MONASH University

FIT 3181/5215 Deep Learning

Quiz for: Stochastic Gradient Descent and Optimization

Trung Le and Tutor Team

Department of Data Science and AI
Faculty of Information Technology, Monash University
Email: trunglm@monash.edu/tuananh.bui@monash.edu

GROUP OF EIGHT AUSTRALIA

Question 1

- Consider the optimization problem to train a feed-forward NN

$$\min_\theta J(\theta) = \Omega(\theta) + \frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$$

where $\theta = \left[\left(W^k, b^k\right)\right]_{k=1}^{L}$ and $\Omega(\theta) = \lambda \sum_k \sum_{i,j} \left(W_{i,j}^{\mathbf{k}}\right)^2 = \lambda \sum_k \left\|\boldsymbol{W}^{\mathbf{k}}\right\|_F^2$. Choose the correct answers. (MC)

- A. Minimizing $\Omega(\theta)$ encourages more weights $W_{i,j}^{\mathbf{k}}$ become 0.
- B. Minimizing $\Omega(\theta)$ encourages complex models.
- C. Minimizing $\Omega(\theta)$ encourages simple models.
- D. Minimizing $\Omega(\theta)$ combats underfitting.
- E. Minimizing $\Omega(\theta)$ combats overfitting.

Question 1

- Consider the optimization problem to train a feed-forward NN

$$\min_\theta J(\theta) = \Omega(\theta) + \frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$$

where $\theta = \left[\left(W^k, b^k\right)\right]_{k=1}^{L}$ and $\Omega(\theta) = \lambda \sum_k \sum_{i,j} \left(W_{i,j}^{\mathbf{k}}\right)^2 = \lambda \sum_k \left\|\boldsymbol{W}^{\mathbf{k}}\right\|_F^2$. Choose the correct answers. (MC)

- A. Minimizing $\Omega(\theta)$ encourages more weights $W_{i,j}^{\mathbf{k}}$ become 0. **[x]**
- B. Minimizing $\Omega(\theta)$ encourages complex models.
- C. Minimizing $\Omega(\theta)$ encourages simple models. **[x]**
- D. Minimizing $\Omega(\theta)$ combats underfitting.
- E. Minimizing $\Omega(\theta)$ combats overfitting. **[x]**

Question 2

- Consider the optimization problem to train a feed-forward NN

$$\min_\theta J(\theta) = \Omega(\theta) + \frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$$

where $\theta = \left[\left(W^k, b^k\right)\right]_{k=1}^{L}$ and $\boldsymbol{f}(\boldsymbol{x_i};\boldsymbol{\theta})$ returns the prediction probabilities for $\boldsymbol{x}_i$. Choose the correct answers. (MC)

- A. $\frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$ is known as a regularization term.
- B. $\frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$ is known as an empirical loss.
- C. Minimizing $\frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$ makes the model more fit to the training set.
- D. Minimizing $\frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$ makes the model more fit to the testing set.
- E. Minimizing $\frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$ alone can lead to overfitting.

Question 2

- Consider the optimization problem to train a feed-forward NN

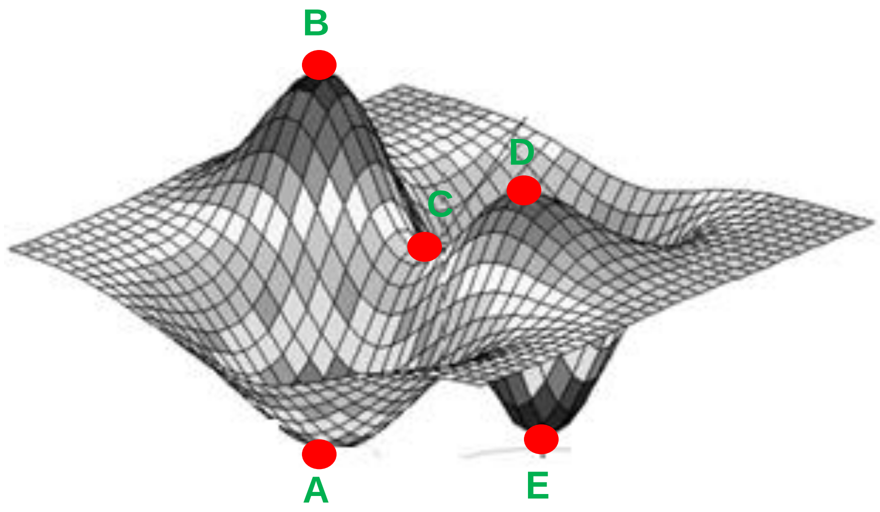
$$\min_\theta J(\theta) = \Omega(\theta) + \frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$$

where $\theta = \left[\left(W^k, b^k\right)\right]_{k=1}^{L}$ and $\boldsymbol{f}(\boldsymbol{x_i};\boldsymbol{\theta})$ returns the prediction probabilities for $\boldsymbol{x}_i$. Choose the correct answers. (MC)

