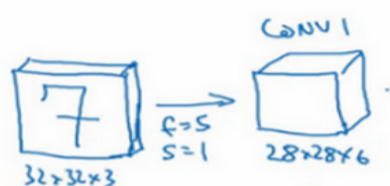


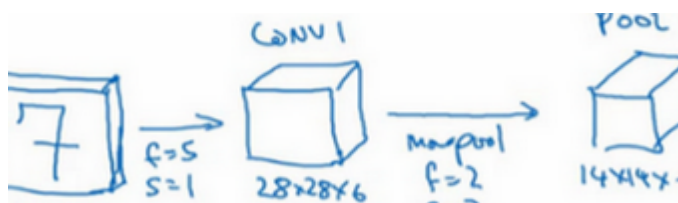
1.10 卷积神经网络示例（Convolutional neural network example）

构建全卷积神经网络的构造模块我们已经掌握得差不多了，下面来看个例子。

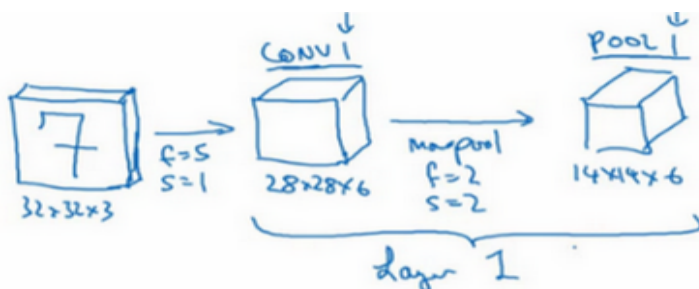
假设，有一张大小为 $32 \times 32 \times 3$ 的输入图片，这是一张 **RGB** 模式的图片，你想做手写体数字识别。 $32 \times 32 \times 3$ 的 **RGB** 图片中含有某个数字，比如 7，你想识别它是从 0-9 这 10 个数字中的哪一个，我们构建一个神经网络来实现这个功能。



我用的这个网络模型和[经典网络 LeNet-5](#) 非常相似，灵感也来源于此。**LeNet-5** 是多年前 **Yann LeCun** 创建的，我所采用的模型并不是 **LeNet-5**，但是受它启发，许多参数选择都与 **LeNet-5** 相似。输入是 $32 \times 32 \times 3$ 的矩阵，假设第一层使用过滤器大小为 5×5 ，步幅是 1，padding 是 0，过滤器个数为 6，那么输出为 $28 \times 28 \times 6$ 。将这层标记为 **CONV1**，它用了 6 个过滤器，增加了偏差，应用了非线性函数，可能是 **ReLU** 非线性函数，最后输出 **CONV1** 的结果。

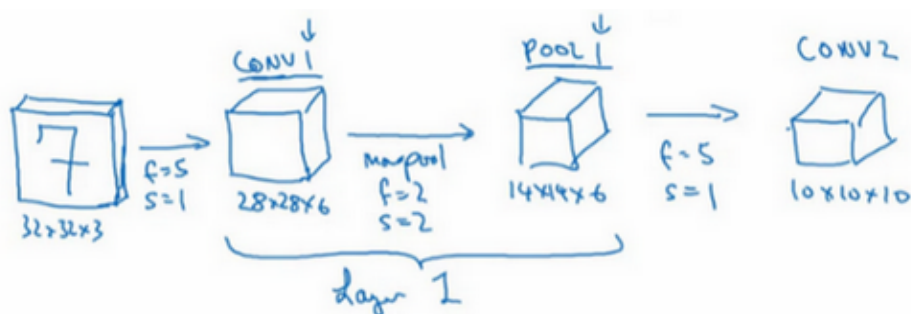


然后构建一个池化层，这里我选择用最大池化，参数 $f = 2$ ， $s = 2$ ，因为 padding 为 0，我就不写出来了。现在开始构建池化层，最大池化使用的过滤器为 2×2 ，步幅为 2，表示层的高度和宽度会减少一半。因此， 28×28 变成了 14×14 ，通道数量保持不变，所以最终输出为 $14 \times 14 \times 6$ ，将该输出标记为 **POOL1**。

