确率。

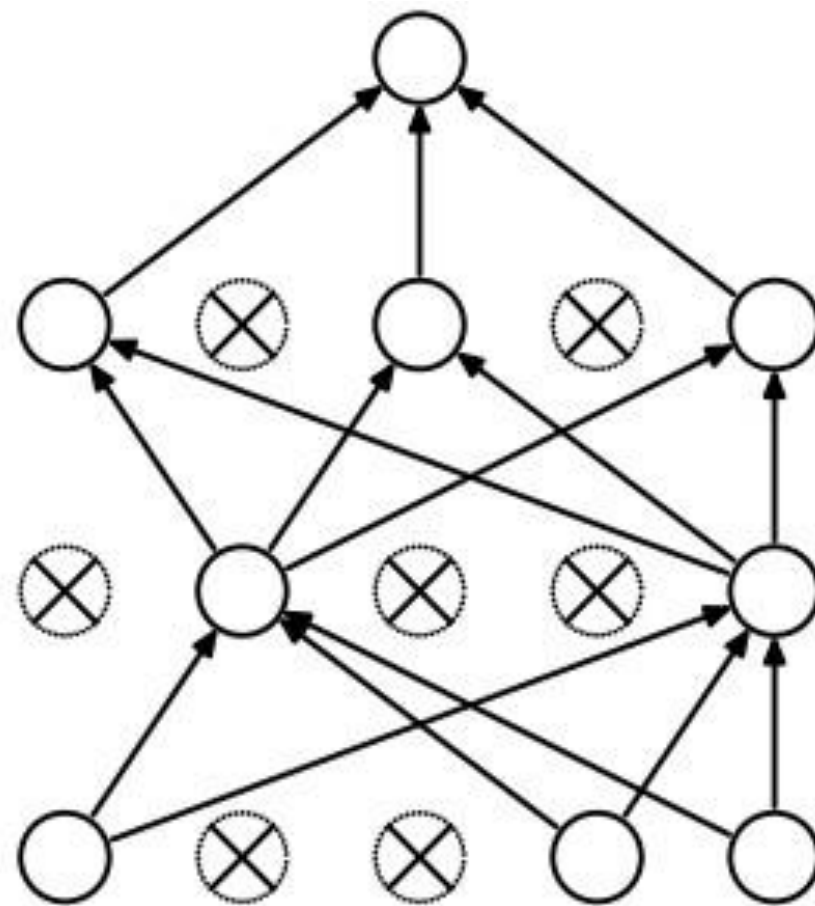
正则化是通过给cost函数添加正则项的方式来解决过拟合，Dropout是通过直接修改神经网络的结构来解决过拟合。

7、2 彻底理解深度学习之卷积神经网络

正则化和dropout



(a) Standard Neural Net

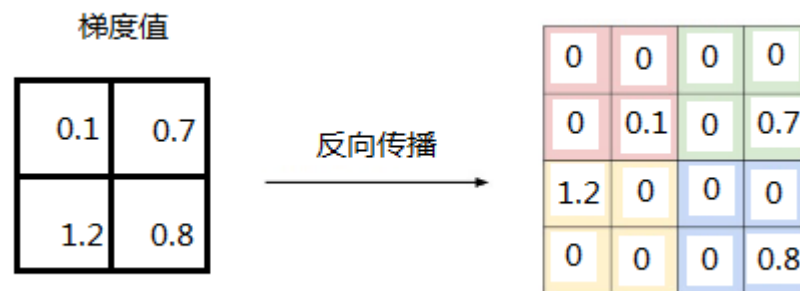
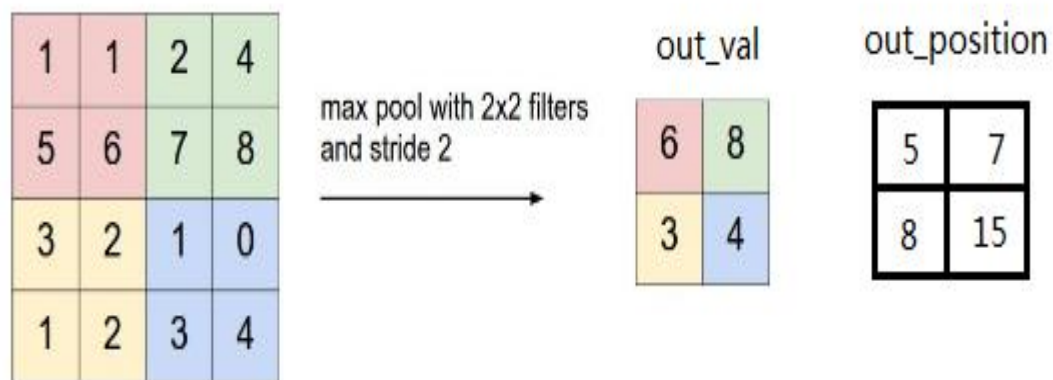


