SC4001 Exam 2022-2023 Sem 1

1. (a) Briefly state the following. Each part carries 2 marks.

(i) The difference between gradient descent (GD) and stochastic gradient descent (SGD) learning algorithms.

Answer

- GD is accurate but computationally heavy.
- SGD is faster, more memory-efficient, but introduces more variance/noise in the updates.

(ii) How the time to weight update varies with batch size in semi-batch SGD learning.

Answer

- Time per update increases as batch size increases (because more data points must be processed before updating weights).
- However, larger batches may lead to fewer updates needed overall for convergence.

(iii) How one could use a linear neuron to learn a given nonlinear equation.

Answer

- As long as the neuron is a linear combiner followed by a non-linear activation function, then regardless of the form of non-linearity used, the neuron can perform pattern classification only on linearly separable patterns.

(iv) How a discrete perceptron is able to perform linear classification.

Answer

- The discrete perceptron performs linear classification by learning a hyperplane that separates two classes, using a weighted sum followed by a step function.

(v) Two ways to initialize weights of a network to improve convergence.

Answer

- Two common methods are small random initialization and Xavier initialization depending on the activation function.

1. (b) A two-layer discrete perceptron network receives 2-dimensional inputs $(x_1, x_2)^2 \in \boldsymbol{R}^2$ and has one output neuron. The shaded region of Figure Q1 shows the input space for which the output of the network, $y = 1$. Draw the network clearly indicating the values of the weights and biases of the neurons.

(13 marks)

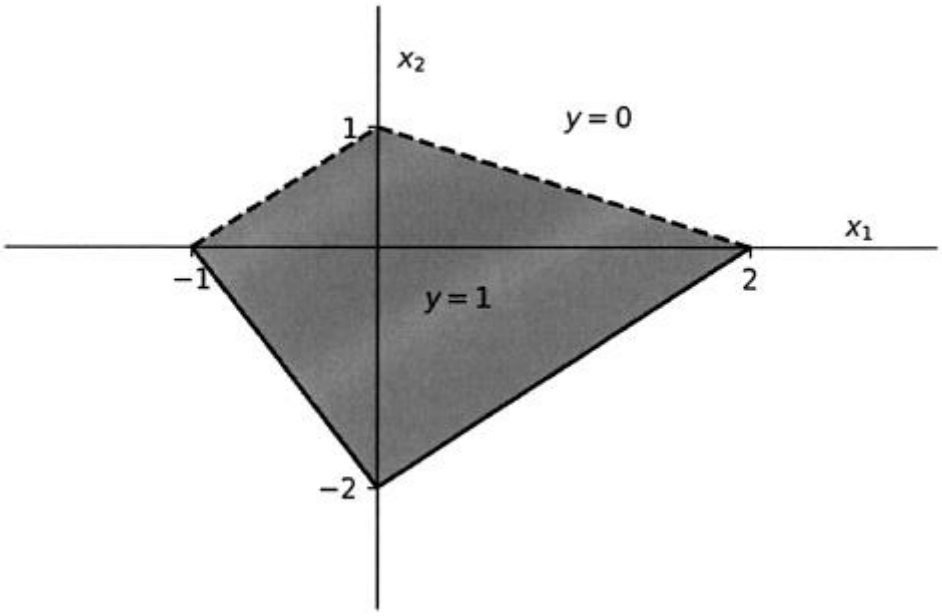


Figure Q1

1. (b) cont

Answer

Line 1 (passing through (-1, 0) and (0, 1))
Line 2 (passing through (0, 1) and (2, 0))
Line 3 (passing through (2, 0) and (0, -2))
Line 4 (passing through (0, -2) and (-1, 0))

shaded above the line is +1 > 0
shaded below the line is -1 ≤ 0

Line 1 (passing through (-1, 0) and (0, 1))

$$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{1 - 0}{0 - (-1)} = 1$$

$$y = mx + c$$
$$0 = 1(-1) + c$$
$$c = 1$$

$$x_2 = (1)x_1 + 1$$
$$x_1 - x_2 + 1 = 0$$

since the shaded-region below the line,
$u_1 = x_1 - x_2 + 1$

Line 2 (passing through (0, 1) and (2, 0))

$$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{0 - 1}{2 - 0} = \frac{-1}{2}$$

$$y = mx + c$$
$$1 = -\frac{1}{2}(0) + c$$
$$c = 1$$

$$x_2 = -\frac{1}{2}x_1 + 1$$
$$2x_2 = -x_1 + 2$$
$$x_1 + 2x_2 - 2 = 0$$

since the shaded-region below the line,
$u_2 = x_1 + 2x_2 - 2$

Line 3 (passing through (2, 0) and (0, -2))

$$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{(-2) - 0}{0 - 2} = 1$$

$$y = mx + c$$
$$0 = 1(2) + c$$
$$c = -2$$

$$x_2 = (1)x_1 - 2$$
$$x_1 - x_2 - 2 = 0$$

since the shaded-region above the line,
$u_3 = -x_1 + x_2 + 2$

Line 4 (passing through (0, -2) and (-1, 0))

$$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{0 - (-2)}{(-1) - 0} = -2$$

