Submission for Deep Learning Exercise 2

Team: dl2022-ryd
Students: Yumna Ali, Rean Fernandes

November 29, 2022

1 Pen and paper task : Convolution in 1-D

1.1 Forward Pass

1.1.1 Stride 0

$$\hat{y}_1 = w \cdot x + b \tag{1a}$$

$$\hat{y}_1 = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \cdot \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix} + b = \quad w_1 \cdot x_1 + w_2 \cdot x_2 + w_3 \cdot x_3 + b \tag{1b}$$

1.1.2 Stride 1

$$\hat{y}_2 = w \cdot x + b \tag{2a}$$

$$\hat{y}_2 = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \cdot \begin{bmatrix} x_2 & x_3 & x_4 \end{bmatrix} + b = \quad w_1 \cdot x_2 + w_2 \cdot x_3 + w_3 \cdot x_4 + b \tag{2b}$$

1.1.3 Loss

$$\mathscr{L}(\hat{y}, y) = \sum_{i=1}^{2} \frac{1}{2} ||y_i - \hat{y}_i||_2^2 \tag{3}$$

1.2 Backward pass

$$\frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \hat{\mathbf{y}}} = \begin{bmatrix} \hat{y_1} - y_1 \\ \hat{y}_2 - y_2 \end{bmatrix} \tag{4a}$$

$$\frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \mathbf{w}} = \frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \hat{\mathbf{y}}} \cdot \frac{\partial \hat{\mathbf{y}}}{\partial \mathbf{w}} \tag{4b}$$

$$\frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \mathbf{w}} = \begin{bmatrix} \hat{y_1} - y_1 \\ \hat{y}_2 - y_2 \end{bmatrix} \cdot \begin{bmatrix} x_1 & x_2 & x_3 \\ x_2 & x_3 & x_4 \end{bmatrix} = \quad \begin{bmatrix} (\hat{y_1} - y_1) \cdot x_1 + (\hat{y}_2 - y_2) \cdot x_2 \\ (\hat{y_1} - y_1) \cdot x_2 + (\hat{y}_2 - y_2) \cdot x_3 \\ (\hat{y_1} - y_1) \cdot x_3 + (\hat{y}_2 - y_2) \cdot x_4 \end{bmatrix} \tag{4c}$$

$$\frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \mathbf{b}} = \frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \hat{\mathbf{y}}} \cdot \frac{\partial \hat{\mathbf{y}}}{\partial \mathbf{b}} \tag{4d}$$

$$\frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \mathbf{b}} = \begin{bmatrix} \hat{y_1} - y_1 \\ \hat{y}_2 - y_2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \hat{y_1} - y_1 \\ \hat{y}_2 - y_2 \end{bmatrix} \tag{4e}$$

The bias matrix has been represented as a 2 by 2 matrix to show that we have two biases $b_1 = b_2 = 1$, the broadcasting here has just been represented in this form to make it mathematically sensible.

1.2.1 Weight update

$$w^{(1)} = w - \alpha \cdot \frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \mathbf{w}} = \begin{bmatrix} w_1 - \alpha \cdot (\hat{y_1} - y_1) \cdot x_1 + (\hat{y}_2 - y_2) \cdot x_2 \\ w_2 - \alpha \cdot (\hat{y_1} - y_1) \cdot x_2 + (\hat{y}_2 - y_2) \cdot x_3 \\ w_3 - \alpha \cdot (\hat{y_1} - y_1) \cdot x_3 + (\hat{y}_2 - y_2) \cdot x_4 \end{bmatrix} \tag{5a}$$

$$b^{(1)} = b - \alpha \cdot \frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \mathbf{b}} = \begin{bmatrix} b_1 - \alpha \cdot (\hat{y_1} - y_1) \\ b_2 - \alpha \cdot (\hat{y_2} - y_2) \end{bmatrix} \tag{5b}$$

2 Performing the convolution with backward pass for given values

$$x = \begin{bmatrix} 1 & 2 & 3 & 4 \end{bmatrix}^T$$
$$w = \begin{bmatrix} 2 & 1 & 3 \end{bmatrix}^T$$
$$b = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$$

2.1 Forward Pass

2.1.1 Stride 0

$$\hat{y}_1 = \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix} \cdot \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} + b = \quad 2 \cdot 1 + 1 \cdot 2 + 3 \cdot 3 + 1 = 14 \tag{6a}$$

2.1.2 Stride 1

$$\hat{y}_1 = \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix} \cdot \begin{bmatrix} 2 & 3 & 4 \end{bmatrix} + b = \quad 2 \cdot 2 + 1 \cdot 3 + 3 \cdot 4 + 1 = 20 \tag{7a}$$