(b) After applying dropout.

7、2 彻底理解深度学习之卷积神经网络

池化层误差反向传播

Maxpool 池化层反向传播，除最大值处继承上层梯度外，其他位置置零。

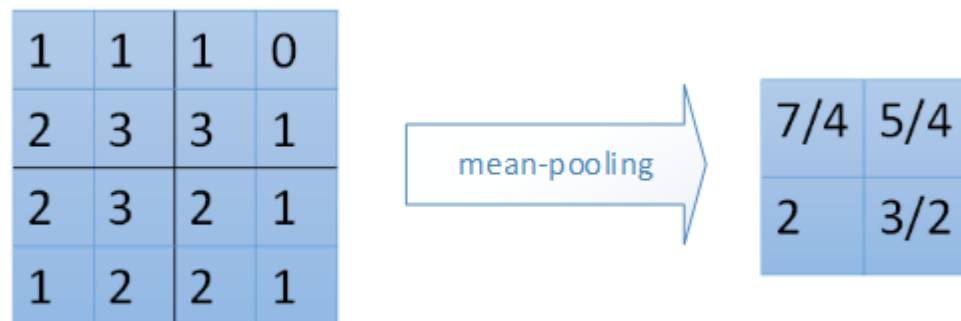


7、2 彻底理解深度学习之卷积神经网络

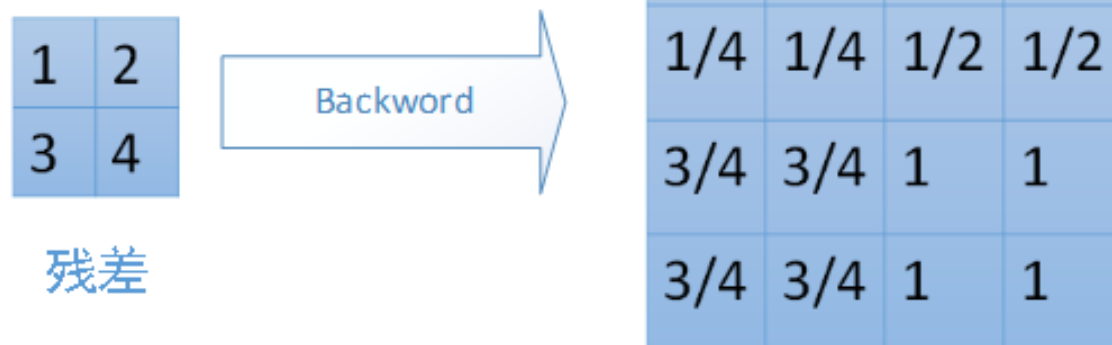
池化层误差反向传播

平均池化，我们需要把残差平均分成 $2*2=4$ 份，传递到前边小区域的4个单元即可。

误差反向传播前



误差反向传播后



残差

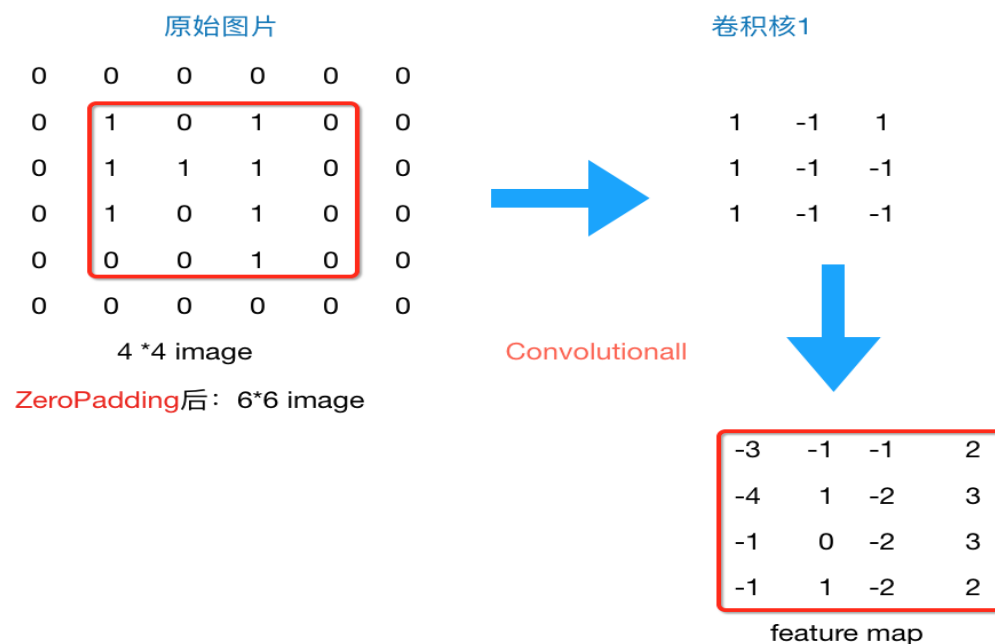
7、2 彻底理解深度学习之卷积神经网络

Zero Padding

所以到现在为止，我们的图片由4*4，通过卷积层变为3*3，再通过池化层变化2*2，如果我们再添加层，那么图片岂不是会越变越小？这个时候我们就会引出“Zero Padding”（补零），它可以帮助我们保证每次经过卷积或池化输出后图片的大小不变，如，上述例子我们如果加入Zero Padding，再采用3*3的卷积核，那么变换后的图片尺寸与原图片尺寸相同。图片做完卷积操作后保持图片大小不变，所以我们一般会选择尺寸为3*3的卷积核和1的zero padding，或者5*5的卷积核与2的zero padding，这样通过计算后，可以保留图片的原始尺寸。那么加入zero padding后的

$\text{feature_map尺寸} = (\text{width} + 2 * \text{padding_size} - \text{filter_size}) / \text{stride} + 1$

