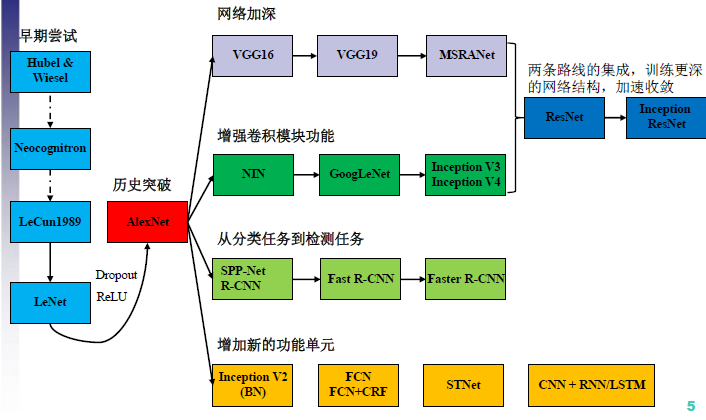
# 一 inception系列演进

## 1.1 inceptionv1

Inceptionv1的核心点

Inceptionv1的创新点

inceptionv1, v2, v3, v4的演进.

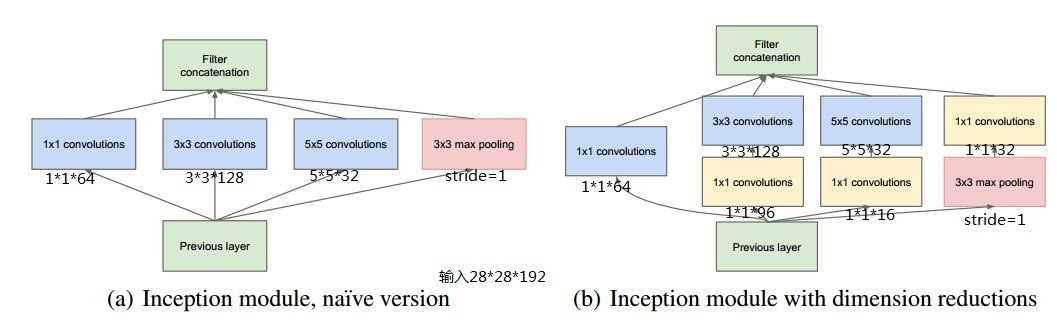


### 1.1.1 inception V1 特点

a) 从NIN中引入1x1卷积.

b) 代表作, googleLeNet.

c) 1x1的卷积,相当于对所有通道(chn)的像素线性组合输出.类似全连接.



假设previous layer的大小为28\*28\*192，则，

a的weights大小，1\*1\*192\*64+3\*3\*192\*128+5\*5\*192\*32=387072

a的输出featuremap大小，28\*28\*64+28\*28\*128+28\*28\*32+28\*28\*192=28\*28\*416

b的weights大小，1\*1\*192\*64+(1\*1\*192\*96+3\*3\*96\*128)+(1\*1\*192\*16+5\*5\*16\*32)+1\*1\*192\*32=163328

b的输出feature map大小，28\*28\*64+28\*28\*128+28\*28\*32+28\*28\*32=28\*28\*256

1\*1 conv，从上面的数据可以**看出一方面减少了weights(?)，另一方面降低了dimension**。

Inception v1的亮点总结如下：

(1)卷积层共有的一个功能，可以实现通道方向的降维和增维，至于是降还是增，取决于卷积层的通道数（滤波器个数），在Inception v1中1\*1卷积用于降维，减少weights大小和feature map维度。◇◇◇

(2)1\*1卷积特有的功能，由于1\*1卷积只有一个参数，相当于对原始feature map做了一个scale，并且这个scale还是训练学出来的，无疑会对识别精度有提升。◇◇

