Introduction to Machine Learning
Homework 8: Convolutional Neural Networks

Prof. Sundeep Rangan

1. (a) Both indices go over the range of $W[k_1, k_2]$: $0 \leq k_1, k_2 < 2$.

   (b) Since, X is 6×5 and W is 2×2 and we are selecting valid locations only, the size will of Z will be
   $$(6 - 2 + 1) \times (5 - 2 + 1) = 5 \times 4.$$

   (c) We have that
   $$Z[i, j] = X[i, j] + X[i + 1, j] - X[i, j + 1] - X[i + 1, j + 1].$$
   So, $Z[i, j]$ will be the largest positive value when there is a large negative change across one column. This occurs at $(i, j) = (1, 3)$:
   $$Z[1, 3] = X[1, 3] + X[2, 3] - X[1, 4] - X[2, 4] = 3 + 3 - 0 - 0 = 6.$$
   We get the same value at $(2, 3)$ and $(3, 3)$.

   (d) For a negative value, we need there to be a large positive change across one column, which occurs at
   $$Z[1, 0] = X[1, 0] + X[2, 0] - X[1, 0] - X[2, 0] = 0 + 0 - 3 - 3 = -6.$$
   We get the same value at $(2, 0)$ and $(3, 0)$.

   (e) You can take $(i, j) = (1, 1)$ or $(1, 2)$. For example,
   $$Z[1, 1] = X[1, 1] + X[2, 1] - X[1, 2] - X[2, 2] = 3 + 3 - 3 - 3 = 0.$$

2. (a) Since each kernel in W is 3×3, each channel of the output is
   $$(48 - 3 + 1) \times (64 - 3 + 1) = 46 \times 62.$$
   There are 20 output channels, so Z is $46 \times 62 \times 20$.

   (b) Since W is $3 \times 3 \times 10 \times 20$, there are 10 input channels and 20 output channels.

   (c) Each output of $Z[i, j, m]$ requires summations over the indices
   $$0 \leq k_1, k_2 < 3, \quad 0 \leq n < 10.$$
   Therefore, there are (3)(3)(10) multiplications for each output of Z. Since there are (46)(62)(20) outputs, there are a total of
   $$(46)(62)(20)(3)(3)(10) = 5.133(10)^6 \text{ multiplications.}$$
   You can see why computing outputs in deep networks takes many operations.

(d) The number of parameters in W and b are:

$$\begin{aligned} W:&\quad (3)(3)(10)(20) = 1800 \text{ parameters} \\ b:&\quad 20 \text{ parameters.} \end{aligned}$$

So, there are a total of 1820 parameters.

3. Suppose that a convolutional layer as a linear convolution followed by a sigmoid activation,

$$\begin{aligned} Z[i,j,m] &= \sum_{k_1}\sum_{k_2}\sum_{n} W[k_1,k_2,n,m]X[i+k_1,j+k_2,n] + b[m], \\ U[i,j,m] &= 1/(1+\exp(-Z[i,j,m])). \end{aligned}$$

Suppose that during back-propagation, we have computed the gradient $\partial J/\partial U$ for some loss function J. That is, we have computed $\partial J/\partial U[i,j,m]$. Show how to compute the following:

(a) We have

$$\frac{\partial U[i,j,m]}{\partial Z[i,j,m]} = \frac{\exp(-Z[i,j,m])}{(1+\exp(-Z[i,j,m])^2} = U[i,j,m](1-U[i,j,m]).$$

By chain rule,

$$\frac{\partial J}{\partial Z[i,j,m]} = \frac{\partial J}{\partial U[i,j,m]}\frac{\partial U[i,j,m]}{\partial Z[i,j,m]} = \frac{\partial J}{\partial U[i,j,m]}U[i,j,m](1-U[i,j,m]).$$

(b) The gradient components $\partial J/\partial W[k_1,k_2,n,m]$. From the convolution equation,

$$\frac{\partial Z[i,j,m]}{\partial W[k_1,k_2,n,m]} = X[i+k_1,j+k_2,n].$$

By chain rule,

$$\begin{aligned} \frac{\partial J}{\partial W[k_1,k_2,n,m]} &= \sum_{i,j}\frac{\partial J}{\partial Z[i,j,m]}\frac{\partial Z[i,j,m]}{\partial W[k_1,k_2,n,m]} \\ &= \sum_{i,j}\frac{\partial J}{\partial Z[i,j,m]}X[i+k_1,j+k_2,n]. \end{aligned}$$

