

FIT 3181/5215 Deep Learning

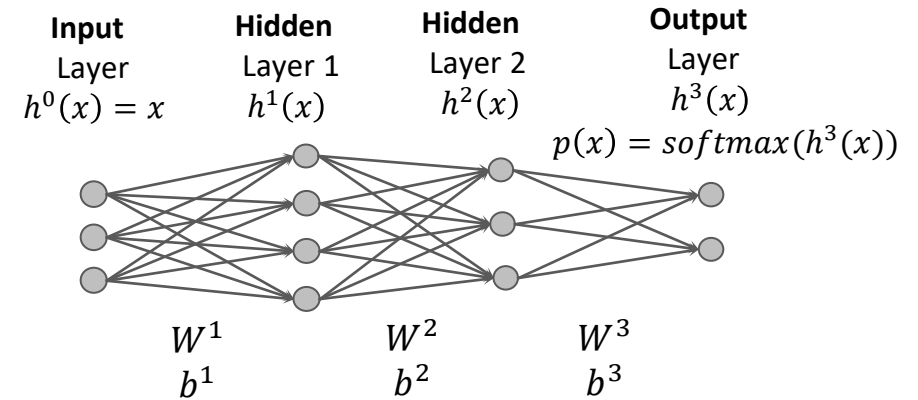
Quiz for:
Feed-forward Neural Nets with TensorFlow

Teaching Team

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

Question 1

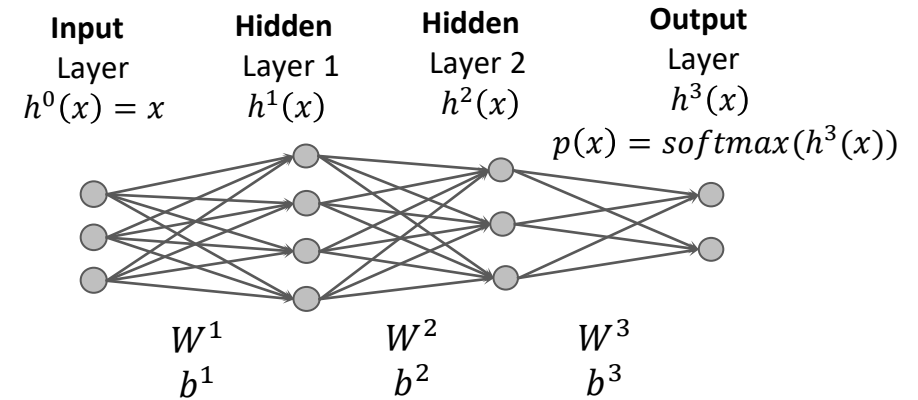
□ Given the following feed-forward neural network. What are the shapes of weight matrices if we follow the convention in the lecture (not TF implementation)?



- A. $W^1 \in \mathbb{R}^{3 \times 3}, W^2 \in \mathbb{R}^{4 \times 4}, W^3 \in \mathbb{R}^{3 \times 3}$
- B. $W^1 \in \mathbb{R}^{4 \times 3}, W^2 \in \mathbb{R}^{3 \times 4}, W^3 \in \mathbb{R}^{2 \times 3}$
- C. $W^1 \in \mathbb{R}^{3 \times 4}, W^2 \in \mathbb{R}^{4 \times 3}, W^3 \in \mathbb{R}^{3 \times 2}$
- D. $W^1 \in \mathbb{R}^{4 \times 4}, W^2 \in \mathbb{R}^{3 \times 3}, W^3 \in \mathbb{R}^{2 \times 2}$

Question 1

□ Given the following feed-forward neural network. What are the shapes of weight matrices if we follow the convention in the lecture (not TF implementation)?

