Task

The input X is a 7 x 7 image. The convolution layer C is produced by convolving X with a 3x3 filter W^x with stride 1, plus bias matrix W_0^x. The activation layer A is produced by applying the ReLU activation function to C. A max pooling of 3x3 with a stride 2 is then applied to C to produce the pooling layer P. A fully connected vector layer $\vec{P}$ is then produced by concatenating rows of P. The output layer consists of one node y. It is produced by $y = \sigma((W^0)^\top \vec{P} + W_0^o))$, where W^o and W_0^o are output layer weight matrix and bias vector repectively.

1. Determine the dimension for each layer.

From input image to convolution layer

The input of the NN X is a 7×7 matrix. The filter for the convolution has dimension 3×3, and the stride of the covolution is 1. So we have the following equation:

$$
\begin{aligned}
C(r,c) =& X * W_0^x(r,c) + w_0^x \\
=& \sum_{i=1}^{K} \sum_{j=1}^{K} X((r+i) \times s - 1, (c+j)) \times s - 1) W(i,j) + W_0 \\
=& \sum_{i=1}^{3} \sum_{j=1}^{3} X(r+i-1, c+j-1) W(i,j) + W_0
\end{aligned}
$$

Because we are not instructed to add zero-padding to the image, The image should shrink to image with width of the image as $\frac{N-K}{s} + 1 = \frac{7-3}{1} + 1 = 5$. So the convolution image should be a 5×5 image.

From convolution layer to activation layer

Since the activation function takes in a scalar and output a scalar, the size of the activation image A should have the same size with the convolution image. So $A^{N_r^a \times N_c^a}$ is a 5×5 matrix.

From activation layer to pooling layer

In the pooling layer, we shrink a $d \times d$ part of image to 1 pxiels by averaging the pxiel values in this $d \times d$ part.

$$
\begin{aligned}
P(r,c) &= \max_{\substack{1\le i\le d\\1\le j\le d}} A((r-1)\times s+i,(c-1)\times s+j)\\
&= \max_{\substack{1\le i\le 3\\1\le j\le 3}} A((r-1)\times 2+i,(c-1)\times 2+j)
\end{aligned}
$$

The heigth and width of the pooling image is shrinked to $\frac{N_r^a-d}{s}+1=\frac{5-3}{2}+1=2$. So the pooling image is a 2×2 matrix

From pooling layer to fully connected layer

In this layer, we sketch the image into a vector, so the dimsion of $\vec{P}$ is 4×1

From fully conncected layer to output layer

Since this is a binary classification problem, using sigmoid function as the activation function. The output layer is a scalar which tells the probability of outputing 1.

2. Perform forward propagation layer by layer to compute the values for each layer, the estimated output value $\hat{y}$, and the gradient of the output $\bigtriangledown\hat{y}$, given y=1, using negative log conditional likelihood loss function.

The C image is:

$$
\begin{bmatrix}
0 & 0.8 & 0.7 & 0.6 & 0\\
0 & 0.8 & 0.7 & 0.6 & 0\\
0 & 0.8 & 0.7 & 0.6 & 0\\
0 & 0.8 & 0.7 & 0.6 & 0\\
0 & 0.8 & 0.7 & 0.6 & 0
\end{bmatrix}
$$

The A image is: still this matrix, because all the value is larger than 0.

In the pooling layer P is:

$$
\begin{bmatrix}
0.8 & 0.7\\
0.8 & 0.7
\end{bmatrix}
$$

The fulling conncected layer stretchs the pooling layer P into a vector $\vec{P}$:

$$
\vec{P}=[0.8,0.7,0.8,0.7]^\top
$$

The output layer:

$$z=(W^{o})^{\top}\vec{P}+W_0^o=0.75-0.2=0.55$$
$$\hat{y}=\frac{\exp(z)}{\exp(z)+1}=.634$$

And $\hat{y}$ is the probability of getting 1 as the result.

The loss function:

$$-\log p(y|x)=-y\log\hat{y}=.456$$

So the gradient of the output $\bigtriangledown\hat{y}=-\frac{y}{\hat{y}}=-\frac{1}{.634}=-1.58$

3. Given graident of $\hat{y}$, perform back-propagation to obtain the weight matrix gradient $\bigtriangledown W^o$ and bias vector gradient $\bigtriangledown W_0^o$ for the output layer and the weight matrix gradient $\bigtriangledown W^x$ and bias matrix gradient $\bigtriangledown W_0^x$ respectively for the convolution layer.

From output layer to fully connected layer

$$\begin{aligned}\bigtriangledown W^o=&\frac{\partial z}{\partial w^o}\frac{\partial\hat{y}}{\partial z}\bigtriangledown\hat{y}\\=&\vec{P}\hat{y}(1-\hat{y})-\frac{y}{\hat{y}}\\=&[.8\quad .7\quad .8\quad .7]^{\top}*.634(1-.634)*(-1.58)\\=&[-.293\quad -.257\quad -.293\quad -.257]^{\top}\end{aligned}$$

$$\begin{aligned}\bigtriangledown W_0^o=&\frac{\partial z}{\partial w_0^o}\frac{\partial\hat{y}}{\partial z}\bigtriangledown\hat{y}\\=&1*\hat{y}(1-\hat{y})-\frac{y}{\hat{y}}\\=&.634(1-.634)*(-1.58)\\=&-.367\end{aligned}$$

From fully connected layer to max pooling layer

$$\bigtriangledown_P=\begin{bmatrix}-.293&-.257\\-.293&-.257\end{bmatrix}$$

From max pooling to activation

$$\bigtriangledown A[(r-1)\times s+i][(c-1)\times s+j]=\begin{cases}\bigtriangledown P[r][c] \text{ if } i=i*[r] \text{ and } j=j*[c]\\ 0 \text{ otherwise}\end{cases}$$