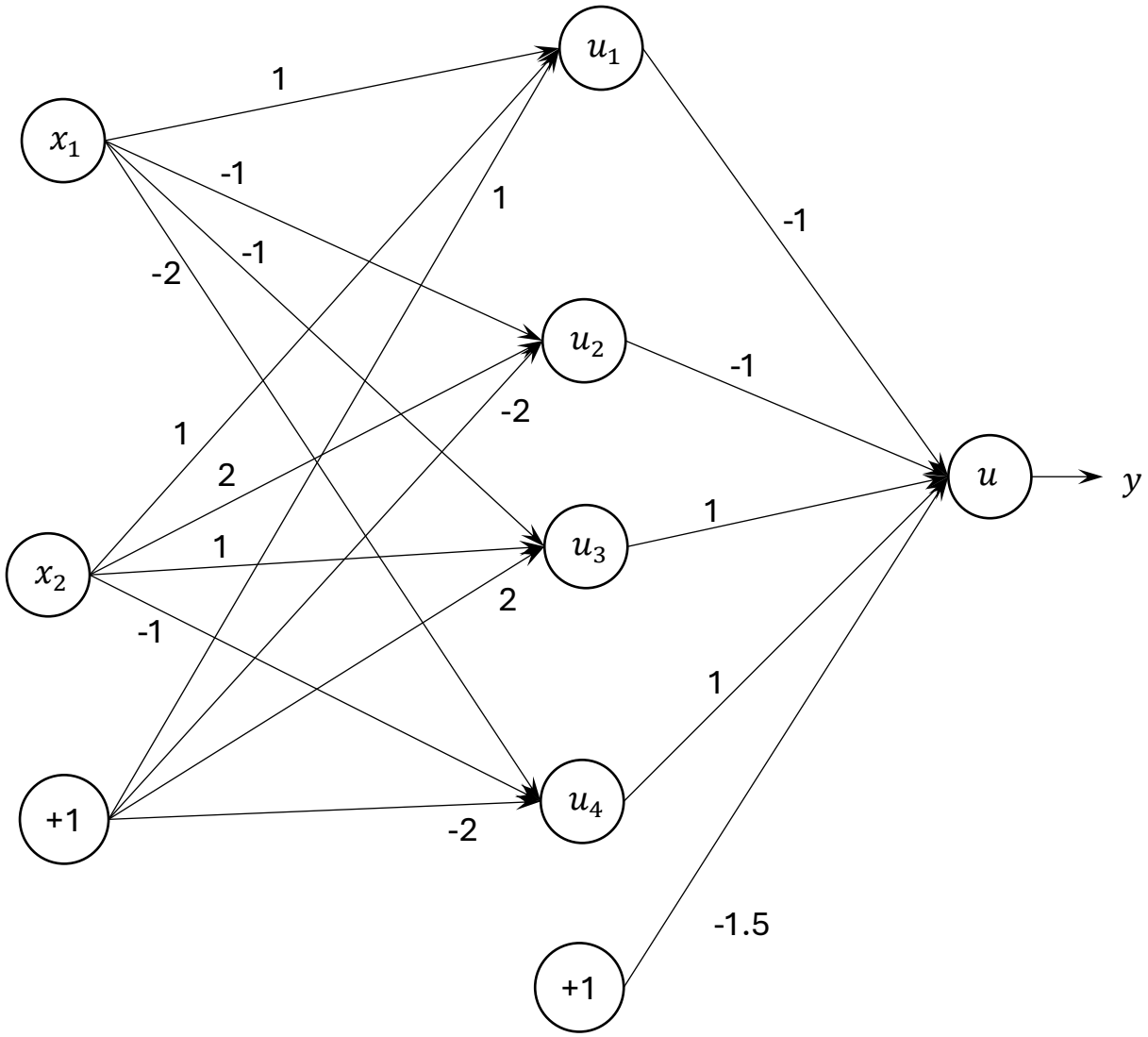
$$y = mx + c$$
$$-2 = -2(0) + c$$
$$c = -2$$

$$x_2 = (-2)x_1 - 2$$
$$2x_1 + x_2 + 2 = 0$$

since the shaded-region above the line,
$u_4 = -2x_1 - x_2 - 2$

1. (b) cont

Answer



$$u = y_1 - y_2 + y_3 - y_4 - 1.5$$

2. A two-layer feedforward neural network shown in Figure Q2 receives two-dimensional inputs $(x_1, x_2) \in \boldsymbol{R}^2$ and produces two-dimensional outputs (y_1, y_2). The hidden layer consists of three perceptrons with activation functions $f(u) = \frac{2}{1+e^{-u}}$ and the output layer is a linear layer with two neurons. The weights of the network are initialized as indicated in the figure and all the biases are initialized to 0.1 (not shown).

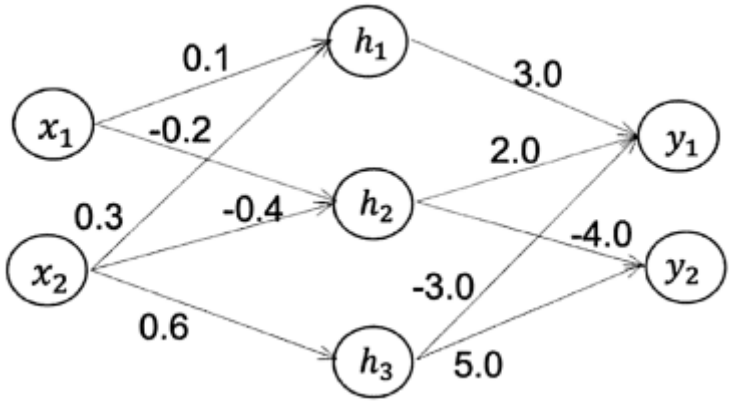


Figure Q2

The network is trained to produce a desired output $\boldsymbol{d} = \begin{pmatrix} -1.0 \\ 1.0 \end{pmatrix}$ for an input $\boldsymbol{x} = \begin{pmatrix} 1.5 \\ 2.0 \end{pmatrix}$. You are to perform one iteration of gradient descent learning with the example $(\boldsymbol{x}, \boldsymbol{d})$. The learning factor $\alpha = 0.1$ Give your answers rounded up to two decimal places.