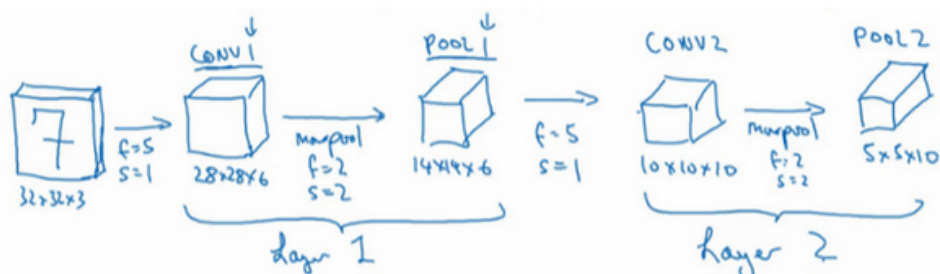


人们发现在卷积神经网络文献中，卷积有两种分类，这与所谓层的划分存在一致性。一

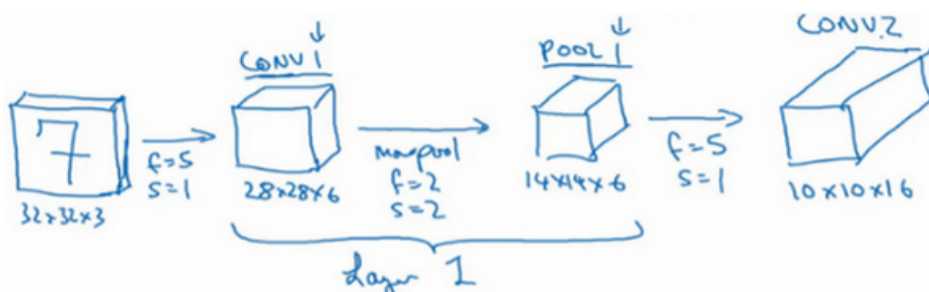
类卷积是一个卷积层和一个池化层一起作为一层，这就是神经网络的 **Layer1**。另一类卷积是把卷积层作为一层，而池化层单独作为一层。人们在计算神经网络有多少层时，通常只统计具有权重和参数的层。因为池化层没有权重和参数，只有一些超参数。这里，我们把 **CONV1** 和 **POOL1** 共同作为一个卷积，并标记为 **Layer1**。虽然你在阅读网络文章或研究报告时，你可能会看到卷积层和池化层各为一层的情况，这只是两种不同的标记术语。一般我在统计网络层数时，只计算具有权重的层，也就是把 **CONV1** 和 **POOL1** 作为 **Layer1**。这里我们用 **CONV1** 和 **POOL1** 来标记，两者都是神经网络 **Layer1** 的一部分，**POOL1** 也被划分在 **Layer1** 中，因为它没有权重，得到的输出是 $14 \times 14 \times 6$ 。



我们再为它构建一个卷积层，过滤器大小为 5×5 ，步幅为 1，这次我们用 10 个过滤器，最后输出一个 $10 \times 10 \times 10$ 的矩阵，标记为 **CONV2**。

