Sharif University of Technology
Electrical Engineering Department

Machine Learning
HW 3

Amir Hossein Yari
99102507

June 8, 2023

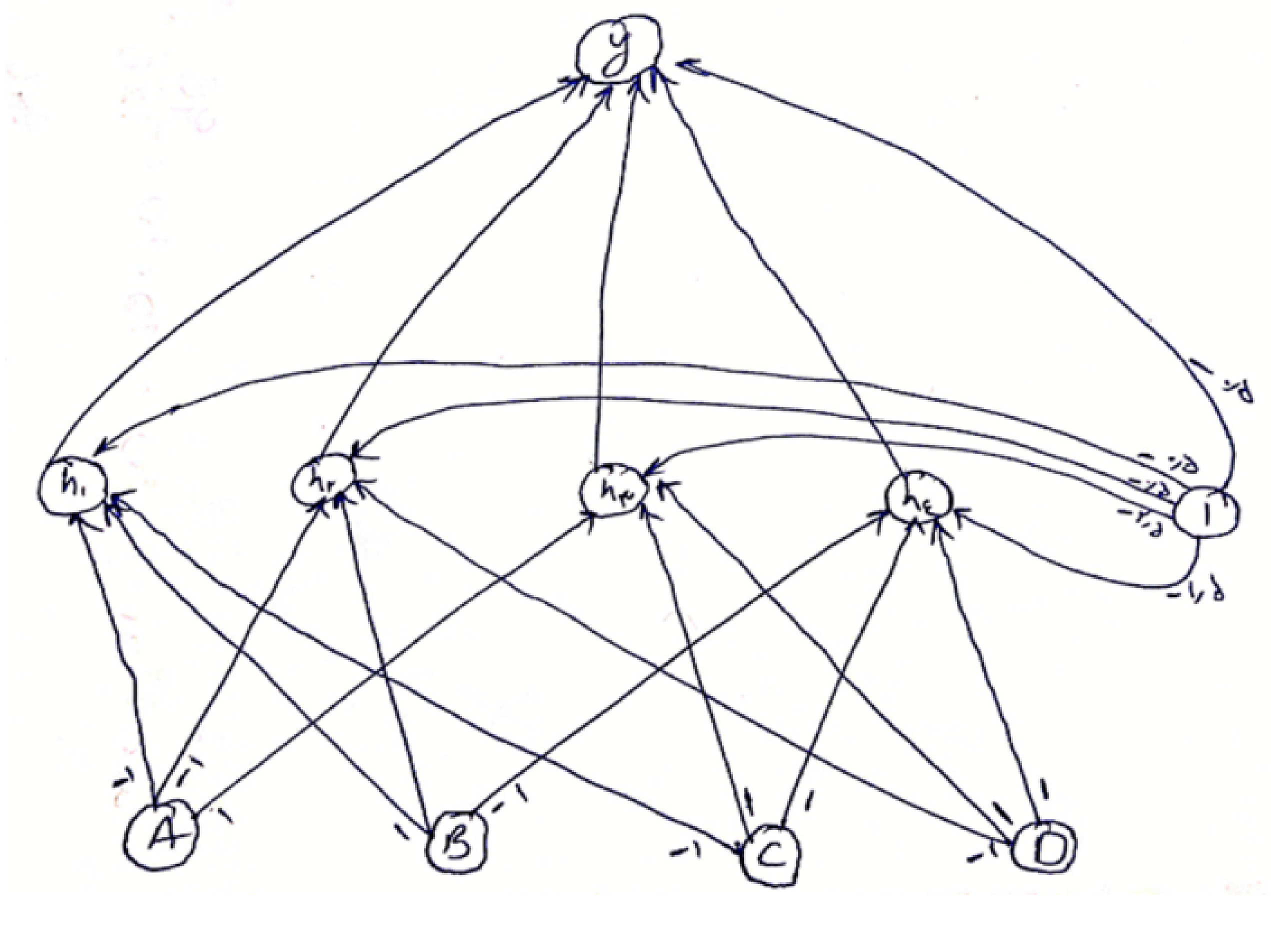
Contents

1. Design Simple Neural Network

$$(A \vee \bar{B}) \oplus (\bar{C} \vee \bar{D}) = [\overline{(A \vee \bar{B}) \wedge (\bar{C} \vee \bar{D})}] \wedge [(A \vee \bar{B}) \vee (\bar{C} \vee \bar{D})]$$
$$= [(\bar{A} \wedge B) \vee (C \wedge D)] \wedge [(A \vee \bar{B}) \vee (\bar{C} \vee \bar{D})]$$
$$= (\bar{A} \wedge B \wedge \bar{C}) \vee (\bar{A} \wedge B \wedge \bar{D}) \vee (A \wedge C \wedge D) \vee (\bar{B} \wedge C \wedge D)$$