2. (a) Write initial weight matrices $\boldsymbol{W}$ and bias vectors $\boldsymbol{b}$ of the hidden layer, and initial weight matrix $\boldsymbol{V}$ and bias vector $\boldsymbol{c}$ of the output layer.

(2 marks)

Answer

$$W = \begin{pmatrix} 0.1 & -0.2 & 0.0 \\ 0.3 & -0.4 & 0.6 \end{pmatrix}, \quad b = \begin{pmatrix} 0.1 \\ 0.1 \\ 0.1 \end{pmatrix}, \quad V = \begin{pmatrix} 3.0 & 0.0 \\ 2.0 & -4.0 \\ -3.0 & 5.0 \end{pmatrix}, c = \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix}$$

2. (b) Find the synaptic inputs $\boldsymbol{z}$ and output $\boldsymbol{h}$ of the hidden layer and the output y of the output layer.

(6 marks)

Answer

$$W = \begin{pmatrix} 0.1 & -0.2 & 0.0 \\ 0.3 & -0.4 & 0.6 \end{pmatrix}, \qquad b = \begin{pmatrix} 0.1 \\ 0.1 \\ 0.1 \end{pmatrix}, \qquad V = \begin{pmatrix} 3.0 & 0.0 \\ 2.0 & -4.0 \\ -3.0 & 5.0 \end{pmatrix}, c = \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix}$$

Synaptic input to hidden-layer,

$$\boldsymbol{z} = \boldsymbol{W}^T\boldsymbol{x} + \boldsymbol{b} = \begin{pmatrix} -0.1 & 0.3 \\ -0.2 & -0.4 \\ 0.0 & 0.6 \end{pmatrix}\begin{pmatrix} 1.5 \\ 2.0 \end{pmatrix} + \begin{pmatrix} 0.1 \\ 0.1 \\ 0.1 \end{pmatrix} = \begin{pmatrix} (-0.1)(1.5) + (0.3)(2) \\ (-0.2)(1.5) + (-0.4)(2) \\ (0.0)(1.5) + (0.6)(2) \end{pmatrix} + \begin{pmatrix} 0.1 \\ 0.1 \\ 0.1 \end{pmatrix}$$

$$= \begin{pmatrix} 0.75 \\ -1.10 \\ 1.20 \end{pmatrix} + \begin{pmatrix} 0.1 \\ 0.1 \\ 0.1 \end{pmatrix} = \begin{pmatrix} 0.85 \\ -1.00 \\ 1.30 \end{pmatrix}$$

Output of the hidden layer,

$$\boldsymbol{h} = g(\boldsymbol{z}) = \frac{2}{1 + e^{-z}} = \begin{pmatrix} 1.40 \\ 0.54 \\ 1.57 \end{pmatrix}$$

Output of output layer

$$\boldsymbol{y} = \boldsymbol{V}^T\boldsymbol{h} + \boldsymbol{c} = \begin{pmatrix} 3.0 & 2.0 & -3.0 \\ 0.0 & -4.0 & 5.0 \end{pmatrix}\begin{pmatrix} 1.40 \\ 0.54 \\ 1.57 \end{pmatrix} + \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix}$$

$$= \begin{pmatrix} (3)(1.40) + (2)(0.54) + (-3)(1.57) \\ (0)(1.40) + (-4)(0.54) + (5)(1.57) \end{pmatrix} + \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix} = \begin{pmatrix} 0.57 \\ 5.69 \end{pmatrix} + \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix} = \begin{pmatrix} 0.67 \\ 5.79 \end{pmatrix}$$

<u>Summary</u>

synaptic input $z = \begin{pmatrix} 0.85 \\ -1.00 \\ 1.30 \end{pmatrix}$

hidden layer $h = \begin{pmatrix} 1.40 \\ 0.54 \\ 1.57 \end{pmatrix}$

output layer $y = \begin{pmatrix} 0.67 \\ 5.79 \end{pmatrix}$

This is not softmax, so no need to perform

$$y = \begin{pmatrix} \frac{e^{6.02}}{e^{6.02} + e^{1.7}} \\ \frac{e^{1.7}}{e^{6.02} + e^{1.7}} \end{pmatrix}$$

2. (c) Find the square error at the output.

(2 marks)

Answer

$$\boldsymbol{d} = \begin{pmatrix} -1.0 \\ 1.0 \end{pmatrix}, \qquad y = \begin{pmatrix} 0.67 \\ 5.79 \end{pmatrix}$$

Square error at output

$$J = \sum_{k=1}^{K} (d_k - y_k)^2$$

$$= (d_1 - y_1)^2 + (d_2 - y_2)^2$$

$$= \left([(-1) - 0.67]^2 + (1 - 5.79)^2\right)$$

$$= [(-1.67)^2 + (-4.79)^2]$$

$$= 25.73$$

Summary

Square error at output $J = 25.73$

2. (d) Find the derivatives $f'(\boldsymbol{z})$ at the hidden layer.

