Homework 2

Guilherme Albertini

October 9, 2022

Theory

Problem 1.1: Convolutional Neural Netoworks

1. Given an input image of dimension 21×12, what will be output dimension after applying a convolution with 4×5 kernel, stride of 4, and no padding?

   5×2

2. Given an input of dimension $C \times H \times W$ what will be the dimension of the output of a convolutional layer with kernel of size $K \times K$, padding P, stride S, dilation D, and F filters. Assume that H $\geq$ K, W $\geq$ K.

Define Padding along height on top P_{H1}

Define Padding along height on bottom P_{H2}

Define Padding along width on left P_{W1}

Define Padding along width on right P_{W2}

Define Kernel width K_H

Define Kernel height K_W

Define Stride horizontal S_W

Define Stride vertical S_H

Define Batch Count B

Effect of adding padding and applying kernel to dimensions:

$$\begin{aligned}
H_P &= P_{H1} + P_{H2} + H \\
W_P &= P_{W1} + P_{W2} + W \\
H_{PK} &= H_1 - [D_H(K_H - 1) + 1] \\
&= P_{H1} + P_{H2} + H - [D_H(K_H - 1) + 1] \\
W_{PK} &= W_1 - [D_W(K_W - 1) + 1] \\
&= P_{W1} + P_{W2} + W - [D_W(K_W - 1) + 1]
\end{aligned}$$

Considering stride to dimensions:

$$\begin{aligned}
H_{PKS} &= \left\lfloor \frac{H_P - [D_H(K_H - 1) + 1] + S_H}{S_H} \right\rfloor \\
&= \left\lfloor \frac{P_{H1} + P_{H2} + H - [D_H(K_H - 1) + 1]}{S_H} \right\rfloor + 1 \\
W_{PKS} &= \left\lfloor \frac{W_P - [D_W(K_W - 1) + 1] + S_W}{S_W} \right\rfloor \\
&= \left\lfloor \frac{P_{W1} + P_{W2} + W - [D_W(K_W - 1) + 1]}{S_W} \right\rfloor + 1
\end{aligned}$$

We can make simplifcations that I think are implied here:

$$
\begin{aligned}
S &= S_W = S_H \\
D &= D_W = D_H \\
K &= K_W = K_H \\
B &= 1 \\
P &= P_{W1} + P_{W2} = P_{H1} + P_{H2}
\end{aligned}
$$

Thus the output dimension is:

$$
F \times \left(\left\lfloor \frac{2P + H - [D(K-1)+1]}{S} \right\rfloor + 1 \right) \times \left(\left\lfloor \frac{2P + W - [D(K-1)+1]}{S} \right\rfloor + 1 \right)
$$

3. Let's consider an input $x[n] \in \mathbb{R}^5$, with $1 \leq n \leq 7$, e.g. it is a length 7 sequence with 5 channels. We consider the convolutional layer f_W with one filter, with kernel size 3, stride of 2, no dilation, and no padding. The only parameters of the convolutional layer is the weight $W, W \in \mathbb{R}^{1\times5\times3}$ and there is no bias and no non-linearity.

   (a) What is the dimension of the output $f_W(x)$? Provide an expression for the value of elements of the convolutional layer output $f_W(x)$. Example answer format here and in the following sub-problems: $f_W(x) \in \mathbb{R}^{42\times42\times42}$, $f_W(x)[i,j,k] = 42$.

$$
f_W(x) \in \mathbb{R}^2
$$

$$
f_W(x)[r] = \sum_{k=1}^{5} \sum_{k=1}^{3} x[i + 2(r-1)][k] W_{k,i}
$$

   (b) What is the dimension of $\frac{\partial f_W(x)}{\partial W}$? What are its values?

$$
\frac{\partial f_W(x)}{\partial W} \in \mathbb{R}^{5\times(1\times5\times3)}
$$

$$
\frac{\partial f_W(x)}{\partial W}[r,c,i,k] = x[i + 2(r-1)][k]
$$

(c) What is the dimension of $\frac{\partial f_W(x)}{\partial x}$? What are its values?

$$\frac{\partial f_W(x)}{\partial x} \in \mathbb{R}^{2\times(2\times7)}$$

$$\frac{\partial f_W(x)}{\partial x}[r,k,i] = \begin{cases} W_{1,k,i-2(r-1)} & \text{if } i-2(r-1) \in [1,3] \\ 0 & \text{otherwise} \end{cases}$$

(d) Now, suppose you are given the gradient of the loss ℓ with respect to the output of the convolutional layer $f_W(x)$, i.e. $\frac{\partial \ell}{\partial f_W(x)}$. What is the dimension of $\frac{\partial \ell}{\partial W}$? Provide its expression. Explain the similarities and differences of this and expression in (a).

$$\frac{\partial \ell}{\partial W} \in \mathbb{R}^{a\times b\times c}$$

$$\left(\frac{\partial \ell}{\partial W}\right)[1,k,i] = \sum_{r=1}^{23232} \left(\frac{\partial \ell}{\partial f_W(x)}\right)[r]x[i+2(r-1),k]$$

The difference is that this is a dilated convolution. Both the backward and forward pass of the conv. layer apply a convolution but the stride dilates in the backward pass.

4. Show

Problem 1.2: Recurrent Neural Networks

$\sigma(z) = \frac{1}{1+\exp(-z)}$.

1. If you want

2. Now

Problem 1.3: Debugging Loss Curves

1. Why is softmax actually softargmax?