2.1.3 Prediction and Loss

$$\hat{\mathbf{y}} = \begin{bmatrix} 14 \\ 20 \end{bmatrix} \tag{8}$$

$$\mathscr{L}(\hat{y}, y) = \sum_{i=1}^{2} \frac{1}{2} ||y_i - \hat{y}_i||_2^2 = \frac{1}{2} \cdot (2 - 14)^2 + \frac{1}{2}(4 - 20)^2 = 200 \tag{9a}$$

2.2 Backward Pass

$$\frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \hat{\mathbf{y}}} = \begin{bmatrix} \hat{y_1} - y_1 \\ \hat{y}_2 - y_2 \end{bmatrix} = \begin{bmatrix} 14 - 2 \\ 20 - 4 \end{bmatrix} = \begin{bmatrix} 12 \\ 16 \end{bmatrix} \tag{10a}$$

$$\frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \mathbf{w}} = \begin{bmatrix} \hat{y_1} - y_1 \\ \hat{y}_2 - y_2 \end{bmatrix} \cdot \begin{bmatrix} x_1 & x_2 & x_3 \\ x_2 & x_3 & x_4 \end{bmatrix} = \quad \begin{bmatrix} 12 \cdot 1 + 16 \cdot 2 \\ 12 \cdot 2 + 16 \cdot 3 \\ 12 \cdot 3 + 16 \cdot 4 \end{bmatrix} = \quad \begin{bmatrix} 44 \\ 72 \\ 100 \end{bmatrix} \tag{10b}$$

$$\frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \mathbf{b}} = \begin{bmatrix} \hat{y_1} - y_1 \\ \hat{y}_2 - y_2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 12 \\ 16 \end{bmatrix} \tag{10c}$$

2.3 Weight update

$$w^{(1)} = w - \alpha \cdot \frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \mathbf{w}} = \quad \begin{bmatrix} 2 - 0.01 \cdot 44 \\ 1 - 0.01 \cdot 72 \\ 3 - 0.01 \cdot 100 \end{bmatrix} = \quad \begin{bmatrix} 1.56 \\ 0.28 \\ 2 \end{bmatrix} \tag{11a}$$

$$b^{(1)} = b - \alpha \cdot \frac{\partial \mathscr{L}(\hat{y}, y)}{\partial \mathbf{b}} = \quad \begin{bmatrix} 1 - 0.01 \cdot 12 \\ 1 - 0.01 \cdot 16 \end{bmatrix} = \quad \begin{bmatrix} 0.88 \\ 0.84 \end{bmatrix} \tag{11b}$$

2.4 Forward pass with updated weights and biases

New weights :

$$w = \begin{bmatrix} 1.56 & 0.28 & 2 \end{bmatrix}^T$$

New biases:

$$b = \begin{bmatrix} 0.88 & 0.84 \end{bmatrix}^T$$

2.5 Forward Pass

2.5.1 Stride 0

$$\hat{y}_1 = \begin{bmatrix} 1.56 \\ 0.28 \\ 2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} + b_1 = \quad 1.56 \cdot 1 + 0.28 \cdot 2 + 2 \cdot 3 + 0.88 = 9 \tag{12a}$$

2.5.2 Stride 1

$$\hat{y}_1 = \begin{bmatrix} 1.56 \\ 0.28 \\ 2 \end{bmatrix} \cdot \begin{bmatrix} 2 & 3 & 4 \end{bmatrix} + b_2 = \quad 1.56 \cdot 2 + 0.28 \cdot 3 + 2 \cdot 4 + 0.84 = 12.8 \tag{13a}$$

2.5.3 Prediction and Loss

$$\hat{\mathbf{y}} = \begin{bmatrix} 9 \\ 12.8 \end{bmatrix} \tag{14}$$

$$\mathscr{L}(\hat{y}, y) = \sum_{i=1}^{2} \frac{1}{2} ||y_i - \hat{y}_i||_2^2 = \frac{1}{2} \cdot (2-9)^2 + \frac{1}{2}(4-12.8)^2 = 63.22 \tag{15a}$$

3 Experiments : Equivariance

3.1

Accuracies of both MLP and convolution models on original validation data are better than on the shifted validation set. Also the convolution model outperforms the MLP in both cases with and without shifting. The explanation for this is that CNN work well with data that has spatial relationship like images

3.2

We can use more regularization methods that make the model more robust to different types of noise like shifting by adding augmentation of shifted data during training. The model can then learn that the shifted and original version of the image belong to the same class.

3.3

Convolution layers are translation equivariant, not translation invariant. Therefore the whole network is not translation invariant. The model was trained on unshifted dataset but to get the same accuracy of the unshifted dataset the model should be trained on the shifted one. That's how we can apply the property of equivalence of translation.