- A. $\frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$ is known as a regularization term.
- B. $\frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$ is known as an empirical loss. **[x]**
- C. Minimizing $\frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$ makes the model more fit to the training set. **[x]**
- D. Minimizing $\frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$ makes the model more fit to the testing set.
- E. Minimizing $\frac{1}{N}\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} CE\big(y_i, f(x_i;\theta)\big)$ alone can lead to overfitting. **[x]**

Question 3

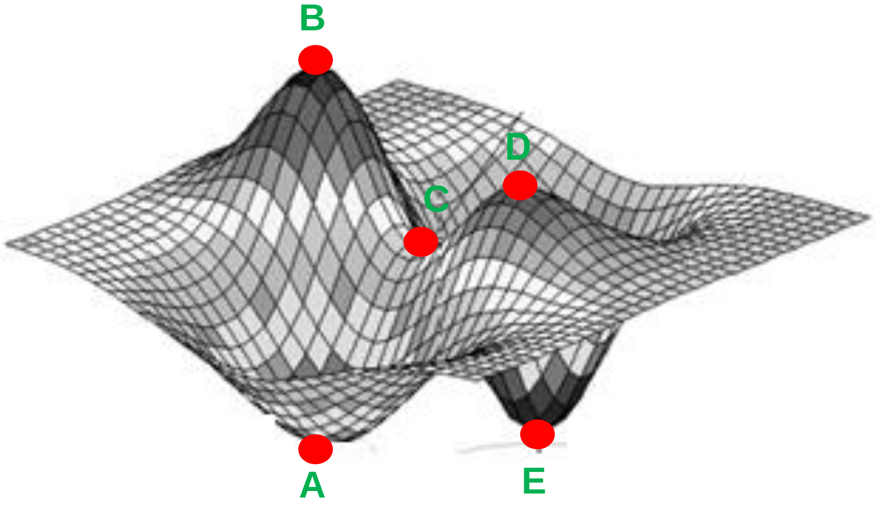
- Given a loss surface as showing, which statements are correct? (MC)



- A. A, B, C, D, E are critical points.
- B. A, E, and C are local minima, while B, D are local maxima.
- C. B and D are local minima, A and C are local maxima, and C is a saddle point
- D. C is a saddle point, B and D are local maxima, while A and E are local minima
- E. B is global maxima, D is local maxima, and C is a saddle point.

Question 3

- Given a loss surface as showing, which statements are correct? (MC)



- A. A, B, C, D, and E are critical points. **[x]**
- B. A, E, and C are local minima, while B, D are local maxima.
- C. B and D are local minima, A and C are local maxima, and C is a saddle point.
- D. C is a saddle point, B and D are local maxima, while A and E are local minima. **[x]**
- E. B is global maxima, D is local maxima, and C is a saddle point. **[x]**

Question 4

- Let $f(\theta) = \theta^2 - 2\theta + 1$. Assume that we use gradient descent with the learning rate $\eta = 0.1$ to solve $\min_{\theta} f(\theta)$. At the iteration t, we are at $\theta_t = 2$. What is θ_{t+1} at the next iteration?

- A. $\theta_{t+1} = 1$.
- B. $\theta_{t+1} = 0$
- C. $\theta_{t+1} = 1.8$.
- D. $\theta_{t+1} = 2.2$.

Question 4

- Let $f(\theta) = \theta^2 - 2\theta + 1$. Assume that we use gradient descent with the learning rate $\eta = 0.1$ to solve $\min_{\theta} f(\theta)$. At the iteration t, we are at $\theta_t = 2$. What is θ_{t+1} at the next iteration?

