50.021 Artificial Intelligence

Theory Homework 4

Due: 3rd July, Monday, 4PM before class starts

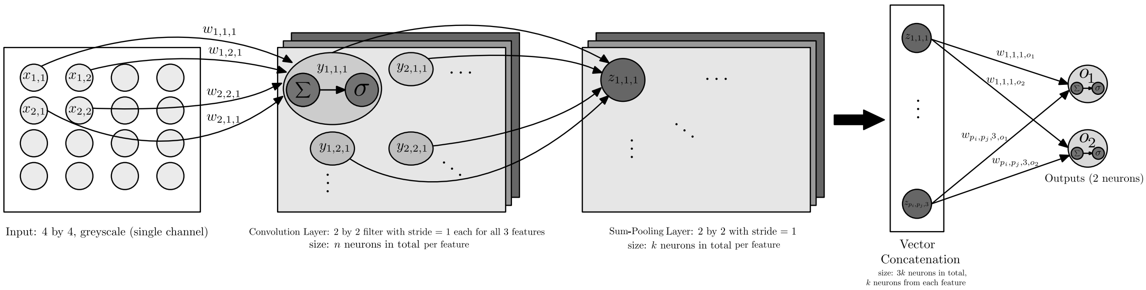

Figure 1: Mini CNN

[Q1]. The small-scaled version of a convolutional neural network shown in figure 1 is consisted of:

1. Input layer: 4 by 4 greyscale (one channel), labeled as $x_{i,j}$ where $i = 1, 2, 3, 4$ and $j = 1, 2, 3, 4$. There are in total 16 pixels of input.

2. Convolution layer : 2 by 2 filter with stride 1 is applied across the 4 by 4 greyscale input. There are in total 3 features in the convolution layer with the same filter size. The neurons in the convolution layer is labeled as $y_{c_i,c_j,k}$, where $c_i = 1, ..., c_w$ (width index), and $c_j = 1, ..., c_h$ (height index) and $k = 1, 2, 3$ (channel index). There are four distinct weights in **each** of the features, they are labelled as $w_{1,1,k}, w_{1,2,k}, w_{2,1,k}$, and $w_{2,2,k}$, where $k = 1, 2, 3$ (channel index).

   Note: in total we only train 12 weights in the entire convolution layer.

3. Sum-pooling layer: apply 2 by 2 sum pooling to the convolution layer. The neurons in the sum-pooling layer is labeled as $z_{p_i,p_j,k}$, where $p_i = 1, ..., p_w$ (width index), and $p_j = 1, ..., p_h$ (height index), and $k = 1, 2, 3$ (channel index).

4. Fully connected layer: concatenate (stack) all neurons in the sum-pooling layer into one long vector. Each neuron in the fully connected layer has two distinct weights, connecting them to each of the two output neurons. The neurons in the fully connected layer is the same neurons as the sum-pooling layer, only being reshaped.

5. Output layer: made up of two neurons, o_1 and o_2.

No zero padding is added in any of the layers, and σ indicates the sigmoid function.

Part 1. Forward Pass

Answer the following questions,

1. Determine the values of c_w, c_h, p_w, and p_h. They are the width and height of the convolution and pooling layers.

   Solution:

   The width and height of each feature in the convolutional layer is $(4 - 2)/1 + 1 = 3$, hence $c_w = c_h = 3$.
   The width and height of each feature in the sum-pooling layer is $(3 - 2)/1 + 1 = 2$, hence $p_w = p_h = 2$.

2. Express $y_{c_i,c_j,k}$ in terms of inputs $x_{i,j}$ and the four filter weights: $w_{1,1,k}, w_{1,2,k}, w_{2,1,k}\ w_{2,2,k}$.

   Solution:

   $y_{c_i,c_j,k} = \sigma(x_{c_i,c_j} \cdot w_{1,1,k} + x_{c_i,c_{j+1}} \cdot w_{1,2,k} + x_{c_{i+1},c_j} \cdot w_{2,1,k} + x_{c_{i+1},c_{j+1}} \cdot w_{2,2,k})$

3. Express $z_{p_i,p_j,k}$ in terms of $y_{c_i,c_j,k}$.

   Solution:

   $z_{p_i,p_j,k} = y_{p_i,p_j,k} + y_{p_i,p_{j+1},k} + y_{p_{i+1},p_j,k} + y_{p_{i+1},p_{j+1},k}$

4. Express o_1 ando_2 in terms of $z_{p_i,p_j,k}$.

   Solution:

   $o_1 = \sigma(\sum_{k=1}^{3}\sum_{p_i=1}^{2}\sum_{p_j=1}^{2}(z_{p_i,p_j,k} \cdot w_{p_i,p_j,k,o_1}))$
   $o_2 = \sigma(\sum_{k=1}^{3}\sum_{p_i=1}^{2}\sum_{p_j=1}^{2}(z_{p_i,p_j,k} \cdot w_{p_i,p_j,k,o_2}))$

