MFDNN HW1

Shin mingyu

April 1, 2024

Problem 1 : *Finite difference with convolution.*

$$w = \begin{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \end{bmatrix}$$

is the desired filter.

Problem 2 : *Average polling as convolution.*

$$w_c = \begin{bmatrix} \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix}, \cdots, \underbrace{\frac{1}{k^2}\begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix}}_{\text{c-th component}}, \cdots, \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \end{bmatrix}$$

$c = 1, \dots, C$, is the desired filter. $X \mapsto Y$ can be represent as a convolution with the $w \in \mathbb{R}^{C \times C \times k \times k}$, stride $= k$.

Problem 3 : *RGB to greyscale mapping with 1×1 convolution.*

$$w = \begin{bmatrix} \begin{bmatrix} 0.299 \end{bmatrix}, \begin{bmatrix} 0.587 \end{bmatrix}, \begin{bmatrix} 0.114 \end{bmatrix} \end{bmatrix}$$

is the desired filter.

Problem 4

For any $X \in \mathbb{R}^{a \times b}$, $\underset{x_{ij} \in X}{argmax}\ x_{ij} = \underset{x_{ij} \in X}{argmax}\ \sigma(x_{ij})$, hence,

$$\Rightarrow \sigma(\underset{x_{ij} \in X}{max}\ x_{ij}) = \underset{x_{ij} \in X}{max}\ \sigma(X_{ij})$$

Finally, considering each sub-matrix of $X \in \mathbb{R}^{m \times n}$ which the max pool operation is applied, yields the result, $\sigma(\rho(X)) = \rho(\sigma(X))$.

Problem 5 : *Non-CE loss function.*

The elapsed time and accuracy is almost the same for both CE Loss and Square Loss.

```
CE Loss Accuracy: 1897/1991(95.28)%
```

(a) CE Loss Accuracy

```
Square Loss Accuracy: 1894/1991(95.13)%
```

(b) SQUARE Loss Accuracy

Problem 6 : *Backporp for MLP.*

(a) Clearly,

$$\frac{\partial y_L}{\partial b_L} = 1, \quad \frac{\partial y_L}{\partial y_{L-1}} = \frac{\partial (A_L y_{L-1} + b_L)}{\partial y_{L-1}} = A_L,$$

since $A_L \in \mathbb{R}^{1 \times n_{L-1}}$, that is a vector. For $\ell = 1, \dots, L-1$,

$$(\frac{\partial y_\ell}{\partial b_\ell})_{ij} = \frac{\partial (y_\ell)_i}{\partial (b_\ell)_j} = \frac{\partial (\sigma(A_\ell y_{\ell-1} + b_\ell))_i}{\partial (b_\ell)_j} = \begin{cases} \sigma'(A_\ell y_{\ell-1} + b_\ell) & i = j \\ 0 & i \neq j \end{cases}$$

$$\Rightarrow \frac{\partial y_\ell}{\partial b_\ell} = \operatorname{diag}(\sigma'(A_\ell y_{\ell-1} + b_\ell)).$$

Similarly, for $\ell = 2, \dots, L-1$,

$$(\frac{\partial y_\ell}{\partial y_{\ell-1}})_{ij} = \frac{\partial (y_\ell)_i}{\partial (y_{\ell-1})_j} = \frac{\partial \sigma((A_\ell)_{i,:} y_{\ell-1} + (b_\ell)_i)}{\partial (y_{\ell-1})_j} = \sigma'((A_\ell)_{i,:} y_{\ell-1} + (b_\ell)_i)(A_\ell)_{ij}$$

$$\Rightarrow (\frac{\partial y_\ell}{\partial y_{\ell-1}})_{i,:} = \sigma'((A_\ell)_{i,:} y_{\ell-1} + (b_\ell)_i)(A_\ell)_{i,:} A_\ell$$

$$\Rightarrow \frac{\partial y_\ell}{\partial y_{\ell-1}} = \operatorname{diag}(\sigma'(A_\ell y_{\ell-1} + b_\ell)) A_\ell.$$

where $A_{i,:}$ is the i-th row of A.