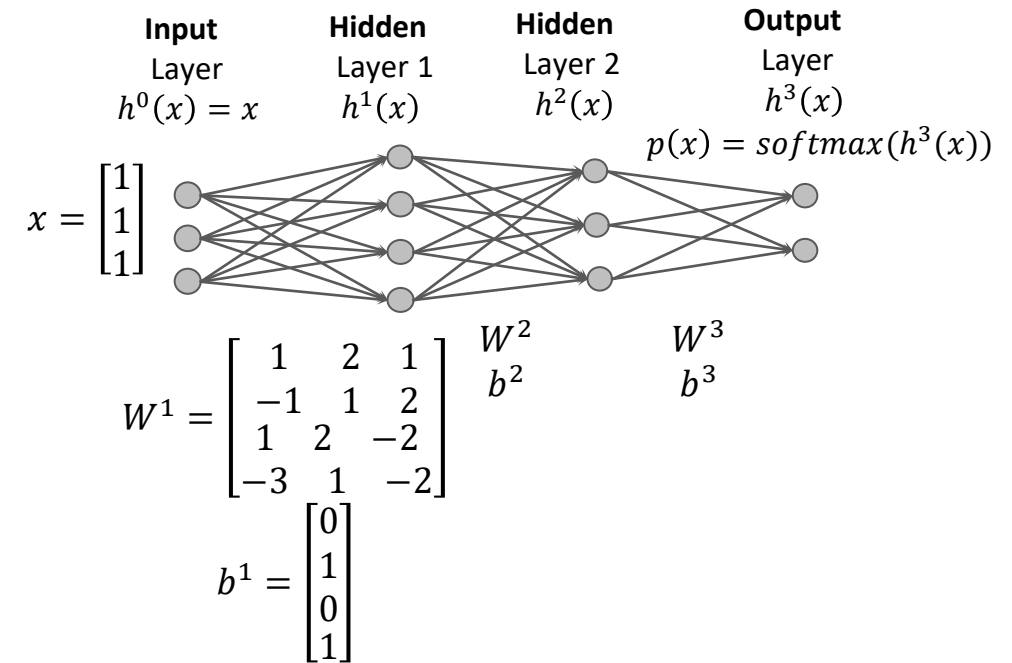


- A. $W^1 \in \mathbb{R}^{3 \times 3}, W^2 \in \mathbb{R}^{4 \times 4}, W^3 \in \mathbb{R}^{3 \times 3}$
- B. $W^1 \in \mathbb{R}^{4 \times 3}, W^2 \in \mathbb{R}^{3 \times 4}, W^3 \in \mathbb{R}^{2 \times 3}$ [x]
- C. $W^1 \in \mathbb{R}^{3 \times 4}, W^2 \in \mathbb{R}^{4 \times 3}, W^3 \in \mathbb{R}^{3 \times 2}$
- D. $W^1 \in \mathbb{R}^{4 \times 4}, W^2 \in \mathbb{R}^{3 \times 3}, W^3 \in \mathbb{R}^{2 \times 2}$

Question 2

□ Given the following feed-forward neural network. Assume that we input to the network feature vector $x = [1 \ 1 \ 1]^T$. What is the values of pre-activations \bar{h}^1 ?

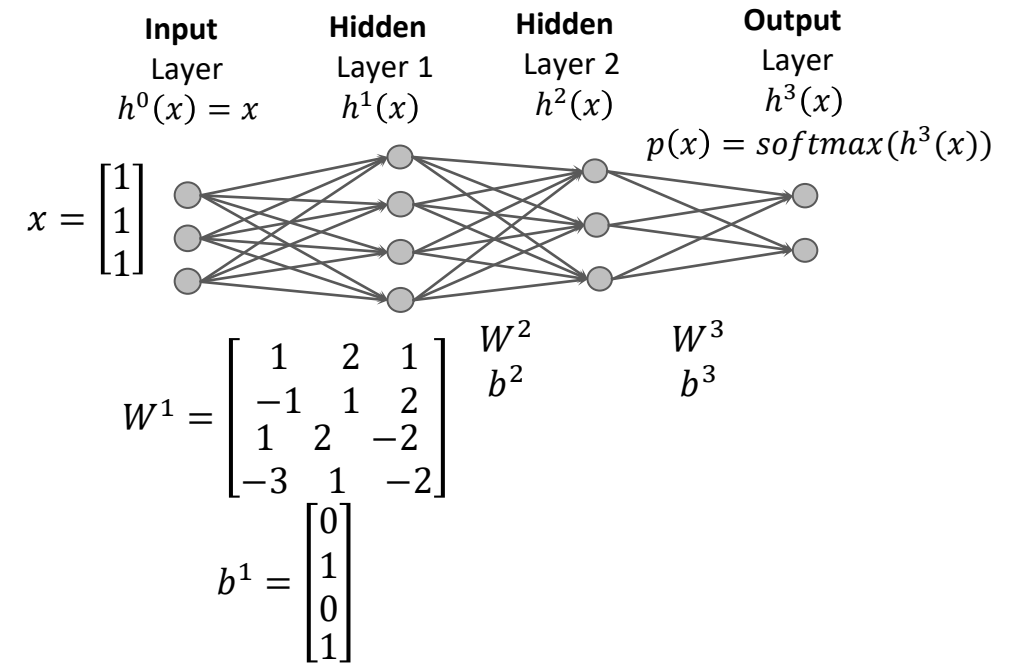
- A. $\bar{h}^1 = [4 \ 2 \ 1 \ -4]$
- B. $\bar{h}^1 = [4 \ 3 \ 1 \ -3]$
- C. $\bar{h}^1 = [4 \ 3 \ 1 \ -3]^T$
- D. $\bar{h}^1 = [4 \ 2 \ 1 \ -4]^T$



Question 2

Given the following feed-forward neural network. Assume that we input to the network feature vector $x = [1 \ 1 \ 1]^T$. What is the values of pre-activations \bar{h}^1 ?

