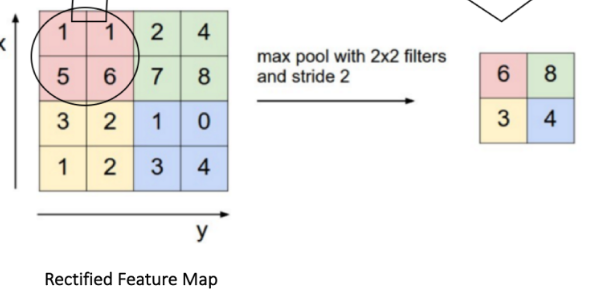


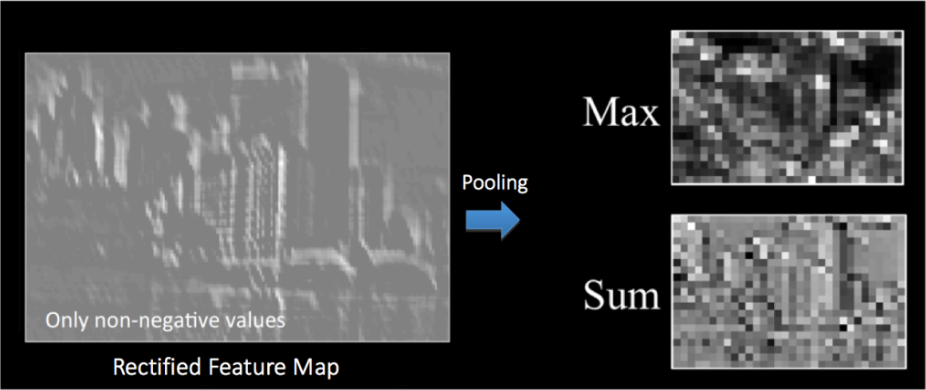
Pooling

类似于卷积的过程，不过不是通过滤波器来提取特征，而是提取每一个矩阵中的最大值/最小值/平均值来简化feature map，每个feature map都会进行池化



Stride: 步数

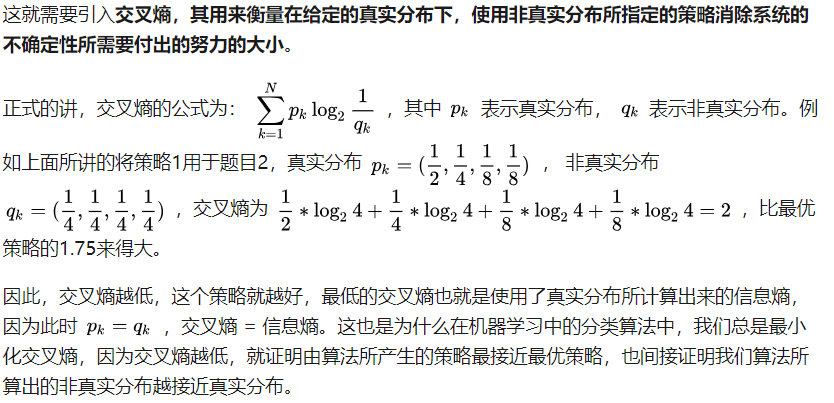
Depth: 深度，指滤波器的数量



The function of Pooling is to progressively reduce the spatial size of the input representation [[4](http://cs231n.github.io/convolutional-networks/)]. In particular, pooling

* makes the input representations (feature dimension) smaller and more manageable
* reduces the number of parameters and computations in the network, therefore, controlling [overfitting](https://en.wikipedia.org/wiki/Overfitting) [[4](http://cs231n.github.io/convolutional-networks/)]
* makes the network invariant to small transformations, distortions and translations in the input image (a small distortion in input will not change the output of Pooling – since we take the maximum / average value in a local neighbourhood).
* helps us arrive at an almost scale invariant representation of our image (the exact term is “equivariant”). This is very powerful since we can detect objects in an image no matter where they are located (read [[18](https://github.com/rasbt/python-machine-learning-book/blob/master/faq/difference-deep-and-normal-learning.md)] and [[19](https://www.quora.com/How-is-a-convolutional-neural-network-able-to-learn-invariant-features)] for details).

交叉熵：



正则化：