(b) Note that for $i = 1, \dots, n_\ell$ and $j = 1, \dots, n_{\ell-1}$,

$$(\frac{\partial y_L}{\partial A_\ell})_{ij} = \frac{\partial y_L}{\partial (A_\ell)_{ij}}.$$

Then,

$$(\frac{\partial y_L}{\partial A_L})_{1j} = \frac{\partial y_L}{\partial (A_L)_{1j}} = \frac{\partial (A_L y_{L-1} + b_L)}{\partial (A_L)_{1j}} = (y_{L-1})_j$$

$$\Rightarrow \frac{\partial y_L}{\partial A_L} = y_{L-1}^\intercal$$

Similarly, note that for $\ell = 1, \dots, L-1$,

$$(\frac{\partial y_L}{\partial A_\ell})_{ij} = \frac{\partial y_L}{\partial (A_\ell)_{ij}} = \frac{\partial y_L}{\partial y_\ell} \frac{\partial y_\ell}{\partial (A_\ell)_{ij}}$$

by chain rule. Since

$$\frac{\partial y_\ell}{\partial (A_\ell)_{ij}} = \frac{\partial \sigma(A_\ell y_{\ell-1} + b_\ell)}{\partial (A_\ell)_{ij}} = \begin{bmatrix} \frac{(A_\ell)_{1,:}y_{\ell-1}+(b_\ell)_1}{\partial (A_\ell)_{ij}} \\ \vdots \\ \frac{(A_\ell)_{n_\ell,:}y_{\ell-1}+(b_\ell)_{n_\ell}}{\partial (A_\ell)_{ij}} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ \sigma'((A_\ell)_{i,:}y_{\ell-1} + (b_\ell)_i)(y_{\ell-1})_j \\ \vdots \\ 0 \end{bmatrix},$$

we have

$$(\frac{\partial y_L}{\partial A_\ell})_{ij} = \frac{\partial y_L}{\partial y_\ell} \begin{bmatrix} 0 \\ \vdots \\ \sigma'((A_\ell)_{i,:}y_{\ell-1} + (b_\ell)_i)(y_{\ell-1})_j \\ \vdots \\ 0 \end{bmatrix} = (\frac{\partial y_L}{\partial y_\ell})_i \sigma'((A_\ell)_{i,:}y_{\ell-1} + (b_\ell)_i)(y_{\ell-1})_j,$$

hence,

$$\frac{\partial y_L}{\partial A_\ell} = (\sigma'(A_\ell y_{\ell-1} + b_\ell) \odot (\frac{\partial y_L}{\partial y_\ell}))^\intercal y_{\ell-1}^\intercal = \mathrm{diag}(\sigma'(A_\ell y_{\ell-1} + b_\ell))(\frac{\partial y_L}{\partial y_\ell})^\intercal y_{\ell-1}^\intercal,$$

which is the answer.

Problem 7

The number of trainable parameters in original C3 layer is

$$\underbrace{(5\times 5)\times 3\times 6+6}_{\text{6 conv module taking 3 channels}} + \underbrace{(5\times 5)\times 4\times 9+9}_{\text{9 conv module taking 4 channels}} + \underbrace{(5\times 5)\times 1\times 6+1}_{\text{1 conv module taking 6 channels}} = 1516,$$

and the number of trainable parameters in regular C3 layer is

$$\underbrace{(5\times 5)\times 6\times 16+16}_{\text{16 conv module taking 6 channels}} = 2416.$$

This result is actually same with the result of the starter code.

```
Total number of trainable parameters: 60806
Specifically, the number of trainable parameters for C3 convolution layer: 1516
```

(a) Original C3 layer

```
Total number of trainable parameters: 61706
Specifically, the number of trainable parameters for C3 convolution layer: 2416
```

(b) Regular C3 layer