- A. $\theta_{t+1} = 1.$
- B. $\theta_{t+1} = 0$
- C. $\theta_{t+1} = 1.8.$ **[x]**
- D. $\theta_{t+1} = 2.2.$

$$\theta_{t+1} = \theta_t - \eta f'(\theta_t)$$
$$= 2 - 0.1(2 \times 2 - 2) = 1.8.$$

Question 5

- Given the function $f(\theta) = \frac{1}{1000}\sum_{i=1}^{1000}(\theta - i)^2$. We need to solve $\min_{\theta} f(\theta)$ using stochastic gradient descent with the learning rate $\eta = 0.1$. Assume we sample a batch $i_1 = 1, i_2 = 2, i_3 = 3, i_4 = 4$ of indices and at the iteration t, we have $\theta_t = 10$. What is the value of θ_{t+1} at the next iteration?

  - A. $\theta_{t+1} = 1000$.
  - B. $\theta_{t+1} = 8$.
  - C. $\theta_{t+1} = 9$.
  - D. $\theta_{t+1} = 8.5$.

Question 5

- Given the function $f(\theta) = \frac{1}{1000}\sum_{i=1}^{1000}(\theta - i)^2$. We need to solve $\min_{\theta} f(\theta)$ using stochastic gradient descent with the learning rate $\eta = 0.1$. Assume we sample a batch $i_1 = 1, i_2 = 2, i_3 = 3, i_4 = 4$ of indices and at the iteration t, we have $\theta_t = 10$. What is the value of θ_{t+1} at the next iteration?

- A. $\theta_{t+1} = 1000.$
- B. $\theta_{t+1} = 8.$
- C. $\theta_{t+1} = 9.$
- D. $\theta_{t+1} = 8.5.$ **[x]**

$$\tilde{f}(\theta) = \frac{1}{4}[(\theta-1)^2 + (\theta-2)^2 + (\theta-3)^4 + (\theta-4)^2]$$

$$\tilde{f}'(\theta) = 2\theta - 5$$

$$\theta_{t+1} = \theta_t - \eta\tilde{f}'(\theta_t) = 10 - 0.1 \times (2 \times 10 - 5) = 8.5.$$

Question 6

- Consider the optimization problem: $\min_{\theta} L(D;\theta) := \frac{1}{N}\sum_{i=1}^{N} l(x_i, y_i; \theta)$ with θ is the model parameter and $D = \{(x_1, y_1), \dots, (x_N, y_N)\}$ is a training set. Let us sample a batch of indices $i_1, \dots, i_b$ uniformly from $\{1, \dots, N\}$. Which statements are correct about the update rule of stochastic gradient descent? (SC)

  - A. $\theta_{t+1} = x_t - \frac{\eta}{N}\sum_{i=1}^{N} \nabla_\theta l(x_i, y_i; \theta_t)$.
  - B. $\theta_{t+1} = x_t + \frac{\eta}{N}\sum_{i=1}^{N} \nabla_\theta l(x_i, y_i; \theta_t)$.
  - C. $\theta_{t+1} = x_t - \frac{\eta}{b}\sum_{k=1}^{b} \nabla_\theta l\left(x_{i_k}, y_{i_k}; \theta_t\right)$.
  - D. $\theta_{t+1} = x_t + \frac{\eta}{b}\sum_{k=1}^{b} \nabla_\theta l\left(x_{i_k}, y_{i_k}; \theta_t\right)$.

Question 6

- Consider the optimization problem: $\min_{\theta} L(D;\theta) := \frac{1}{N}\sum_{i=1}^{N} l(x_i, y_i;\theta)$ with θ is the model parameter and $D = \{(x_1, y_1), \ldots, (x_N, y_N)\}$ is a training set. Let us sample a batch of indices $i_1, \ldots, i_b$ uniformly from $\{1, \ldots, N\}$. Which statements are correct about the update rule of stochastic gradient descent? (SC)

  - A. $\theta_{t+1} = \theta_t - \frac{\eta}{N}\sum_{i=1}^{N} \nabla_\theta l(x_i, y_i;\theta_t)$.
  - B. $\theta_{t+1} = \theta_t + \frac{\eta}{N}\sum_{i=1}^{N} \nabla_\theta l(x_i, y_i;\theta_t)$.
  - C. $\theta_{t+1} = \theta_t - \frac{\eta}{b}\sum_{k=1}^{b} \nabla_\theta l\left(x_{i_k}, y_{i_k};\theta_t\right)$. **[x]**
  - D. $\theta_{t+1} = \theta_t + \frac{\eta}{b}\sum_{k=1}^{b} \nabla_\theta l\left(x_{i_k}, y_{i_k};\theta_t\right)$.