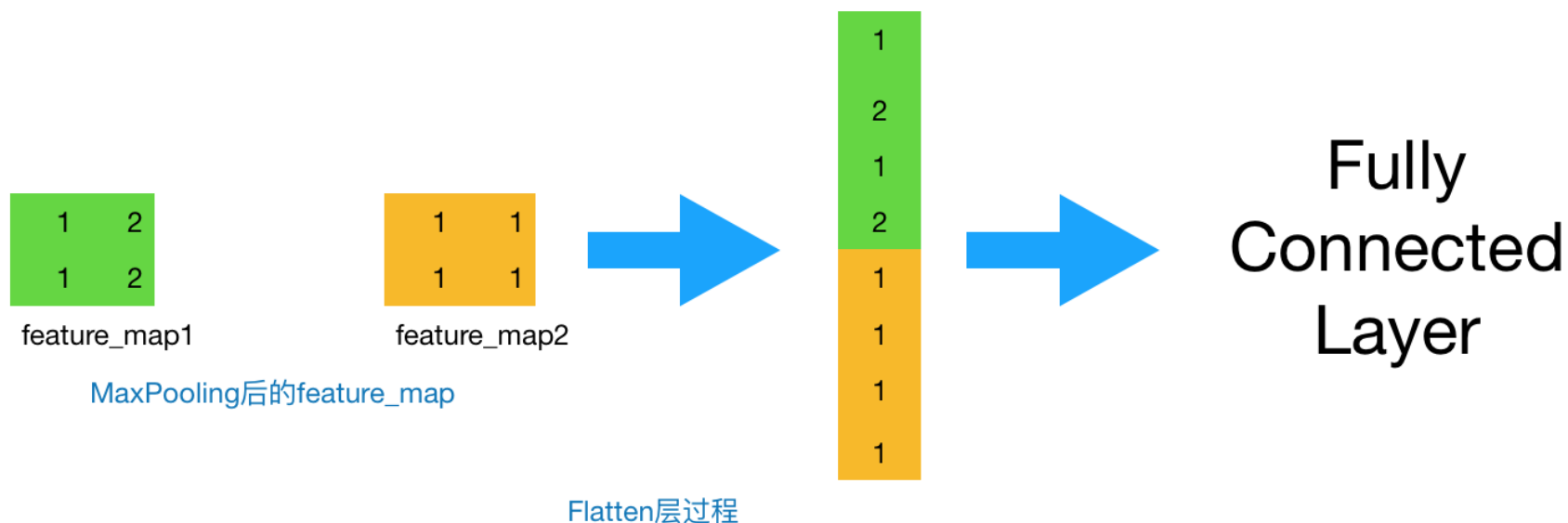
注：这里的width也可换成height，此处是默认正方形的卷积核， $\text{weight} = \text{height}$ ，如果两者不相等，可以分开计算，分别补零。



7、2 彻底理解深度学习之卷积神经网络

Flatten层 & Fully Connected Layer

到这一步，其实我们的一个完整的“卷积部分”就算完成了，如果想要叠加层数，一般也是叠加“Conv-MaxPooling”，通过不断的设计卷积核的尺寸，数量，提取更多的特征，最后识别不同类别的物体。做完Max Pooling后，我们会把这些数据“拍平”，丢到Flatten层，然后把Flatten层的output放到full connected Layer里，采用softmax对其进行分类。



秦路主讲

七周成为数据分析师

七周为期，Get一条数据分析师职业黄金通道！



Python

数据分析与挖掘

集Python爬虫、数据采集、数据处理、数据分析与数据挖掘于一体，打造Python全栈工程师

主讲老师: 韦玮

VIP会员群+在线答疑+录播复习+1年反复观看

案例为师,实战为王

开启Python机器学习之路

科学规划全套课程体系，从入门到进阶，从理论到技巧，嵌入丰富课程案例讲解，逐步推进

讲师: 唐宇迪 深度学习领域多年一线实践研究专家

独一无二的 数据仓库建模指南系列教程升级版

- 从企业视角进行数据规划以及数据仓库模型的搭建
- 高质量的数据库模型和技巧，以及丰富的例子
- 数据仓库架构理论和实践要领

资深讲师: BAO胖子 15年+BI从业经验
涉足电力、快消品、医药、信息服务行业的BI老兵

业务知识一站通

技术+业务，挣钱有门路！

讲师: 陈文



自己动手 丰衣足食

Python3网络爬虫实战案例

一循序渐进，案例为王，诠释全面，思路制胜一

讲师: 崔庆才 北航硕士，百万级热度爬文博主



讲师 丘祐玮

人人都爱数据科学家

Python数据科学精华实战课程

数据分析报告制作

秘籍升级版

讲师: 陈丹奕 知乎大神，前百度资深数据分析师

先机致胜 破冰AI

深度学习模型/框架与实战

讲师: 唐宇迪 同济大学硕士
深度学习领域多年一线实践研究专家



BI、商业智能
数据挖掘 大数据
数据分析师
R语言 Python
机器学习
深度学习
人工智能
Hive Hadoop
Tableau
BIEE ETL
数据科学家
PowerBI