(c) We want to first compute the partial derivatives,

$$\frac{\partial Z[i',j',m]}{\partial X[i,j,n]},$$

for all output components $Z[i',j',m]$ and inputs $X[i,j,n]$. Note that we had to add the indices i',j' at the output, to differentiate between the input indices i,j. To compute this derivative, we need to write $Z[i',j',m]$ in terms of the inputs $X[i,j,n]$. This is matter of re-indexing. First, rewrite the summation in the convolution as,

$$Z[i',j',m] = \sum_{k_1}\sum_{k_2}\sum_{n} W[k_1,k_2,n,m]X[i'+k_1,j'+k_2,n] + b[m].$$

All we have done here is replace i, j with i', j'. Next make the substitution,

$$i = i' + k_1, \quad j = j' + k_2 \Rightarrow k_1 = i - i', \quad k_2 = j - j'.$$

Then, we can sum over i, j instead of over k_1, k_2:

$$Z[i', j', m] = \sum_i \sum_j \sum_n W[i - i', j - j', n, m] X[i, j, n] + b[m].$$

Now, we have $Z[i', j', m]$ in terms of inputs $X[i, j, n]$.
From this, we see that

$$\frac{\partial Z[i', j', m]}{\partial X[i, j, n]} = W[i - i', j - j', n, m].$$

Hence, by chain rule,

$$\begin{aligned}
\frac{\partial J}{\partial X[i, j, n]} &= \sum_{i'} \sum_{j'} \sum_m \frac{\partial J}{\partial Z[i', j', m]} \frac{\partial Z[i', j', m]}{\partial X[i, j, n]} \\
&= \sum_{i'} \sum_{j'} \sum_m \frac{\partial J}{\partial Z[i', j', m]} W[i - i', j - j', n, m].
\end{aligned}$$

If you got this far, you will get full marks. But, if we let $k_1 = i - i'$ and $k_2 = j - j'$ and sum over k_1, k_2 instead of i', j', we get

$$\frac{\partial J}{\partial X[i, j, n]} = \sum_{k_1} \sum_{k_2} \sum_n \frac{\partial J}{\partial Z[i - k_1, j - k_2, m]} W[k_1, k_2, n, m].$$

We see that the gradient is also a convolution, but with the reversal.

4. (a) For the mini-batch case, we need to add an index over the samples in the mini-batch. In this case, X, Z and U are fourth-order tensors:

$$X[\ell, i, j, n], \quad Z[\ell, i, j, m], \quad U[\ell, i, j, m],$$

where ℓ is the sample index; i, j are the row-column indices; n is the input channel index; and m is the output channel index.

(b) The equations would be mostly the same, just add the sample index ℓ:

$$\begin{aligned}
Z[\ell, i, j, m] &= \sum_{k_1} \sum_{k_2} \sum_n W[k_1, k_2, n, m] X[\ell, i + k_1, j + k_2, n] + b[m], \\
U[\ell, i, j, m] &= 1/(1 + \exp(-Z[\ell, i, j, m])).
\end{aligned}$$

(c) The gradients are identical, except that we add the sample index:

$$\begin{aligned}
\frac{\partial J}{\partial Z[\ell, i, j, m]} &= \frac{\partial J}{\partial U[\ell, i, j, m]} U[\ell, i, j, m](1 - U[\ell, i, j, m]), \\
\frac{\partial J}{\partial W[k_1, k_2, n, m]} &= \sum_{\ell, i, j} \frac{\partial J}{\partial Z[\ell, i, j, m]} X[\ell, i + k_1, j + k_2, n] \\
\frac{\partial J}{\partial X[\ell, i, j, n]} &= \sum_{k_1} \sum_{k_2} \sum_m \frac{\partial J}{\partial Z[\ell, i - k_1, j - k_2, m]} W[k_1, k_2, n, m].
\end{aligned}$$