Part 2. Backpropagation

Given the following error function for one sample,

$$E = \frac{1}{2}\left((o_1 - y_1)^2 + (o_2 - y_2)^2\right)$$

where y_1 and y_2 are the true labels. Answer the following questions,

1. Derive the expression for the weight error $\frac{\partial E}{\partial w_{p_i,p_j,k,o_1}}$ and $\frac{\partial E}{\partial w_{p_i,p_j,k,o_2}}$

   Solution:

$$\begin{aligned}\frac{\partial E}{\partial w_{p_i,p_j,k,o_1}} &= \frac{\partial E}{\partial o_1} \cdot \frac{\partial o_1}{\partial w_{p_i,p_j,k,o_1}} \\ &= (o_1 - y_1) \cdot o_1(1 - o_1) z_{p_i,p_j,k}\end{aligned}$$

$$\begin{aligned}\frac{\partial E}{\partial w_{p_i,p_j,k,o_2}} &= \frac{\partial E}{\partial o_2} \cdot \frac{\partial o_2}{\partial w_{p_i,p_j,k,o_2}} \\ &= (o_2 - y_2) \cdot o_2(1 - o_2) z_{p_i,p_j,k}\end{aligned}$$

2. Derive the expression for the pooling layer neuron error $\frac{\partial E}{\partial z_{p_i,p_j,k}}$

   Solution:

$$\begin{aligned}\frac{\partial E}{\partial z_{p_i,p_j,k}} &= \frac{\partial E}{\partial o_1} \cdot \frac{\partial o_1}{\partial z_{p_i,p_j,k}} + \frac{\partial E}{\partial o_2} \cdot \frac{\partial o_2}{\partial z_{p_i,p_j,k}} \\ &= (o_1 - y_1) \cdot o_1(1 - o_1) w_{p_i,p_j,k,o_1} + (o_2 - y_2) \cdot o_2(1 - o_2) w_{p_i,p_j,k,o_2}\end{aligned}$$

3. Since there are no parameters on the pooling layer, we can simply propagate back the error in the sum-pooling layer to the convolution layer,

$$\frac{\partial E}{\partial y_{c_i,c_j,k}} = \frac{\partial E}{\partial z_{p_i,p_j,k}} \cdot \frac{\partial z_{p_i,p_j,k}}{\partial y_{c_i,c_j,k}}$$

So for any $y_{c_i,c_j,k}$ in the sum-pooling case,

$$\frac{\partial z_{p_i,p_j,k}}{\partial y_{c_i,c_j,k}} = 1$$

It means that the error from the sum-pooling neuron is added to the error of the four neurons in convolution layers of which that neuron take the value sum from, for example,

$$\frac{\partial E}{\partial y_{c_i,c_j,k}}, \frac{\partial E}{\partial y_{c'_i,c'_j,k}}, \frac{\partial E}{\partial y_{c''_i,c''_j,k}}, \frac{\partial E}{\partial y_{c'''_i,c''_j,k}} \mathrel{+}= \frac{\partial E}{\partial z_{p_i,p_j,k}},$$

where $c_i, c'_i, c''_i, c'''_i, c_j, c'_j, c''_j, c'''_j$ are the width and height indexes of the four neurons in the convolution layer of which the sum value is taken from, producing the neuron $z_{p_i,p_j,k}$ in the sum-pooling layer. The symbol $+=$ is used indicating that the error in neurons in the convolution layer is composed from the addition of the errors of the sum-pooling layer that involves them, because there exists overlaps since the stride is 1.

Explicitly rewrite the sets of equation above in terms of all indexes of the convolutional layer $c_i = 1, ..., c_w$ (width), $c_j = 1, ..., c_h$ (height) and sum-pooling layer $p_i = 1, ..., p_w$ (width), $p_j = 1, ..., p_h$ (height). For example, the first set is written for you,

$$\frac{\partial E}{\partial y_{1,1,k}}, \frac{\partial E}{\partial y_{1,2,k}}, \frac{\partial E}{\partial y_{2,1,k}}, \frac{\partial E}{\partial y_{2,2,k}} \mathrel{+}= \frac{\partial E}{\partial z_{1,1,k}}$$

Hint: you should have $p_w \times p_h$ sets of equation in total
Solution:

$$\frac{\partial E}{\partial y_{1,2,k}}, \frac{\partial E}{\partial y_{1,3,k}}, \frac{\partial E}{\partial y_{2,2,k}}, \frac{\partial E}{\partial y_{2,3,k}} \mathrel{+}= \frac{\partial E}{\partial z_{1,2,k}}$$
$$\frac{\partial E}{\partial y_{2,1,k}}, \frac{\partial E}{\partial y_{2,2,k}}, \frac{\partial E}{\partial y_{3,1,k}}, \frac{\partial E}{\partial y_{3,2,k}} \mathrel{+}= \frac{\partial E}{\partial z_{2,1,k}}$$
$$\frac{\partial E}{\partial y_{2,2,k}}, \frac{\partial E}{\partial y_{2,3,k}}, \frac{\partial E}{\partial y_{3,2,k}}, \frac{\partial E}{\partial y_{3,3,k}} \mathrel{+}= \frac{\partial E}{\partial z_{2,2,k}}$$