(3 marks)

Answer

Derivatives of $f(u)$

$$y = f(y) = \frac{1}{1+e^{-u}}$$
$$f'(y) = y(1-y)$$

$$z = f(z) = \frac{2}{1+e^{-u}} = 2\text{y}$$
$$y = \frac{z}{2}$$

$$f'(z) = 2y(1-y) = 2\frac{z}{2}\left(1-\frac{z}{2}\right) = z(1-\frac{z}{2})$$

Hidden layer h,

$$\boldsymbol{h} = g(\boldsymbol{z}) = \frac{2}{1+e^{-z}} = \begin{pmatrix} 1.40 \\ 0.54 \\ 1.57 \end{pmatrix}$$

$$f'(z) = z(1-\frac{z}{2}) = \begin{pmatrix} 1.40\left[1-\left(\frac{1.40}{2}\right)\right] \\ 0.54\left[1-\left(\frac{0.54}{2}\right)\right] \\ 1.57\left[1-\left(\frac{1.57}{2}\right)\right]) \end{pmatrix} = \begin{pmatrix} 0.42 \\ 0.39 \\ 0.34 \end{pmatrix}$$

Summary

Derivation of $f'(z)$

$$f'(z) = \begin{pmatrix} 0.42 \\ 0.39 \\ 0.34 \end{pmatrix}$$

2. (e) Find gradients $\nabla_{\boldsymbol{u}} J$ and $\nabla_{\boldsymbol{z}} J$ of the cost J with respect to $\boldsymbol{u}$ and $\boldsymbol{z}$, respectively.

(6 marks)

Answer

$$\boldsymbol{d} = \begin{pmatrix} -1.0 \\ 1.0 \end{pmatrix}, \qquad y = \begin{pmatrix} 0.67 \\ 5.79 \end{pmatrix}$$

Gradient $\nabla_{\boldsymbol{u}} J$,

$$\nabla_{\boldsymbol{u}} J = -(d - y) = -\left(\begin{pmatrix} -1.0 \\ 1.0 \end{pmatrix} - \begin{pmatrix} 0.67 \\ 5.79 \end{pmatrix}\right) = -\begin{pmatrix} -1.67 \\ -4.79 \end{pmatrix} = \begin{pmatrix} 1.67 \\ 4.79 \end{pmatrix}$$

Gradient $\nabla_{\boldsymbol{z}} J$,

$$V = \begin{pmatrix} 3.0 & 0.0 \\ 2.0 & -4.0 \\ -3.0 & 5.0 \end{pmatrix}, \qquad f'(z) = \begin{pmatrix} 0.42 \\ 0.39 \\ 0.34 \end{pmatrix}$$

$$\nabla_{\boldsymbol{z}} J = V(\nabla_u J) \cdot f'(\boldsymbol{z}) = \begin{pmatrix} 3.0 & 0.0 \\ 2.0 & -4.0 \\ -3.0 & 5.0 \end{pmatrix} \begin{pmatrix} 1.67 \\ 4.79 \end{pmatrix} \cdot \begin{pmatrix} 0.42 \\ 0.39 \\ 0.34 \end{pmatrix}$$

$$= \begin{pmatrix} (3)(1.67) + (0)(4.79) \\ (2)(1.67) + (-4)(4.79) \\ (-3)(1.67) + (5)(4.79) \end{pmatrix} \cdot \begin{pmatrix} 0.42 \\ 0.39 \\ 0.34 \end{pmatrix}$$

$$= \begin{pmatrix} 5.01 \\ -15.82 \\ 18.94 \end{pmatrix} \cdot \begin{pmatrix} 0.42 \\ 0.39 \\ 0.34 \end{pmatrix} = \begin{pmatrix} 2.10 \\ -6.17 \\ 6.44 \end{pmatrix}$$

Summary

Gradient $\nabla_{\boldsymbol{u}} J$ $\qquad \nabla_{\boldsymbol{u}} J = \begin{pmatrix} 1.67 \\ 4.79 \end{pmatrix}$

Gradient $\nabla_{\boldsymbol{z}} J$ $\qquad \nabla_{\boldsymbol{z}} J = \begin{pmatrix} 2.10 \\ -6.17 \\ 6.44 \end{pmatrix}$

2. (f) Find gradients $\nabla_{\boldsymbol{V}} J$, $\nabla_{\boldsymbol{c}} J$, $\nabla_{\boldsymbol{W}} J$, and $\nabla_{\boldsymbol{b}} J$ of the cost J with respect $\boldsymbol{V}$, c, $\boldsymbol{W}$, and $\boldsymbol{b}$, respectively.

(4 marks)

Answer

$$\nabla_{\boldsymbol{u}} J = \begin{pmatrix} 1.67 \\ 4.79 \end{pmatrix}, \qquad \nabla_{\boldsymbol{z}} J = \begin{pmatrix} 2.10 \\ -6.17 \\ 6.44 \end{pmatrix} \qquad \boldsymbol{h} = \begin{pmatrix} 1.40 \\ 0.54 \\ 1.57 \end{pmatrix}$$

Output layer:

$$\nabla_V J = h(\nabla_U J)^T = \begin{pmatrix} 1.40 \\ 0.54 \\ 1.57 \end{pmatrix} (1.67 \quad 4.79) = \begin{pmatrix} (1.40)(1.67) & (1.40)(4.79) \\ (0.54)(1.67) & (0.54)(4.79) \\ (1.57)(1.67) & (1.57)(4.79) \end{pmatrix}$$

$$= \begin{pmatrix} 2.34 & 6.71 \\ 0.90 & 2.59 \\ 2.62 & 7.52 \end{pmatrix}$$

$$\nabla_c J = \nabla_{\boldsymbol{u}} J \quad = \begin{pmatrix} 1.67 \\ 4.79 \end{pmatrix}$$