(3)增加了网络的深度

(4)增加了网络的宽度

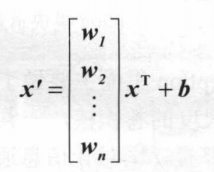
(5)同时使用了1\*1，3\*3，5\*5的卷积，增加了网络对尺度的适应性

### 1.1.2 1x1卷积用处

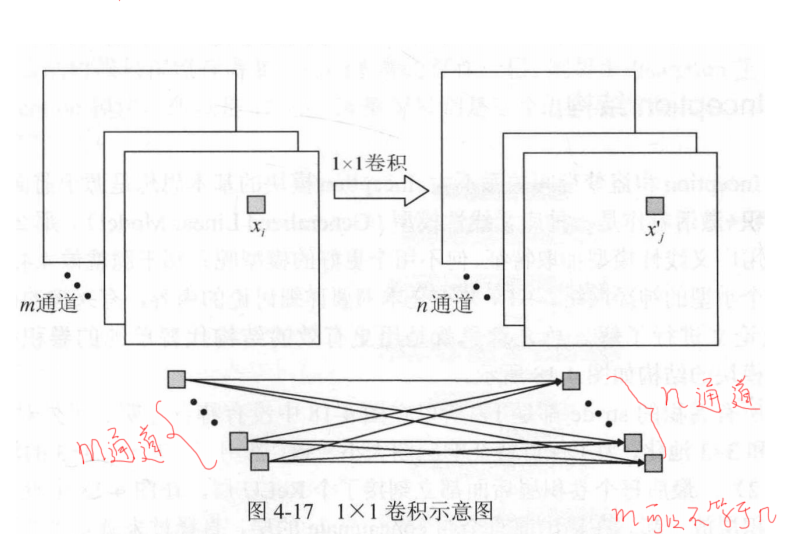
1. (直观上)给特征响应图都乘以一个系数. 这个只是单通道是每个Fmap值都乘以一个系数.但是多通道就不是了.
2. M通道上同一位置的像素, 经1x1卷积后, 会在对应位置产生一组新的n个通道像素. ◇
3. 如果只看”同一位置”像素的话, 就是一个全连接层. ◇◇

Xi是第i个输入通道上的像素值, Xj’是第j个输出通道上的像素值.





1. 类似同一位置上像素做的一个线性组合. ◇◇
2. 如果m>n, 则会有降维效果.(通道数降低) ◇◇◇



## 1.2 Inception V2

Inception v2的网络，代表作为加入了BN（Batch Normalization）层，并且使用2个3\*3替代1个5\*5卷积的改进版GoogleNet。

Inception v2的亮点总结如下：

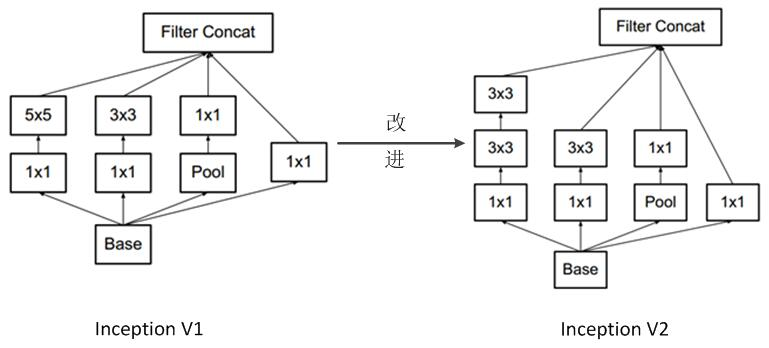
(1)加入了BN层，减少了InternalCovariate Shift（内部neuron的数据分布发生变化），使每一层的输出都规范化到一个N(0, 1)的高斯，从而增加了模型的鲁棒性，可以以更大的学习速率训练，收敛更快，初始化操作更加随意，同时作为一种正则化技术，可以减少dropout层的使用。