Question 7

- Consider the optimization problem: $\min_{\theta} L(D;\theta) := \frac{1}{N}\sum_{i=1}^{N} l(x_i, y_i; \theta)$ with θ is the model parameter and $D = \{(x_1, y_1), \dots, (x_N, y_N)\}$ is a training set. Let us sample a batch of indices $i_1, \dots, i_b$ uniformly from $\{1, \dots, N\}$. Which statements are correct about the update rule of gradient descent? (SC)

  - A. $\theta_{t+1} = \theta_t - \frac{\eta}{N}\sum_{i=1}^{N} \nabla_\theta l(x_i, y_i; \theta_t)$.
  - B. $\theta_{t+1} = \theta_t + \frac{\eta}{N}\sum_{i=1}^{N} \nabla_\theta l(x_i, y_i; \theta_t)$.
  - C. $\theta_{t+1} = \theta_t - \frac{\eta}{b}\sum_{k=1}^{b} \nabla_\theta l\left(x_{i_k}, y_{i_k}; \theta_t\right)$.
  - D. $\theta_{t+1} = \theta_t + \frac{\eta}{b}\sum_{k=1}^{b} \nabla_\theta l\left(x_{i_k}, y_{i_k}; \theta_t\right)$.

Question 7

- Consider the optimization problem: $\min_{\theta} L(D;\theta) := \frac{1}{N}\sum_{i=1}^{N} l(x_i, y_i; \theta)$ with θ is the model parameter and $D = \{(x_1, y_1), \dots, (x_N, y_N)\}$ is a training set. Let us sample a batch of indices $i_1, \dots, i_b$ uniformly from $\{1, \dots, N\}$. Which statements are correct about the update rule of gradient descent? (SC)

  - A. $\theta_{t+1} = \theta_t - \frac{\eta}{N}\sum_{i=1}^{N} \nabla_{\theta} l(x_i, y_i; \theta_t)$. **[x]**
  - B. $\theta_{t+1} = \theta_t + \frac{\eta}{N}\sum_{i=1}^{N} \nabla_{\theta} l(x_i, y_i; \theta_t)$.
  - C. $\theta_{t+1} = \theta_t - \frac{\eta}{b}\sum_{k=1}^{b} \nabla_{\theta} l\left(x_{i_k}, y_{i_k}; \theta_t\right)$.
  - D. $\theta_{t+1} = \theta_t + \frac{\eta}{b}\sum_{k=1}^{b} \nabla_{\theta} l\left(x_{i_k}, y_{i_k}; \theta_t\right)$.

Question 8

- Matching A,B,C and 1,2,3

A) **Saddle points**

B) **Local minima**

C) **Local maxima**

1) **Local minima in all directions going through them**

2) **Local maxima in all directions going through them**

3) **Local minima in some directions and local maxima in other directions**

Question 8

- Matching A,B,C and 1,2,3

A) **Saddle points**

B) **Local minima**

C) **Local maxima**

1) **Local minima in all directions going through them**

2) **Local maxima in all directions going through them**

3) **Local minima in some directions and local maxima in other directions**

Question 9

- What are correct about this magic code line? (MC)
  - `optimizer = torch.optim.Adam(my_model.parameters(), lr=0.001)`

- A. optimizer is a SGD optimizer.
- B. optimizer is an Adam optimizer.
- C. optimizer holds a reference to the parameters of my_model, so it can update these parameters.
- D. optimizers cannot access the parameters of my_model.
- E. lr = 0.001 is a parameter of my_model.
- F. lr = 0.001 is a parameter of the optimizer specifying how much we want to update the model parameters.