Hidden layer:

$$\boldsymbol{x} = \begin{pmatrix} 1.5 \\ 2.0 \end{pmatrix}$$

$$\nabla_W J = x(\nabla_Z J)^T = \begin{pmatrix} 1.5 \\ 2.0 \end{pmatrix} (2.10 \quad -6.17 \quad 6.44)$$

$$= \begin{pmatrix} (1.5)(2.1) & (1.5)(-6.17) & (1.5)(6.44) \\ (2.0)(2.1) & (2.0)(-6.17) & (2.0)(6.44) \end{pmatrix}$$

$$= \begin{pmatrix} 3.15 & -9.26 & 9.66 \\ 4.20 & -12.34 & 12.88 \end{pmatrix}$$

$$\nabla_b J = \nabla_Z J \quad = \begin{pmatrix} 2.10 \\ -6.17 \\ 6.44 \end{pmatrix}$$

Summary

$$\nabla_V J = \begin{pmatrix} 2.34 & 6.71 \\ 0.90 & 2.59 \\ 2.62 & 7.52 \end{pmatrix}$$

$$\nabla_c J = \begin{pmatrix} 1.67 \\ 4.79 \end{pmatrix}$$

$$\nabla_W J = \begin{pmatrix} 3.15 & -9.26 & 9.66 \\ 4.20 & -12.34 & 12.88 \end{pmatrix}$$

$$\nabla_b J \quad = \begin{pmatrix} 2.10 \\ -6.17 \\ 6.44 \end{pmatrix}$$

2. (g) Find the updated values of $\boldsymbol{V}, \boldsymbol{c}, \boldsymbol{W}$, and $\boldsymbol{b}$.

(2 marks)

Answer

$$W = \begin{pmatrix} 0.1 & -0.2 & 0.0 \\ 0.3 & -0.4 & 0.6 \end{pmatrix}, \quad b = \begin{pmatrix} 0.1 \\ 0.1 \\ 0.1 \end{pmatrix}, \quad V = \begin{pmatrix} 3.0 & 0.0 \\ 2.0 & -4.0 \\ -3.0 & 5.0 \end{pmatrix}, \quad c = \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix} \quad \alpha = 0.1$$

$$\nabla_V J = \begin{pmatrix} 2.34 & 6.71 \\ 0.90 & 2.59 \\ 2.62 & 7.52 \end{pmatrix}$$

$$\nabla_c J = \begin{pmatrix} 1.67 \\ 4.79 \end{pmatrix}$$

$$\nabla_W J = \begin{pmatrix} 3.15 & -9.26 & 9.66 \\ 4.20 & -12.34 & 12.88 \end{pmatrix}$$

$$\nabla_b J = \begin{pmatrix} 2.10 \\ -6.17 \\ 6.44 \end{pmatrix}$$

$$\boldsymbol{V} = \boldsymbol{V} - \alpha \nabla_V J = \begin{pmatrix} 3.0 & 0.0 \\ 2.0 & -4.0 \\ -3.0 & 5.0 \end{pmatrix} - 0.1 \begin{pmatrix} 2.34 & 6.71 \\ 0.90 & 2.59 \\ 2.62 & 7.52 \end{pmatrix} = \begin{pmatrix} 3.0 & 0.0 \\ 2.0 & -4.0 \\ -3.0 & 5.0 \end{pmatrix} - \begin{pmatrix} 0.23 & 0.67 \\ 0.09 & 0.26 \\ 0.26 & 0.75 \end{pmatrix}$$

$$= \begin{pmatrix} 2.77 & -0.67 \\ 1.91 & -4.26 \\ -3.26 & 4.25 \end{pmatrix}$$

$$\boldsymbol{c} = \boldsymbol{c} - \alpha \nabla_c J = \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix} - 0.1 \begin{pmatrix} 1.67 \\ 4.79 \end{pmatrix} = \begin{pmatrix} 0.1 \\ 0.1 \end{pmatrix} - \begin{pmatrix} 0.17 \\ 0.48 \end{pmatrix} = \begin{pmatrix} 0.07 \\ -0.38 \end{pmatrix}$$

$$\boldsymbol{W} = W - \alpha \nabla_W J = \begin{pmatrix} 0.1 & -0.2 & 0.0 \\ 0.3 & -0.4 & 0.6 \end{pmatrix} - 0.1 \begin{pmatrix} 3.15 & -9.26 & 9.66 \\ 4.20 & -12.34 & 12.88 \end{pmatrix}$$

$$= \begin{pmatrix} 0.1 & -0.2 & 0.0 \\ 0.3 & -0.4 & 0.6 \end{pmatrix} - \begin{pmatrix} 0.32 & -0.93 & 0.97 \\ 0.42 & -1.23 & 1.29 \end{pmatrix} = \begin{pmatrix} -0.22 & 0.73 & -0.97 \\ -0.12 & 0.83 & -0.69 \end{pmatrix}$$

$$\boldsymbol{b} = \boldsymbol{b} - \alpha \nabla_b J = \begin{pmatrix} 0.1 \\ 0.1 \\ 0.1 \end{pmatrix} - 0.1 \begin{pmatrix} 2.10 \\ -6.17 \\ 6.44 \end{pmatrix} = \begin{pmatrix} 0.1 \\ 0.1 \\ 0.1 \end{pmatrix} - \begin{pmatrix} 0.21 \\ -0.62 \\ 0.64 \end{pmatrix} = \begin{pmatrix} -0.11 \\ 0.72 \\ -0.54 \end{pmatrix}$$