(2)用2个连续的3\*3 conv替代inception模块中的5\*5，从而实现网络深度的增加，

**网络参数减少:**

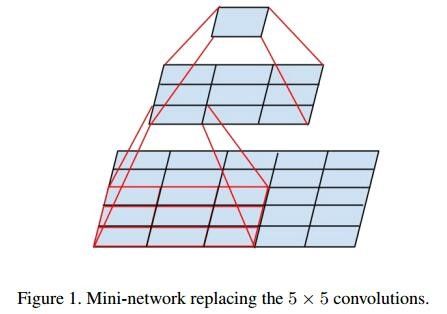
1. 5x5xchn个参数.
2. 2x(3x3xchn) 个参数

减少了(1 – 18/25) = 28%



### 5x5替换成2个级联3x3在参数量和计算量上的影响

大尺寸的卷积核可以带来更大的感受野，但也意味着更多的参数，比如5x5卷积核参数(仅考虑w不考虑b)是3x3卷积核的（5\*5+1）/（3\*3+1）=2.6倍。为此，《Rethinking the Inception Architecture for Computer Vision》的作者提出可以用2个连续的3x3卷积层(stride=1)组成的小网络来代替单个的5x5卷积层，(保持感受野范围的同时又减少了参数量)如下图：



#### 参数对比

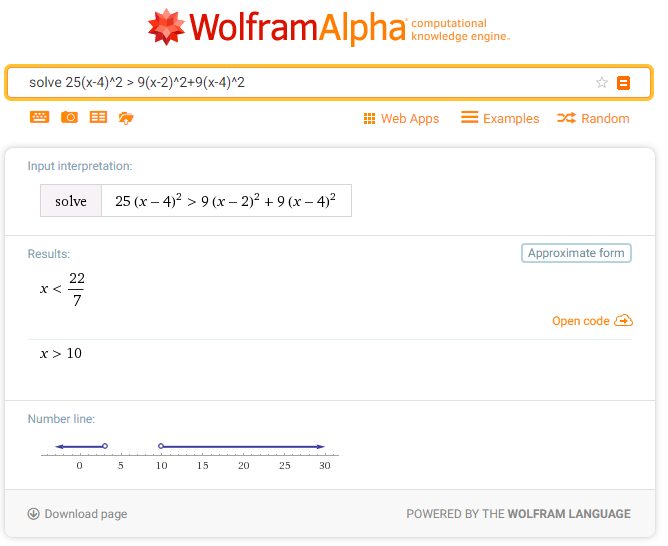
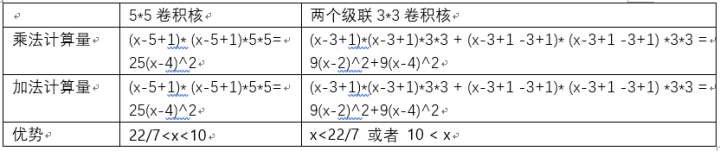


由于参数个数仅与卷积核大小有关，所以3\*3级联卷积核占优势。

#### 计算量对比

输入记为x，为了方便讨论假设padding=0,stride=1。此时卷积计算公式 output =( input – kernel + 2padding) / stride + 1简化为output = input – kernel + 1。

* 5\*5卷积：有(x-5+1)\* (x-5+1)个输出点，每个输出点对应5\*5次乘法和5\*5次加法（5\*5次乘法的结果求和再加上b，一共5\*5+1个数相加，所以需要5\*5次加法）
* 3\*3卷积：第一个3\*3卷积有(x-3+1)\*(x-3+1)个输出点，每个输出点对应3\*3次乘法和3\*3次加法，第二个3\*3卷积的输入是(x-3+1)\*(x-3+1)，在其上做卷积有(x-3+1 -3+1)\* (x-3+1 -3+1)个输出点，每个输出点对应3\*3次乘法和3\*3次加分。



总的来说当**x<22/7 或者10<x** ，两个3\*3的卷积核在参数个数和计算量上都占优势。