Question 9

- What are correct about this magic code line? (MC)
  - `optimizer = torch.optim.Adam(my_model.parameters(), lr=0.001)`

- A. optimizer is a SGD optimizer.
- B. optimizer is an Adam optimizer. **[x]**
- C. optimizer holds a reference to the parameters of my_model, so it can update these parameters. **[x]**
- D. optimizers cannot access the parameters of my_model.
- E. lr = 0.001 is a parameter of my_model.
- F. lr = 0.001 is a parameter of the optimizer specifying how much we want to update the model parameters. **[x]**

Question 10

- Consider the following code snippet, what are correct? (MC)

```
x = torch.tensor([1.0, 2.0, 3.0], requires_grad=False)
y = torch.tensor([4.0])
W = torch.rand(3,1, requires_grad=True)
b = torch.rand(1, requires_grad=True)
```

```
y_hat = torch.matmul(x,W) + b
l = (y_hat - y)**2
```

- A. We can compute the gradient of the loss l w.r.t. x.
- B. We can compute the gradient of the loss l w.r.t. W.
- C. We can compute the gradient of the loss l w.r.t. b.
- D. W has W.grads to store the gradient and W.data to store its real values.
- E. b has b.grads to store the gradient and b.data to store its real values.
- F. x has x.grads to store the gradient and x.data to store its real values.

Question 10

- Consider the following code snippet, what are correct? (MC)

```
x = torch.tensor([1.0, 2.0, 3.0], requires_grad=False)
y = torch.tensor([4.0])
W = torch.rand(3,1, requires_grad=True)
b = torch.rand(1, requires_grad=True)
```

```
y_hat = torch.matmul(x,W) + b
l = (y_hat - y)**2
```

- A. We can compute the gradient of the loss l w.r.t. x.
- B. We can compute the gradient of the loss l w.r.t. W. **[x]**
- C. We can compute the gradient of the loss l w.r.t. b. **[x]**
- D. W has W.grads to store the gradient and W.data to store its real values. **[x]**
- E. b has b.grads to store the gradient and b.data to store its real values. **[x]**
- F. x has x.grads to store the gradient and x.data to store its real values.

Question 11

- Consider the following code snippet, what are correct? (MC)

```
x = torch.tensor([1.0, 2.0, 3.0], requires_grad=False)
y = torch.tensor([4.0])
W = torch.rand(3,1, requires_grad=True)
b = torch.rand(1, requires_grad=True)
```

```
y_hat = torch.matmul(x,W) + b
l = (y_hat - y)**2
```

```
l.backward(retain_graph=True)
#l.backward() raises RuntimeError if we try to
#suggests that you might be attempting to call
#on a tensor that has already had its gradient
print(f'W_grad = {W.grad}')
print(f'W_value={W.data}')
print(f'b_grad = {b.grad}')
print(f'b_value={b.data}')
```

- A. l.backward(…) performs forward propagation to compute $\hat{y}$ and l.
- B. l.backward(…) performs backward propagation to compute W.data and b.data.
- C. l.backward(…) performs backward propagation to compute W.grads and b.grads.
- D. l.backward(…) traverses from x to $\hat{y}$ and l.
- E. l.backward(…) traverses from l to $\hat{y}$ and x.

Question 11

- Consider the following code snippet, what are correct? (MC)

```
x = torch.tensor([1.0, 2.0, 3.0], requires_grad=False)
y = torch.tensor([4.0])
W = torch.rand(3,1, requires_grad=True)
b = torch.rand(1, requires_grad=True)
```

```
y_hat = torch.matmul(x,W) + b
l = (y_hat - y)**2
```

```
l.backward(retain_graph=True)
#l.backward() raises RuntimeError if we try to
#suggests that you might be attempting to call
#on a tensor that has already had its gradient
print(f'W_grad = {W.grad}')
print(f'W_value={W.data}')
print(f'b_grad = {b.grad}')
print(f'b_value={b.data}')
```

- A. l.backward(…) performs forward propagation to compute $\hat{y}$ and l.
- B. l.backward(…) performs backward propagation to compute W.data and b.data.
- C. l.backward(…) performs backward propagation to compute W.grads and b.grads. **[x]**
- D. l.backward(…) traverses from x to $\hat{y}$ and l.
- E. l.backward(…) traverses from l to $\hat{y}$ and x. **[x]**

Question 12

Given 4 implementations as below. What are f1/f2/f3/f4? (SC).

- ❑ A. sigmoid/tanh/softmax/relu
- ❑ B. softmax/tanh/relu/sigmoid
- ❑ C. sigmoid/tanh/relu/softmax
- ❑ D. softmax/tanh/sigmoid/relu

```
def f1(x):
    return 1. / (1 + np.exp(-x))


def f2(x):
    return (np.exp(2*x)-1.) / (np.exp(2*x)+1.)

def f3(x):
    return x*(x>0)

def f4(x):
    return np.exp(x) / np.sum(np.exp(x))
```

Question 12

Given 4 implementations as below. What are f1/f2/f3/f4? (SC).