Summary

Updated $V = \begin{pmatrix} 2.77 & -0.67 \\ 1.91 & -4.26 \\ -3.26 & 4.25 \end{pmatrix}$

Updated $\boldsymbol{c} = \begin{pmatrix} 0.07 \\ -0.38 \end{pmatrix}$

Updated $W = \begin{pmatrix} -0.22 & 0.73 & -0.97 \\ -0.12 & 0.83 & -0.69 \end{pmatrix}$

Updated $\boldsymbol{b} = \begin{pmatrix} -0.11 \\ 0.72 \\ -0.54 \end{pmatrix}$

3. In the following questions, the size of input or output volume is represented as *D x H x W*, where *D* is the number of channels, and *H x W* is the spatial size.

3. (a) Given an input volume of size 3 x 512 x 512, we have 64 convolution filters each with a size of 3 x 5 x 5, stride = 1.

3. (a) (i) What is the output volume size if we use a padding size of 2?

(3 marks)

Answer

- N = 512
- F = 5
- S = 1
- P = 2

$$\text{Output} = \frac{N - F + 2P}{S} + 1 = \frac{512 - 5 + 2(2)}{1} + 1 = 511 + 1 = 512$$

- number of filters = 64

Output volume size = 64 x 512 x 512

3. (a) (ii) Give a reason why one would use padding in a convolution layer.

(2 marks)

Answer

- Padding is used in a convolution layer to preserve the spatial dimensions of the input volume in the output, ensuring no loss of resolution and proper handling of edge pixels.

3. (a) (iii) What is the total number of parameters in this layer? Be reminded to account for the bias terms.

(3 marks)

Answer

- filter size = d x F x F = 3 x 5 x 5
- d = 3
- F = 5
- number of filters = 64, n = 64

Parameters per filer	= (d * F * F) + 1	= (3 * 5 * 5) + 1	= 76
Total parameters	= param * n	= 76 * 64	= 4864

3. (b) (i) Calculate the FLOPs of a standard convolution layer. Assume the filter size is 3 x 3, the input volume is 3 x 45 x 45, and the output volume is 128 x 43 x 43. Be reminded to account for the bias terms.

(3 marks)

Answer

Standard convolution layer

- Since input size: 3 x 45 x 45, $D_1 = 3$
- Since filter size: 3 x 3, $F = 3$
- Output volume: 128 x 43 x 43, $D_2 * H_2 * W_2 = 128 * 43 * 43$

Standard Convolution Layer

$$\text{FLOPs}_{\text{standard}} = (2 * D_1 * F^2) * D_2 * H_2 * W_2$$

$$= 2 * 3 * 3^2 * 128 * 43 * 43 = 12{,}780{,}288$$

3. (b) (ii) Compute the reduction rate of FLOPs if we replace the standard convolution with a depthwise separable convolution (i.e., "depthwise + pointwise" convolutions). Assume the filter size of the depthwise convolution is 3 x 3, the input volume is still 3 x 45 x 45, and the output volume is 128 x 43x 43. Be reminded to account for the bias terms.

(3 marks)

Answer

From Q3(b)(i),

Depthwise convolution layer

- Input size 3 x 45 x 45, $D_1 = 3$
- size of filter 3 x 3, $F = 3$

Pointwise convolution layer

- Output volume: 128 x 43 x 43 $\Rightarrow D_2 * H_2 * W_2 = 128 * 43 * 43$

Depthwise separable convolution

$\text{FLOPs}_{\text{depthwise}} = (2 * F^2) * D_1 * H_2 * W_2$ = (2 * 3^2)* 3 * 43 * 43 = 99,846

$\text{FLOPs}_{\text{pointwise}} = (2 * D_1) * D_2 * H_2 * W_2$ = (2 * 3) * 128 * 43 * 43 = 1,420,032

$\text{FLOPs}_{\text{separable}} = \text{FLOPs}_{\text{depthwise}} + \text{FLOPs}_{\text{pointwise}}$

= 99,846 + 1,42-,032 = 1,519,878

Standard Convolution Layer

$\text{FLOPs}_{\text{standard}}$ = 12,780,288

$$\text{Reduction Rate} = \frac{\text{FLOPs}_{\text{standard}} - \text{FLOPs}_{\text{separable}}}{\text{FLOPs}_{\text{standard}}}$$

$$= \frac{12{,}780{,}288 - 1{,}519{,}878}{12{,}780{,}288} = \frac{1015}{1155}$$

$$= 0.881 = 88.1\%$$

Summary

Reduction rate = 0.881 = 88.1%

3. (b) (iii) Figure Q3 on page 5 depicts a block that consists of three convolutional layers. The input volume has a size of 256 x 32 x 32 and the second layer has 32 convolution filters each with a size of 64 x 3 x 3, stride = 1 and padding = 1.

Provide the values of n_1, d_1, F_1, n_2, d_2, and F_2 to form a valid block. Explain your design.

