CS280 Spring 2023 Assignment 2
Part A

Convolutional Neural Network

March 15, 2023

Name: Bingnan Li

Student ID: 2020533092

1. Convolution Cost (10 points)

Assume an input of shape $c_i \times h \times w$ and a convolution kernel of shape $c_o \times c_i \times k_h \times k_w$, padding of (p_h, p_w) , and stride of (s_h, s_w) .

- What is the computational cost (multiplications and additions) for the forward propagation?
  Solution:

  The output size has the formula as follows:

$$c_{out} = \#\{kernels\}$$

$$h_{out} = \left\lfloor \frac{h_{in} + 2 \times padding\_size_h - (kernel\_size_h - 1) - 1}{stride\_size_h} + 1 \right\rfloor$$

$$w_{out} = \left\lfloor \frac{w_{in} + 2 \times padding\_size_w - (kernel\_size_w - 1) - 1}{stride\_size_w} + 1 \right\rfloor$$

  Substitute $h_{in}, w_{in}, padding\_size_h, padding\_size_w, stride\_size_h, stride\_size_w, kernel\_size_h$ and $kernel\_size_w$ with $h, w, p_h, p_w, s_h, s_w, k_h$ and k_w respectively, the output size is

$$c_{out} = c_0$$

$$h_{out} = \left\lfloor \frac{h + 2p_h - k_h}{s_h} + 1 \right\rfloor$$

$$w_{out} = \left\lfloor \frac{w + 2p_w - k_w}{s_w} + 1 \right\rfloor$$

  Then for each element in output, it is calculated by $c_i \times k_h \times k_w$ times multiplications and $c_i \times k_h \times k_w - 1$ times additions. Thus, the computational cost is

$$cost = c_0 \left\lfloor \frac{h + 2p_h - k_h}{s_h} + 1 \right\rfloor \left\lfloor \frac{w + 2p_w - k_w}{s_w} + 1 \right\rfloor (2c_i k_h k_w - 1)$$

- What is the memory footprint?
  Solution:

  Based on the previous analysis, we know the output size is

$$\left(c_0, \left\lfloor \frac{h + 2p_h - k_h}{s_h} + 1 \right\rfloor, \left\lfloor \frac{w + 2p_w - k_w}{s_w} + 1 \right\rfloor \right)$$

  So the memory footprint of output is

$$c_0 \left\lfloor \frac{h + 2p_h - k_h}{s_h} + 1 \right\rfloor \left\lfloor \frac{w + 2p_w - k_w}{s_w} + 1 \right\rfloor$$

  and the total memory footprint including input and kernels is

$$c_0 \left\lfloor \frac{h + 2p_h - k_h}{s_h} + 1 \right\rfloor \left\lfloor \frac{w + 2p_w - k_w}{s_w} + 1 \right\rfloor + c_i hw + c_0 c_i k_h k_w$$

2. Convolution Kernel (10 points)

Assume there are two convolution kernels of size k_1 and k_2 respectively (with no nonlinear activation function in-between).

- Prove that the results of the two convolution operations can be expressed by a single convolution operation.
  Proof:

  Define the input as X and Y, Z are the output of the first and the second convolution layer, where

$$Y = conv_1(X)$$
$$Z = conv_2(Y)$$

  Then define $A(x, y)$ as the element at position (x, y) of A. By the definition of convolution operation, we have

$$Z(i,j) = \sum_{p=0}^{k_2-1}\sum_{q=0}^{k_2-1} kernel_2(p,q)Y(i+p,j+q)$$
$$Y(i+p,j+q) = \sum_{m=0}^{k_1-1}\sum_{n=0}^{k_1-1} kernel_1(m,n)X(i+p+m,j+q+n)$$

  Substitute $Y(i+p, j+q)$ into $Z(i, j)$ and we get:

$$\begin{aligned}
Z(i,j) &= \sum_{p=0}^{k_2-1}\sum_{q=0}^{k_2-1} kernel_2(p,q)\sum_{m=0}^{k_1-1}\sum_{n=0}^{k_1-1} kernel_1(m,n)X(i+p+m,j+q+n)\\
&= \sum_{p=0}^{k_2-1}\sum_{q=0}^{k_2-1}\sum_{m=0}^{k_1-1}\sum_{n=0}^{k_1-1} kernel_2(p,q)kernel_1(m,n)X(i+p+m,j+q+n)\\
&= \sum_{a=0}^{k_3-1}\sum_{b=0}^{k_3-1} kernel_3(a,b)X(i+a,j+b)
\end{aligned}$$

  where $k_3 = k_2 + k_1 - 1$ and

$$\begin{aligned}
kernel_3(a,b) &= \sum_{x=\max(0,a-k_1+1)}^{\min(a,k_2-1)}\ \sum_{x=\max(0,b-k_1+1)}^{\min(b,k_2-1)} kernel_2(x,y)\cdot kernel_1(a-x,b-y)\\
&= kernel_2 * kernel_1
\end{aligned}$$

  Remark: $kernel_3$ is the "real" convolution result between $kernel_1$ and $kernel_2$. It's different from the image convolution operation described in the question, which is actually correlation operation.

  Since $Z(i,j) = \sum_{a=0}^{k_3-1}\sum_{b=0}^{k_3-1} kernel_3(a,b)X(i+a,j+b)$, $Z(i,j)$ can be computed from single convolution.

- What is the dimensionality of the equivalent single convolution?
  Solution:

  Based on the previous proof, we know $Z(i,j) = \sum_{a=0}^{k_3-1}\sum_{b=0}^{k_3-1} kernel_3(a,b)X(i+a,j+b)$ where $k_3 = k_1 + k_2 - 1$, we know that the kernel size of equivalent single convolution is $(k_1 + k_2 - 1) \times (k_1 + k_2 - 1)$.

- Is the converse true, i.e., Can a convolution operation be decomposed into two smaller convolution operations?
  Solution:

  The converse is false,