So the $\bigtriangledown A$ is:

$$\bigtriangledown C=\begin{bmatrix}0 & -.293 & -.257 & 0 & 0\\ 0 & 0 & 0 & 0 & 0\\ 0 & -.293 & -.257 & 0 & 0\\ 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0\end{bmatrix}$$

From activation to convolution layer

$$\bigtriangledown C=\begin{bmatrix}0 & -.293 & -.257 & 0 & 0\\ 0 & 0 & 0 & 0 & 0\\ 0 & -.293 & -.257 & 0 & 0\\ 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0\end{bmatrix}$$

From the convolution layer to input layer

$$\begin{aligned}
\bigtriangledown W^x &= \frac{\partial C}{\partial W^x}\bigtriangledown C\\
&=\sum_{r=1}^{N_r^c}\sum_{c=1}^{N_c^c}\frac{\partial C[r][c]}{\partial w^x}\bigtriangledown C\\
&=\sum_{r=1}^{N_r^c}\sum_{c=1}^{N_c^c}\begin{bmatrix}\frac{\partial C[r][c]}{\partial W^x[1][1]} & \frac{\partial C[r][c]}{\partial W^x[1][2]} & \dots & \frac{\partial C[r][c]}{\partial W^x[1][K]}\\ \dots & & & \\ \frac{\partial C[r][c]}{\partial W^x[K][1]} & \frac{\partial C[r][c]}{\partial W^x[K][2]} & \dots & \frac{\partial C[r][c]}{\partial W^x[1][K]}\end{bmatrix}\bigtriangledown C[r][c]\\
&=\sum_{r=1}^{N_r^c}\sum_{c=1}^{N_c^c}\begin{bmatrix}X[r][c] & X[r][c+1] & \dots & X[r][c+K]\\ \dots & & & \\ X[r+K][c] & X[r+K][c+1] & \dots & X[r+K][c+K]\end{bmatrix}\bigtriangledown C[r][c]\\
&=\begin{bmatrix}X[1][2] & X[1][3] & X[1][4]\\ X[2][2] & X[2][3] & X[2][4]\\ X[3][2] & X[3][3] & X[3][4]\end{bmatrix}*-.293+\begin{bmatrix}X[2][3] & X[2][4] & X[2][5]\\ X[3][3] & X[3][4] & X[3][5]\\ X[4][3] & X[4][4] & X[4][5]\end{bmatrix}*\\
&\quad -.257\begin{bmatrix}X[3][2] & X[3][3] & X[3][4]\\ X[4][2] & X[4][3] & X[4][4]\\ X[5][2] & X[5][3] & X[5][4]\end{bmatrix}*-.293+\begin{bmatrix}X[3][3] & X[3][4] & X[3][5]\\ X[4][3] & X[4][4] & X[4][5]\\ X[5][3] & X[5][4] & X[5][5]\end{bmatrix}*-.257\\
&=\begin{bmatrix}0 & -.514 & -.586\\ 0 & -.514 & -.586\\ 0 & -.514 & -.586\end{bmatrix}
\end{aligned}$$

$$\triangledown W^x = \frac{\partial C}{\partial W_0^x} \triangledown C = \sum_{r=1}^{N_r^c} \sum_{c=1}^{N_c^c} \triangledown[r][c] = -1.613$$

4. Update the weights for the convolution and the output layers with their estimted gradient, using a learning rate of .5, and then compute the new output value $\hat{y}$ using the updated weight matrices. Verify that the new $\hat{y}$ reduced the output loss function, as compared to the previous $\hat{y}$.

$$W^x = W^x - .5 * \triangledown W^x = \begin{bmatrix} .2 & .356 & .007 \\ .1 & .456 & .107 \\ .3 & .657 & -.193 \end{bmatrix}$$

$$W_0^x = W_0^x - .5 * \triangledown W_0^x = .806$$

$$W^o = W^o - .5 * \triangledown W^o = [0.3465, 0.2285, 0.4465, 0.5285]^\top$$

$$W_0^o = W_0^o - .5 * \triangledown W_0^o = -0.017$$

After the convolution layer:

$$C = \begin{bmatrix} 0.806 & 0.727 & 2.277 & 1.406 & 0.806 \\ 0.806 & 0.727 & 2.277 & 1.406 & 0.806 \\ 0.806 & 0.727 & 2.277 & 1.406 & 0.806 \\ 0.806 & 0.727 & 2.277 & 1.406 & 0.806 \\ 0.806 & 0.727 & 2.277 & 1.406 & 0.806 \end{bmatrix}$$

After the activation layer:

$$C = \begin{bmatrix} 0.806 & 0.727 & 2.277 & 1.406 & 0.806 \\ 0.806 & 0.727 & 2.277 & 1.406 & 0.806 \\ 0.806 & 0.727 & 2.277 & 1.406 & 0.806 \\ 0.806 & 0.727 & 2.277 & 1.406 & 0.806 \\ 0.806 & 0.727 & 2.277 & 1.406 & 0.806 \end{bmatrix}$$

After the max pooling:

$$P = \begin{bmatrix} 2.277 & 2.277 \\ 2.277 & 2.277 \end{bmatrix}$$

After flatten:

$$\vec{P} = \begin{bmatrix} 2.277 & 2.277 & 2.277 & 2.277 \end{bmatrix}$$

After fully connected layer:

$$z = 3.51285$$

After sigmoid function:

$$\hat{y} = \frac{\exp(z)}{1 + \exp(z)} = 0.97105119$$

The new loss:

$$\mathcal{L} = -\log(\hat{y}) = 0.0293761$$