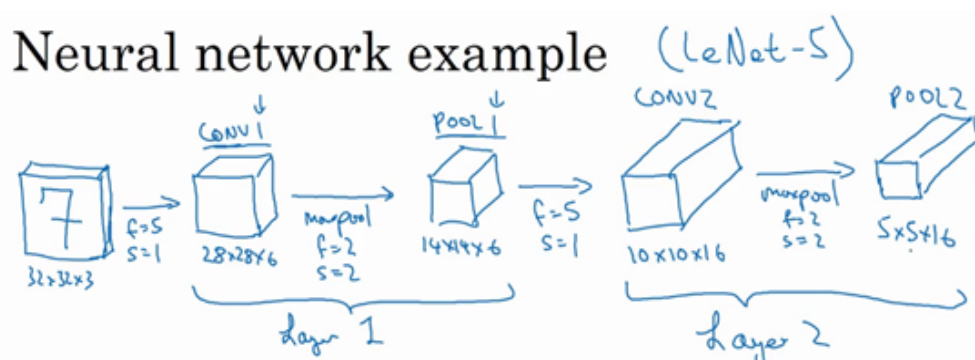


然后做最大池化，超参数 $f = 2$ ， $s = 2$ 。你大概可以猜出结果， $f = 2$ ， $s = 2$ ，高度和宽度会减半，最后输出为 $5 \times 5 \times 10$ ，标记为 **POOL2**，这就是神经网络的第二个卷积层，即 **Layer2**。

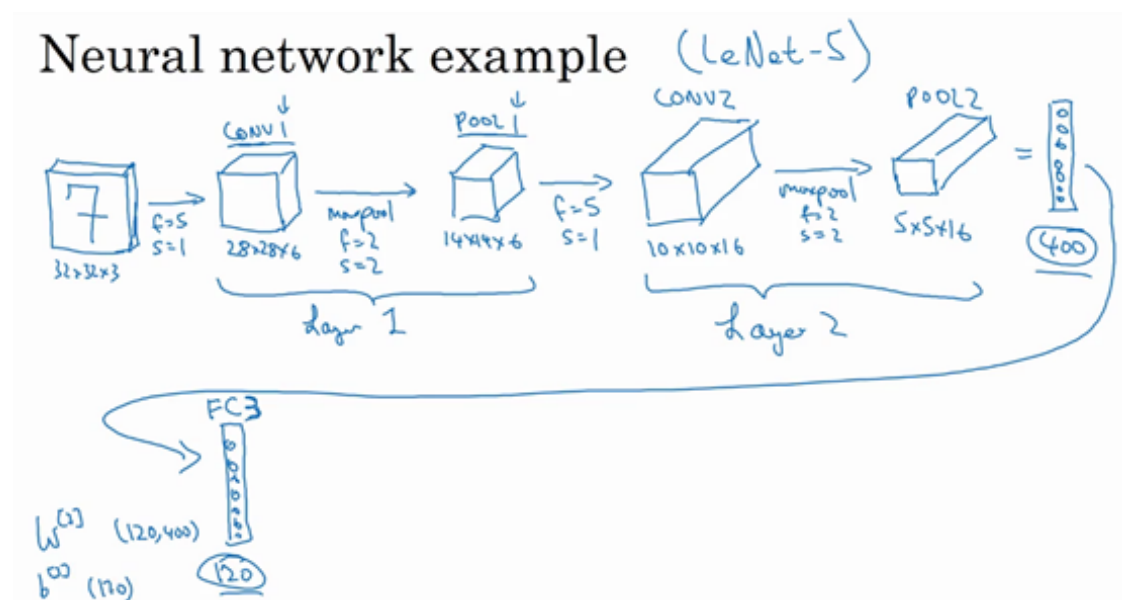


如果对 **Layer1** 应用另一个卷积层，过滤器为 5×5 ，即 $f = 5$ ，步幅是 1，padding 为 0，所以这里省略了，过滤器 16 个，所以 **CONV2** 输出为 $10 \times 10 \times 16$ 。我们看看 **CONV2**，这是