(8 marks)

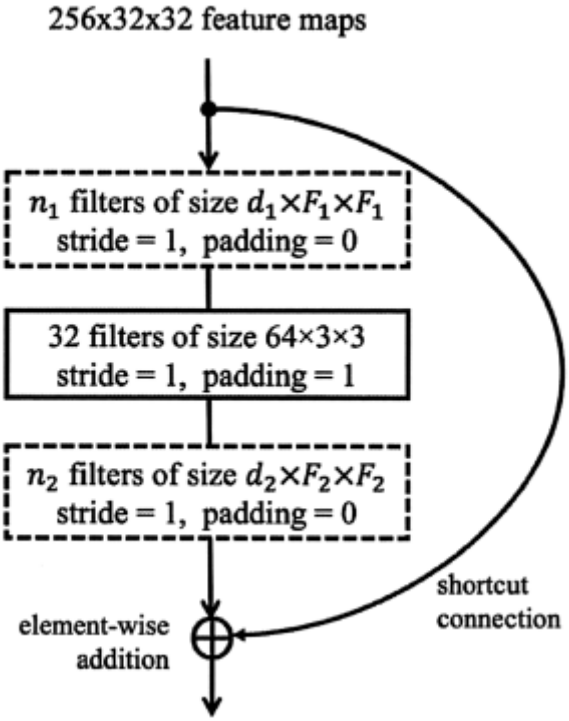


Figure Q3

Answer

Layer 1

- Input volume: 256 x 32 x 32 $\Rightarrow d_1 = 256$
- stride S = 1
- padding P = 0

$$\text{Output} = \frac{N - F + 2P}{S} + 1 = \frac{N - F_1 + 2(0)}{1} + 1 = N - F_1 + 1$$

- To maintain same size, output must be equal to N

$$\text{Output} = N$$
$$N - F_1 + 1 = N$$
$$F_1 = 1$$

3. (b) (iii) cont

Answer

Layer 2

- number of filters = 32 $\Rightarrow d_2 = 32$
- filer size = 64 x 3 x 3 $\Rightarrow n_1 = 64$

Layer 3

- number of filters $\Rightarrow n_2 = d_1 = 256$
- filer size = 64 x 3 x 3 $\Rightarrow n_1 = 64$

- element-wise addition, use 1 x 1 to keep spatial resolution

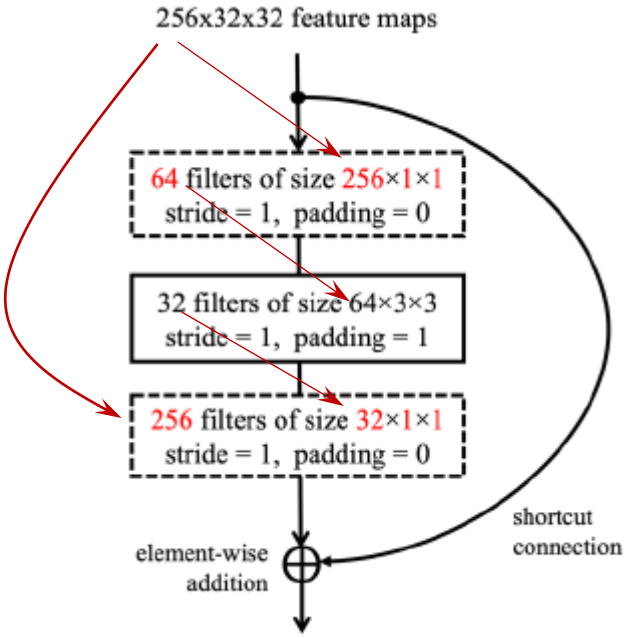
$\Rightarrow F_2 = 1$

To form a valid block, the size the of output volume at the residual branch has to be the same size as the input volume, which is 256×32×32, such that element-wise addition can be performed, thus n_2 = 256 and 1×1 is chosen to keep the spatial resolution.

The values of d_1 and d_2 are chosen to match the depth of their corresponding input. n_1 is chosen to match the filter size of the second layer.

Summary

- $n_1 = 64$
- $d_1 = 256$
- $F_1 = 1$
- $n_2 = 256$
- $d_2 = 32$
- $F_2 = 1$



3. (c) Select the correct option (A, B, C or D) for each question.

A. Both statements are TRUE.
B. Statement I is TRUE, but statement II is FALSE.
C. Statement I is FALSE, but statement II is TRUE.
D. Both statements are FALSE.

Answer

D (i) Statement I: Autoencoders are a supervised learning technique.
Statement II. Autoencoder's output is exactly the same as the input.

(1 mark)

Answer

F

A (ii) Statement I: One way to implement undercomplete autoencoder is to constrain the number of nodes present in hidden layer(s) of the neural network

T

Statement II. To train a denoising encoder, we use a loss between the original input and the reconstruction from a noisy version of the input.

T (1 mark)

C (iii) Statement I: Sparse autoencoders introduce information bottleneck by reducing the number of nodes at hidden layers. F

Statement II. With the sparsity constraint, we will observe more neuron outputs that are close to zero. T

(1 mark)

4. (a) Consider an Elman-type recurrent neural network (RNN) that receives 2-dimensional input patterns $\boldsymbol{x} \in \boldsymbol{R^2}$ and has one hidden layer. The RNN has five neurons in the hidden layer and two neurons in the output layer.

We denote the weight matrices connecting the input to the hidden layer as $\boldsymbol{U}$, the weight matrices connecting the previous hidden state to the next hidden layer as $\boldsymbol{W}$, and the weight matrices connecting the hidden output to the output layer as $\boldsymbol{V}$.

4. (a) (i) What is the dimensions of $\boldsymbol{U}, \boldsymbol{W}$ and $\boldsymbol{V}$, respectively.