4. Finally, find the expression for the filter weight error in the convolution layer, $\frac{\partial E}{\partial w_{f_i,f_j,k}}$, where $f_i = 1, 2$ and $f_j = 1, 2$.
Solution:

$$\frac{\partial E}{\partial w_{f_i,f_j,k}} = \sum_{c_i=1}^{3} \sum_{c_j=1}^{3} \frac{\partial E}{\partial y_{c_i,c_j,k}} \cdot \frac{\partial y_{c_i,c_j,k}}{\partial w_{f_i,f_j,k}}$$

$$\frac{\partial y_{c_i,c_j,k}}{\partial w_{f_i,f_j,k}} = y_{c_i,c_j,k}(1 - y_{c_i,c_j,k}) x_{c_i+1[f_i==2],c_j+1[f_j==2]}$$

Hence from answers in the previous section we have,

$$
\begin{aligned}
\frac{\partial E}{\partial w_{f_i,f_j,k}} =& y_{1,1,k}(1-y_{1,1,k}) \cdot x_{1+1[f_i==2],1+1[f_j==2]} \cdot \frac{\partial E}{\partial z_{1,1,k}} + \\
& y_{1,2,k}(1-y_{1,2,k}) \cdot x_{1+1[f_i==2],2+1[f_j==2]} \cdot \left(\frac{\partial E}{\partial z_{1,1,k}} + \frac{1}{4}\frac{\partial E}{\partial z_{1,2,k}} \right) + \\
& y_{1,3,k}(1-y_{1,3,k}) \cdot x_{1+1[f_i==2],3+1[f_j==2]} \cdot \frac{\partial E}{\partial z_{1,2,k}} + \\
& y_{2,1,k}(1-y_{2,1,k}) \cdot x_{2+1[f_i==2],1+1[f_j==2]} \cdot \left(\frac{\partial E}{\partial z_{1,1,k}} + \frac{1}{4}\frac{\partial E}{\partial z_{2,2,k}} \right) + \\
& y_{2,2,k}(1-y_{2,2,k}) \cdot x_{2+1[f_i==2],2+1[f_j==2]} \cdot \left(\sum_{p_i=1}^{2} \sum_{p_j=1}^{2} \frac{\partial E}{\partial z_{p_i,p_j,k}} \right) + \\
& y_{2,3,k}(1-y_{2,3,k}) \cdot x_{2+1[f_i==2],3+1[f_j==2]} \cdot \left(\frac{\partial E}{\partial z_{1,2,k}} + \frac{1}{4}\frac{\partial E}{\partial z_{2,2,k}} \right) + \\
& y_{3,1,k}(1-y_{3,1,k}) \cdot x_{3+1[f_i==2],1+1[f_j==2]} \cdot \frac{\partial E}{\partial z_{2,1,k}} + \\
& y_{3,2,k}(1-y_{3,2,k}) \cdot x_{3+1[f_i==2],2+1[f_j==2]} \cdot \left(\frac{\partial E}{\partial z_{2,1,k}} + \frac{1}{4}\frac{\partial E}{\partial z_{2,2,k}} \right) + \\
& y_{3,3,k}(1-y_{3,3,k}) \cdot x_{3+1[f_i==2],3+1[f_j==2]} \cdot \frac{\partial E}{\partial z_{2,2,k}}
\end{aligned}
$$

5. Now consider if max-pooling is used instead of sum-pooling. The window size is still the same, 2 by 2 with stride of 1. What is the new neuron error $\frac{\partial E}{\partial y_{c_i,c_j,k}}$ in the convolution layer?

   Note: you can use an indicator function. An indicator function looks something like this: $1[\textit{some-logical-expression}]$*, and will take the value 1 if the expression is true, or zero otherwise. An example of it is* $1[z < 9]$.

   Solution:

   If max-pooling is used, the error is only propagated to the neuron in the convolution layer that produces the max value. So in the 2 by 2 window, only one neuron in the convolution layer gets the backpropagated error from the neuron in the max-pooling layer. The rest of the neurons in the convolution gets zero error.

   The derivative of the max function is,

$$\frac{\partial z_{p_i,p_j,k}}{\partial y_{c_i,c_j,k}} = 1 \text{ if } y_{c_i,c_j,k} == \max_{\text{2 by 2}}$$

   where $\max_{\text{2 by 2}}$ is the maximum value in the 2 by 2 max-pooling window,

$$\max_{\text{2 by 2}} = \max\{y_{p_i,p_j,k} + y_{p_i,p_{j+1},k} + y_{p_{i+1},p_j,k} + y_{p_{i+1},p_{j+1},k}\}$$

   Thus,

$$\frac{\partial E}{\partial y_{c_i,c_j,k}} = \frac{\partial E}{\partial z_{p_i,p_j,k}} 1\left[y_{c_i,c_j,k} == \max_{\text{2 by 2}}\right]$$