CONV2 层。



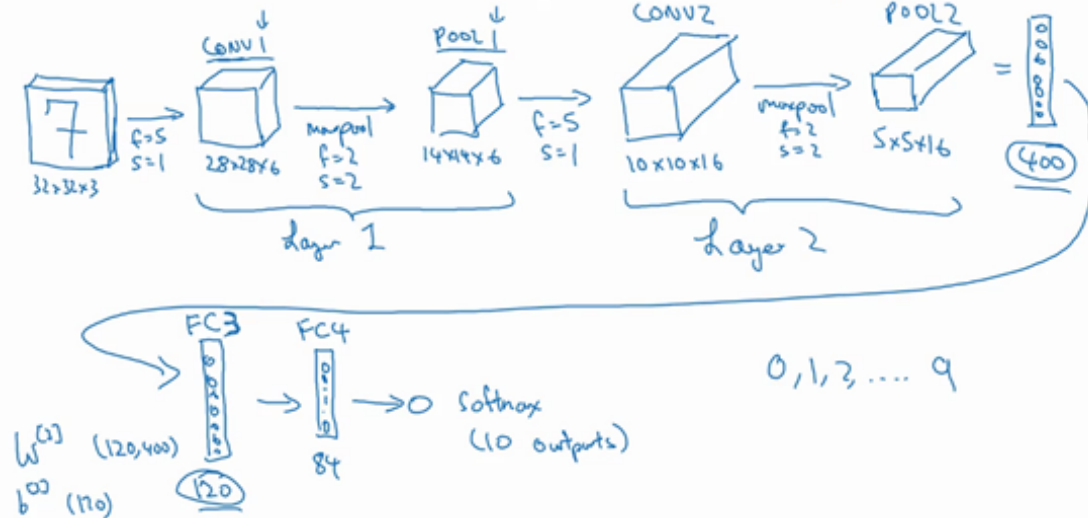
继续执行做大池化计算，参数 $f = 2, s = 2$ ，你能猜到结果么？对 $10 \times 10 \times 16$ 输入执行最大池化计算，参数 $f = 2, s = 2$ ，高度和宽度减半，计算结果猜到了吧。最大池化的参数 $f = 2, s = 2$ ，输入的高度和宽度会减半，结果为 $5 \times 5 \times 16$ ，通道数和之前一样，标记为 **POOL2**。这是一个卷积，即 **Layer2**，因为它只有一个权重集和一个卷积层 **CONV2**。



$5 \times 5 \times 16$ 矩阵包含 400 个元素，现在将 **POOL2** 平整化为一个大小为 400 的一维向量。我们可以把平整化结果想象成这样的一个神经元集合，然后利用这 400 个单元构建下一层。下一层含有 120 个单元，这就是我们第一个全连接层，标记为 **FC3**。这 400 个单元与 120 个单元紧密相连，这就是全连接层。它很像我们在第一和第二门课中讲过的单神经网络层，这是一个标准的神经网络。它的权重矩阵为 $W^{[3]}$ ，维度为 120×400 。这就是所谓的“全连接”，因为这 400 个单元与这 120 个单元的每一项连接，还有一个偏差参数。最后输出 120 个维度，因为有 120 个输出。

然后我们对这个 120 个单元再添加一个全连接层，这层更小，假设它含有 84 个单元，标记为 **FC4**。

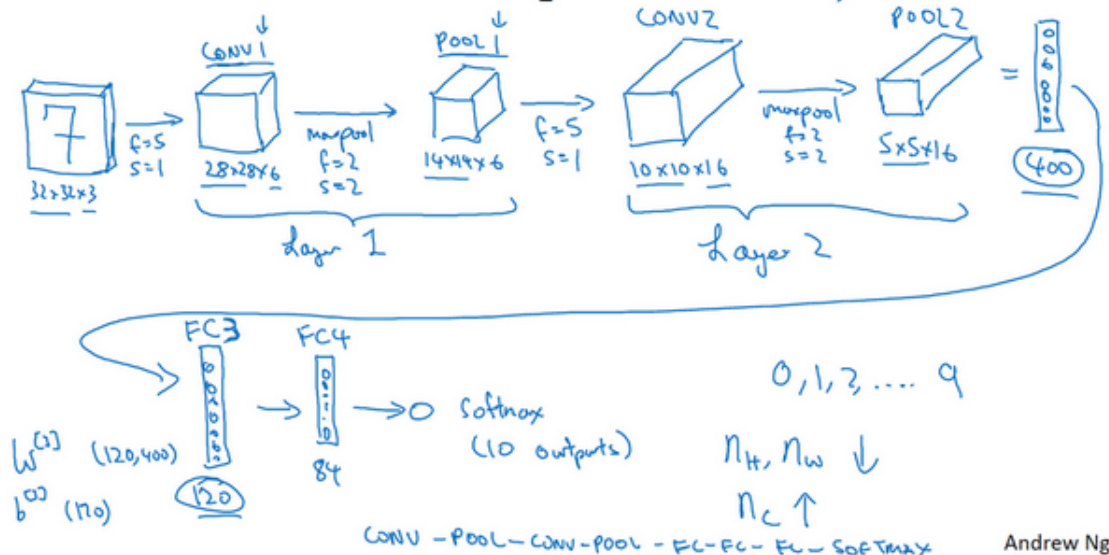
Neural network example (LeNet-5)