$$h1 = (\bar{A} \wedge B \wedge \bar{C}) = \begin{bmatrix}-0.5 & -1 & 1 & -1 & 0\end{bmatrix} \begin{bmatrix}1 & A & B & C & D\end{bmatrix}^T$$
$$h2 = (\bar{A} \wedge B \wedge \bar{D}) = \begin{bmatrix}-0.5 & -1 & 1 & 0 & -1\end{bmatrix} \begin{bmatrix}1 & A & B & C & D\end{bmatrix}^T$$
$$h3 = (A \wedge C \wedge D) = \begin{bmatrix}-2.5 & 1 & 0 & 1 & 1\end{bmatrix} \begin{bmatrix}1 & A & B & C & D\end{bmatrix}^T$$
$$h4 = (\bar{B} \wedge C \wedge D) = \begin{bmatrix}-1.5 & 0 & -1 & 1 & 1\end{bmatrix} \begin{bmatrix}1 & A & B & C & D\end{bmatrix}^T$$

$$y = h_1 \vee h_2 \vee h_3 \vee h_4 = \begin{bmatrix}-0.5 & 1 & 1 & 1 & 1\end{bmatrix} \begin{bmatrix}1 & h_1 & h_2 & h_3 & h_4\end{bmatrix}^T$$



2. Vector Derivative

$$J_{f_1}(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}) = \begin{bmatrix} 0 & cos(\pi x_2) \\ x_2^2\, e^{x_1 - 1} & 2\, e^{x_1 - 1}\, x_2 \\ x_2 & x_1 \end{bmatrix}$$

$$J_{f_2}(\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}) = \begin{bmatrix} 1 & 1 & 1 \\ 2x_1 & 2x_2 & 2x_3 \end{bmatrix}$$

$$J_{f_1}(\begin{bmatrix} 1 \\ 2 \end{bmatrix}) \times e_1 = \begin{bmatrix} 0 & 1 \\ 4 & 4 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 4 \\ 2 \end{bmatrix} = v_1$$

$$J_{f_2}(\begin{bmatrix} 1 \\ 2 \end{bmatrix}) \times e_2 = \begin{bmatrix} 0 & 1 \\ 4 & 4 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 1 \end{bmatrix} = v_2$$

$$f_2(f_1(\mathbf{x})) = f_2(\begin{bmatrix} 0 \\ 4 \\ 2 \end{bmatrix})$$

$$J_{f_2}(\begin{bmatrix} 0 \\ 4 \\ 2 \end{bmatrix}) \times v_1 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 8 & 4 \end{bmatrix} \begin{bmatrix} 0 \\ 4 \\ 2 \end{bmatrix} = \begin{bmatrix} 6 \\ 40 \end{bmatrix} = v_1^{new}$$

$$J_{f_2}(\begin{bmatrix} 0 \\ 4 \\ 2 \end{bmatrix}) \times v_2 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 8 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 4 \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \\ 36 \end{bmatrix} = v_2^{new}$$

$$\frac{\partial \mathbf{f}}{\partial \mathbf{x}} = \begin{bmatrix} 6 & 6 \\ 40 & 36 \end{bmatrix}$$

3. One Convolutional Layer

$$z_i = b + \sum_{j=1}^{d} k_j \, x_{i+j-1}$$

$$\frac{\partial L}{\partial k_j} = \frac{\partial L}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial k_j} = \sum_{i=1}^{m} \alpha_i \frac{\partial z_i}{\partial k_j} = \sum_{i=1}^{m} \alpha_i \, x_{i+j-1}$$

$$\frac{\partial L}{\partial \mathbf{b}} = \frac{\partial L}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial \mathbf{b}} = \sum_{i=1}^{m} \alpha_i \frac{\partial z_i}{\partial \mathbf{b}} = \sum_{i=1}^{m} \alpha_i$$