(3 marks)

Answer

- Dimension of U: input x hidden neurons = 2 x 5
- Dimension of W: hidden x hidden = 5 x 5 (elman)
- Dimension of V: hidden x output = 5 x 2

4. (a) (ii) If we change this RNN to a Jordan-type RNN, and keep the same number of input dimensions, hidden and output neurons, what is the dimension of the top-down recurrence weight matrix $\boldsymbol{W}$?

(1 mark)

Answer

- Dimension of W: output x hidden = 2 x 5 (jordan)

4. (a) (iii) Explain the reason of observing exploding gradients in RNN training. Describe a way to address this problem.

(5 mark)

Answer

If the weights in this matrix are large, it can lead to a situation where the gradient signal is so large that it can cause learning to diverge. This is often referred to as exploding gradients.

Exploding gradient easily solved by clipping the gradients at a predefined threshold value.

Gradient Clipping:
Before updating weights, check if the gradient's norm exceeds a threshold. If it does, rescale the gradient so its norm equals the threshold.

4. (b) The statements below are all related to Transformers. Answer "TRUE" or "FALSE" to the following statements. Each part carries 1 mark.

Answer

F (i) Divide the dot product of the query vector with the key vector by the square root of the dimension of the key vectors leads to vanishing gradients.

F (ii) Multi-head attention alleviates the need for positional encoding.

T (iii) The encoder-decoder attention layer of each decoder layer accepts the query and value matrices obtained from the output of the encoder stack as input.

T (iv) In the same encoder layer, we apply the same feedforward network independently to each position.

T (v) Transformer uses positional encoding to help learn better attention.

T (vi) Positional encoding can either be pre-defined or made learnable.

(6 marks)

4. (c) Consider a self-attention layer in a Transformer, which receives the following key (**K**), query (**Q**), and value (**V**) matrices:

$$\mathrm{K} = \begin{pmatrix} \sqrt{2} & \sqrt{2} \\ \sqrt{2} & \sqrt{2} \end{pmatrix}, \qquad \mathrm{Q} = \begin{pmatrix} \ln 2 & \ln 3 \\ \ln 1 & \ln 4 \end{pmatrix} \text{ and } \mathrm{V} = \begin{pmatrix} 3 & 4 \\ 5 & 6 \end{pmatrix}$$

Compute the output of the scaled-dot product attention, Attention(Q, K, V).

(7 marks)

Answer

$$\mathrm{Attentiion}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathrm{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{\mathrm{D_k}}}\right)\mathbf{V}$$

$$\mathbf{Q}\mathbf{K}^{\mathrm{T}} = \begin{pmatrix} \ln 2 & \ln 3 \\ \ln 1 & \ln 4 \end{pmatrix}\begin{pmatrix} \sqrt{2} & \sqrt{2} \\ \sqrt{2} & \sqrt{2} \end{pmatrix} = \begin{pmatrix} (\ln 2)(\sqrt{2}) + (\ln 3)(\sqrt{2}) & (\ln 2)(\sqrt{2}) + (\ln 3)(\sqrt{2}) \\ (\ln 1)(\sqrt{2}) + (\ln 4)(\sqrt{2}) & (\ln 1)(\sqrt{2}) + (\ln 4)(\sqrt{2}) \end{pmatrix}$$
$$= \begin{pmatrix} 2.534 & 2.534 \\ 1.961 & 1.961 \end{pmatrix}$$

Perform row-wise SoftMax

$$\frac{e^{Z}j}{\sum_j e^{Z}j} = \frac{e^{1.792}}{e^{1.792} + e^{1.792}} = 0.5$$

$$\mathrm{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{\mathrm{D_k}}}\right) = \mathrm{Softmax}\left(\frac{\begin{pmatrix} 2.534 & 2.534 \\ 1.961 & 1.961 \end{pmatrix}}{\sqrt{2}}\right)$$
$$= \mathrm{Softmax}\begin{pmatrix} 1.792 & 1.792 \\ 1.387 & 1.387 \end{pmatrix} = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}$$

$$\mathrm{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{\mathrm{D_k}}}\right)\mathbf{V} = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}\begin{pmatrix} 3 & 4 \\ 5 & 6 \end{pmatrix} = \begin{pmatrix} (0.5)(3) + (0.5)(5) & (0.5)(4) + (0.5)(6) \\ (0.5)(3) + (0.5)(5) & (0.5)(4) + (0.5)(6) \end{pmatrix}$$
$$= \begin{pmatrix} 4 & 5 \\ 4 & 5 \end{pmatrix}$$

Summary

$$\mathrm{Attentiion}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \begin{pmatrix} 4 & 5 \\ 4 & 5 \end{pmatrix}$$

4. (c) Describe the changes that you need to make to turn an unconditional Generative Adversarial Network (GAN) to a conditional one that takes class labels. Explain one advantage of the conditional GAN in comparison to the unconditional GAN. (3 marks)

Answer

To turn an unconditional Generative Adversarial Network (GAN) into a conditional GAN (cGAN) that takes class labels, the key change is to provide the class label information as an additional input to both the generator and the discriminator.

Generator:

Instead of generating output only from random noise z, the generator will now take both z and a class label y as inputs. We can:

- Concatenate the label y with z before feeding it to the generator.
- Alternatively, embed y and fuse it later (e.g., through feature-wise transformations).

Discriminator:

The discriminator must also receive the class label y along with the input image. It learns to judge whether the given image matches the provided label and whether it is real or fake.

- Again, we can concatenate y with the image features.