最后，用这 84 个单元填充一个 **softmax** 单元。如果我们想通过手写数字识别来识别手写 0-9 这 10 个数字，这个 **softmax** 就会有 10 个输出。

此例中的卷积神经网络很典型，看上去它有很多超参数，关于如何选定这些参数，后面我提供更多建议。常规做法是，尽量不要自己设置超参数，而是查看文献中别人采用了哪些超参数，选一个在别人任务中效果很好的架构，那么它也有可能适用于你自己的应用程序，这块下周我会细讲。

Neural network example (LeNet-5)



CONV - POOL - CONV - POOL - FC - FC - FC - Softmax

Andrew Ng

现在，我想指出的是，随着神经网络深度的加深，高度 n_H 和宽度 n_W 通常都会减少，前面我就提到过，从 32x32 到 28x28，到 14x14，到 10x10，再到 5x5。所以随着层数增加，高度和宽度都会减小，而通道数量会增加，从 3 到 6 到 16 不断增加，然后得到一个全连接层。

在神经网络中，另一种常见模式就是一个或多个卷积后面跟随一个池化层，然后一个或多个卷积层后面再跟一个池化层，然后是几个全连接层，最后是一个 **softmax**。这是神经网络的另一种常见模式。

接下来我们讲讲神经网络的激活值形状，激活值大小和参数数量。输入为 $32 \times 32 \times 3$ ，这些数做乘法，结果为 3072，所以激活值 $a^{[0]}$ 有 3072 维，激活值矩阵为 $32 \times 32 \times 3$ ，输入层没有参数。计算其他层的时候，试着自己计算出激活值，这些都是网络中不同层的激活值形状和激活值大小。

Neural network example

	Activation shape	Activation Size	# parameters
Input:	(32,32,3)	— 3,072 $a^{[0]}$	0
CONV1 (f=5, s=1)	(28,28,8)	6,272	208 ←
POOL1	(14,14,8)	1,568	0 ←
CONV2 (f=5, s=1)	(10,10,16)	1,600	416 ←
POOL2	(5,5,16)	400	0 ←
FC3	(120,1)	120	48,001 }
FC4	(84,1)	84	10,081 }
Softmax	(10,1)	10	841

有几点要注意，**第一，池化层和最大池化层没有参数；第二卷积层的参数相对较少**，前面课上我们提到过，**其实许多参数都存在于神经网络的全连接层**。观察可发现，随着神经网络的加深，激活值尺寸会逐渐变小，如果激活值尺寸下降太快，也会影响神经网络性能。示例中，激活值尺寸在第一层为 6000，然后减少到 1600，慢慢减少到 84，最后输出 **softmax** 结果。我们发现，许多卷积网络都具有这些属性，模式上也相似。

神经网络的基本构造模块我们已经讲完了，一个卷积神经网络包括卷积层、池化层和全连接层。许多计算机视觉研究正在探索如何把这些基本模块整合起来，构建高效的神经网络，整合这些基本模块确实需要深入的理解。