4. Backpropagation Algorithm

1. calculate $h1$, h_2, $\hat{y}$

$$h_1 = 0 \times 2.5 + 1 \times 1 + 1.5 \times 1 = 2.5 \Rightarrow \sigma(2.5) = 0.924$$
$$h_2 = 0 \times (-1.5) + 1 \times (-3) + 2 \times 1 = -1 \Rightarrow \sigma(-1) = 0.269$$
$$\hat{y} = 1 \times 0.924 + 0.5 \times 0.269 + 1 \times (-1) = 0.0585 \Rightarrow \sigma(0.0585) = 0.515$$

2. one step of backpropagation algorithm

$$\mathbf{z}_h = W_h \mathbf{x} + \mathbf{b}_h$$
$$z_o = W_o \sigma(\mathbf{z}_h) + b_o$$

Error of the output:

$$\mathcal{L} = \mathbb{H}(\sigma(z_o), y) = \mathbb{H}(0.515, 1) = 0.2885$$

Back Propagation:

$$\delta_o = \frac{\partial\, \mathbb{H}(\sigma(z_o), y)}{\partial\, \sigma(z_0)} = -\frac{1}{\sigma(z_0)}\, \sigma(z_0)(1 - \sigma(z_0)) = -0.4854$$

$$\mathbf{g}_{W_o} = \delta_o\, \sigma(\mathbf{z}_h) = -0.4854 \begin{bmatrix} 0.9241 & 0.2689 \end{bmatrix} = \begin{bmatrix} -0.4486 & -0.1305 \end{bmatrix}$$

$$g_{b_o} = \delta_o = -0.4854$$

$$\boldsymbol{\delta}_h = (W_o\, \delta_o\, \sigma'(\mathbf{z}_h))^T = (-0.4854 \begin{bmatrix} 1 & 0.5 \end{bmatrix} \begin{bmatrix} \sigma(2.5)(1 - \sigma(2.5)) & 0 \\ 0 & \sigma(-1)(1 - \sigma(-1)) \end{bmatrix})^T = \begin{bmatrix} -0.034 \\ -0.0477 \end{bmatrix}$$

$$\mathbf{g}_{W_h} = \boldsymbol{\delta}_h\, \mathbf{x}^T = \begin{bmatrix} -0.034 \\ -0.0477 \end{bmatrix} \begin{bmatrix} 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -0.034 \\ 0 & -0.0477 \end{bmatrix}$$

$$\mathbf{g}_{b_h} = \boldsymbol{\delta}_h = \begin{bmatrix} -0.034 \\ -0.0477 \end{bmatrix}$$

3. updated weights

$$W_h^{new} = W_h^{old} - \eta\, \mathbf{g}_{W_h} = \begin{bmatrix} 2.5 & 1 \\ -1.5 & -3 \end{bmatrix} - 0.1 \begin{bmatrix} 0 & -0.034 \\ 0 & -0.0477 \end{bmatrix} = \begin{bmatrix} 2.5 & 1.0034 \\ -1.5 & -2.99523 \end{bmatrix}$$

$$\mathbf{b}_h^{new} = \mathbf{b}_h^{old} - \eta\, g_{b_h} = \begin{bmatrix} 1.5 \\ 2 \end{bmatrix} - 0.1 \begin{bmatrix} -0.034 \\ -0.0447 \end{bmatrix} = \begin{bmatrix} 1.5034 \\ 2.00477 \end{bmatrix}$$

$$W_o^{new} = W_o^{old} - \eta\, g_{W_o} = \begin{bmatrix} 1 & 0.5 \end{bmatrix} = -0.1 \begin{bmatrix} -0.4486 & -0.1305 \end{bmatrix} = \begin{bmatrix} 1.04486 & 0.51306 \end{bmatrix}$$

$$\mathbf{b}_o^{new} = \mathbf{b}_o^{old} - \eta\, g_{b_o} = -1 - 0.1(-0.4854) = -0.95146$$

5. Model Parameters

1. size of the kernel

$\frac{205-H_W+3}{3} = 66 \Rightarrow H_W = W_W = 10\ ,\ C = 10\ ,\ D = 96$

96 channel of $10 \times 10 \times 10$

2. number of trainable parameters

number of trainable parameters in this layer is

$$H_W \times W_W \times C \times D = 10 \times 10 \times 10 \times 96 = 96000$$

3. number of multiplication operations

each output element need $H_W \times W_W \times C$. So the number of multiplication operations required to obtain the output is $66 \times 66 \times 96 \times 10 \times 10 \times 10 = 418176000$