1 Convolutional Neural Network

1.1 Convolution Operation of Matrix

We denote the convolution operation as the following

$$Z = A * W + b$$

Where A, W, b are matrices with proper dimension. The star $*$ represent the convolution operation of the two matrix A and W. We define the matrix A with dimension $I^{in} \times J^{in}$. The matrix W has the dimension $P \times Q$ and we assume $P \leq I^{in}$, $Q \leq J^{in}$. The matrix Z has the dimension $I^{out} \times J^{out}$. And b is a scalar constant. The convolution operation is defined as

$$z_{ij} = \sum_{p=i}^{i+P} \sum_{q=j}^{j+Q} a_{pq} w_{pq} + b$$

The input dimension and output dimension have the following relationship

$$P = I^{in} - I^{out} + 1$$
$$Q = J^{in} - J^{out} + 1$$

Namely,

$$I^{out} = I^{in} - P + 1$$
$$J^{out} = J^{in} - Q + 1$$

An intuitive way of thinking the above operation is we put matrix W on top of matrix A and align the top left corner, meaning that a_{11} aligns with w_{11}. And for each overlapped element of matrix A and Z, we multiply them and get each product. Finally we sum all the product, add the bias b, then we get z_{11}. To get z_{12}, we still put W on top of A, but this time we align a_{12} with w_{11}. Then we follow the same procedure to calculate the product and summation. We keep going to slide W matrix along the matrix A by one element each time, we can get all the element of matrix Z.

1.2 Convolution Operation of Matrix with Padding

In the above convolution operation, the dimension of the output matrix is usually smaller than the input matrix. In order to keep the output dimension the same as the input dimension, we can pad the input matrix with zeros along the border.

1.3 Convolution Operation of Matrix with Stride

In the previous convolution operation, we slide W matrix along the matrix A by one element each time. This means if the first step we place w_{11} on top of a_{11}, next time we move W horizontally and place w_{11} on top of a_{12}. In the

convolution operation with stride, we slide W matrix by several element each time. We call the slide length stride. For example, if $stride = 2$, then after we place w_{11} on the top of a_{11}, we place w_{11} on top of a_{13}.
The convolution operation is defined as

$$z_{ij} = \sum_{p=i'}^{i^{'}+P} \sum_{q=j'}^{j^{'}+Q} a_{pq} w_{pq} + b$$

Where

$$\begin{aligned} i^{'} &= (i-1) * stride + 1 \\ j^{'} &= (j-1) * stride + 1 \end{aligned}$$

And the output dimension is

$$\begin{aligned} I^{out} &= \frac{I^{in} - P + 1}{stride} \\ J^{out} &= \frac{J^{in} - Q + 1}{stride} \end{aligned}$$

1.4 Convolution Operation of Tensors

We denote the convolution operation as the following(same as above)

$$Z = A * W + b$$

Where A, W, b are tensors with proper rank and dimension. The star $*$ represent the convolution operation of the two matrix A and W. We define the tensor A with $rank = 3$ and dimension $I^{in} \times J^{in} \times K$. The tensor W has rank 4 and the dimension $P \times Q \times K \times N$. The tensor Z has rank 3 and dimension $I^{out} \times J^{out} \times N$. And b is a vector of dimension N. The convolution operation without stride is defined as

$$z_{ijn} = \sum_{p=i}^{i+P} \sum_{q=j}^{j+Q} \sum_{k=1}^{K} a_{pqk} w_{pqk}^{(n)} + b^{(n)}$$

Example Take A as a $2 \times 2 \times 3$ tensor, and W as a $2 \times 2 \times 3 \times 2$ tensor, then

$$\begin{aligned} z_{111} = {} & a_{111} w_{111}^{(1)} + a_{121} w_{121}^{(1)} + a_{211} w_{211}^{(1)} + a_{221} w_{221}^{(1)} \\ & + a_{112} w_{112}^{(1)} + a_{122} w_{122}^{(1)} + a_{212} w_{212}^{(1)} + a_{222} w_{222}^{(1)} \\ & + a_{113} w_{113}^{(1)} + a_{123} w_{123}^{(1)} + a_{213} w_{213}^{(1)} + a_{223} w_{223}^{(1)} \\ & + b^{1} \end{aligned}$$

$$\begin{aligned} z_{112} = {} & a_{111}w^{(2)}_{111} + a_{121}w^{(2)}_{121} + a_{211}w^{(2)}_{211} + a_{221}w^{(2)}_{221} \\ & + a_{112}w^{(2)}_{112} + a_{122}w^{(2)}_{122} + a_{212}w^{(2)}_{212} + a_{222}w^{(2)}_{222} \\ & + a_{113}w^{(2)}_{113} + a_{123}w^{(2)}_{123} + a_{213}w^{(2)}_{213} + a_{223}w^{(2)}_{223} \\ & + b^1 \end{aligned}$$

The convolution operation with stride is defined as

$$z_{ijn} = \sum_{p=i'}^{i^{'}+P} \sum_{q=j'}^{j+Q} \sum_{k=1}^{K} a_{pqk} w^{(n)}_{pqk} + b^{(n)}$$

Where

$$\begin{aligned} i^{'} &= (i-1) * stride + 1 \\ j^{'} &= (j-1) * stride + 1 \end{aligned}$$

1.5 Pooling Operation

Pooling operation is used to reduce the dimension of the matrix therefore reduce the number of parameters in the following hidden layers. The max pooling operation is defined as

$$z_{ijk} = max(a_{i' j' k})$$

where

$$\begin{aligned} i^{'} &\in [(i-1) * stride + 1, i * stride + 1] \\ j^{'} &\in [(j-1) * stride + 1, j * stride + 1] \end{aligned}$$

The average pooling operation is defined as

$$z_{ijk} = average(a_{i' j' k})$$

And the definition of $a_{i'}$ and $a_{j'}$ is the same as in max pooling.

1.6 Fully Connected Layer

The output of the convolution layer is tensor. However, most final output of convolutional neural network has to be a vector. So in the final step, we need an operation to reduce the rank of the tensor. If we take the input tensor A which has dimension IxJxK, and W which has dimension IxJxKxN, then the output has dimension 1x1x1xN. This dimension can be view effectively has a vector which as dimension N. The output is connected with every element of the input tensor, so this layer is called fully connected layer.

2 LeNet-5 Example

We work out an exercise to calculate the output dimensions and number of parameters in convolutional neural network. Below Conv is convolution layer, MP is max pooling layer, FC is fully connected layer.

Type	input	parameter	padding	stride	output	number of parameters
Conv	32x32x1	5x5x1x6	no	1	28x28x6	(5x5x1+1)x6
MP	28x28x6	2x2	no	2	14x14x6	n/a
Conv	14x14x6	5x5x6x16	no	1	10x10x16	(5x5x6+1)x16
MP	10x10x16	2x2	no	2	5x5x16	n/a
FC	5x5x16	5x5x16x120	n/a	n/a	1x1x120	(5x5x16+1)x120
FC	120	120x84	n/a	n/a	84	120x84 + 84
FC	84	84x10	n/a	n/a	10	84x10+10