- ❑ A. sigmoid/tanh/softmax/relu
- ❑ B. softmax/tanh/relu/sigmoid
- ❑ C. sigmoid/tanh/relu/softmax **[x]**
- ❑ D. softmax/tanh/sigmoid/relu

```
def f1(x):
    return 1. / (1 + np.exp(-x))


def f2(x):
    return (np.exp(2*x)-1.) / (np.exp(2*x)+1.)

def f3(x):
    return x*(x>0)

def f4(x):
    return np.exp(x) / np.sum(np.exp(x))
```

Question 13

Which is $\frac{\partial h}{\partial \bar{h}}$ *if* $h = \sigma(\bar{h})$, $\bar{h}$ *is a vector and* σ *is an activation function?*

- A. $\sigma'(\bar{h})$
- B. $\mathrm{d}iag(\bar{h})$
- C. $\mathrm{d}iag(\sigma(\bar{h}))$
- D. $\mathrm{d}iag(\sigma'(\bar{h}))$

Question 13

Which is $\frac{\partial h}{\partial \overline{h}}$ *if* $h = \sigma(\overline{h})$, $\overline{h}$ *is a vector and* σ *is an activation function?*

- A. $\sigma'(\overline{h})$
- B. $\mathrm{d}iag(\overline{h})$
- C. $\mathrm{d}iag(\sigma(\overline{h}))$
- D. $\mathrm{d}iag(\sigma'(\overline{h}))$ **[x]**

Question 14

Which is $\frac{\partial h}{\partial \overline{h}}$ *if* $h = \sigma(\overline{h})$, $\overline{h}$ = *[-1, 1, 2] and* σ *is ReLU activation function?*

$$A = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad D = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Question 14

Which is $\frac{\partial h}{\partial \overline{h}}$ *if* $h = \sigma(\overline{h})$, $\overline{h}$ *= [-1, 1, 2] and* σ *is ReLU activation function?*

$$A = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ [x] } D = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Question 15

Which is $\frac{\partial l}{\partial \overline{h}}$ *if* $h = \sigma(\overline{h})$, $\overline{h}$ *=[-1,1,2],* σ *is ReLU activation function and* $\frac{\partial l}{\partial h}$ = [1,1,1]

A. [-1, 1, 2]
B. [0, 1, 2]
C. [0, 1, 1]
D. [-1, 1, 1]

Question 15

Which is $\frac{\partial l}{\partial \overline{h}}$ *if* $h = \sigma(\overline{h})$, $\overline{h}$ *=[-1,1,2],* σ *is ReLU activation function and* $\frac{\partial l}{\partial h}$ = [1,1,1]

A. [-1, 1, 2]
B. [0, 1, 2]
C. [0, 1, 1] **[x]**
D. [-1, 1, 1]

Question 16

Which is $\frac{\partial l}{\partial W}$ *if* $\overline{h} = xW + b, \frac{\partial l}{\partial \overline{h}}$ = [0,1,1], $x = [1,2,3]$, $b = [0,1,2]$

A) $\begin{bmatrix} 0 & 1 & 1 \\ 0 & 2 & 2 \\ 0 & 3 & 3 \end{bmatrix}$

B) $\begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & 3 \\ 1 & 2 & 3 \end{bmatrix}$

C) $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$

D) $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Question 16

Which is $\frac{\partial l}{\partial W}$ *if* $\overline{h} = xW + b$, $\frac{\partial l}{\partial \overline{h}}$ = [0,1,1], $x = [1,2,3]$, $b = [0,1,2]$

A) $\begin{bmatrix} 0 & 1 & 1 \\ 0 & 2 & 2 \\ 0 & 3 & 3 \end{bmatrix}$ **[x]**

B) $\begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & 3 \\ 1 & 2 & 3 \end{bmatrix}$

C) $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$

D) $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

$$\frac{\partial l}{\partial W} = \frac{\partial l}{\partial \overline{h}} \frac{\partial \overline{h}}{\partial W} = [1\ 2\ 3]^T [0\ 1\ 1]$$

$$= \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} [0\ 1\ 1] = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 2 & 2 \\ 0 & 3 & 3 \end{bmatrix}$$