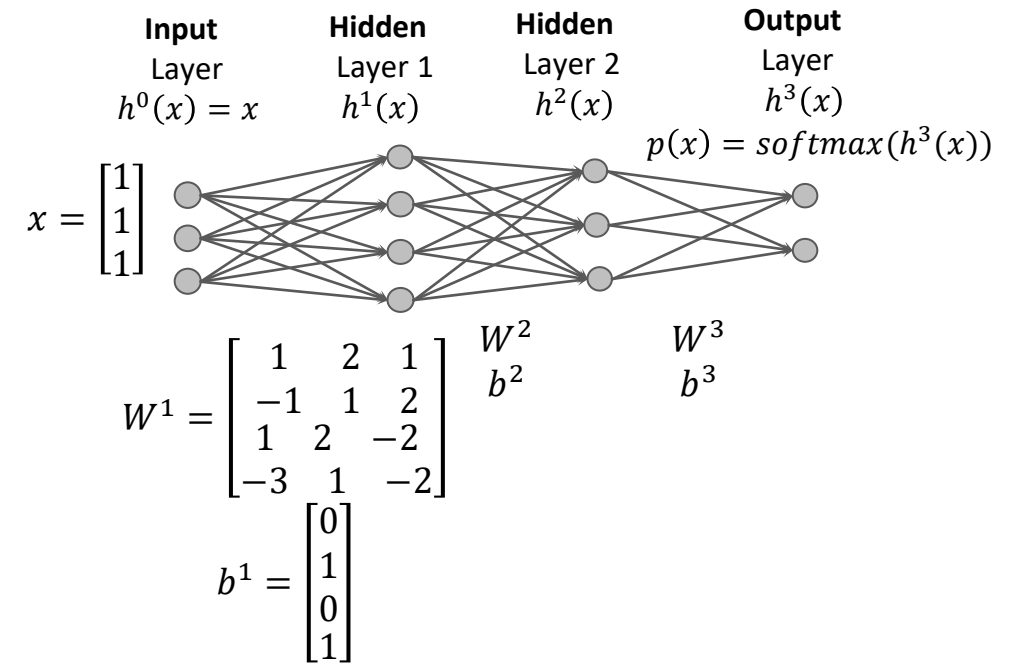
- ☐ A. $\bar{h}^1 = [4 \ 2 \ 1 \ -4]$
- ☐ B. $\bar{h}^1 = [4 \ 3 \ 1 \ -3]$
- ☐ C. $\bar{h}^1 = [4 \ 3 \ 1 \ -3]^T$ [x]
- ☐ D. $\bar{h}^1 = [4 \ 2 \ 1 \ -4]^T$



Question 3

□ Given the following feed-forward neural network. Assume that we input to the network feature vector $x = [1 \ 1 \ 1]^T$. What is the hidden values h^1 if we use ReLU activation function?

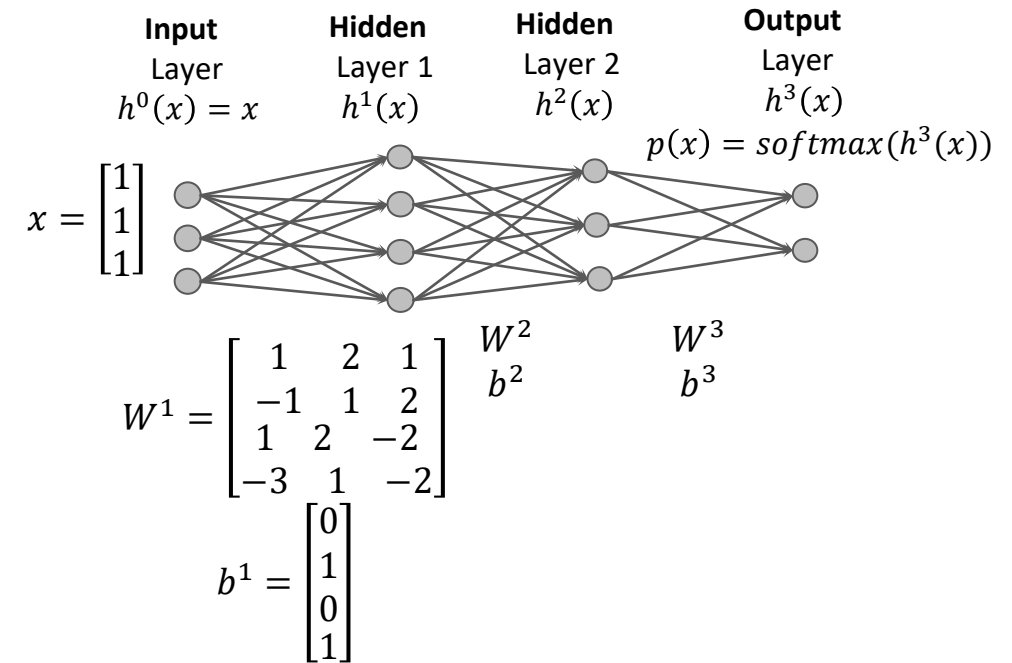
- A. $h^1 = [0 \ 0 \ 0 \ -4]^T$
- B. $h^1 = [4 \ 3 \ 1 \ 0]^T$
- C. $h^1 = [0 \ 0 \ 0 \ -3]^T$
- D. $h^1 = [4 \ 2 \ 1 \ -4]^T$



Question 3

Given the following feed-forward neural network. Assume that we input to the network feature vector $x = [1 \ 1 \ 1]^T$. What is the hidden values h^1 if we use ReLU activation function?

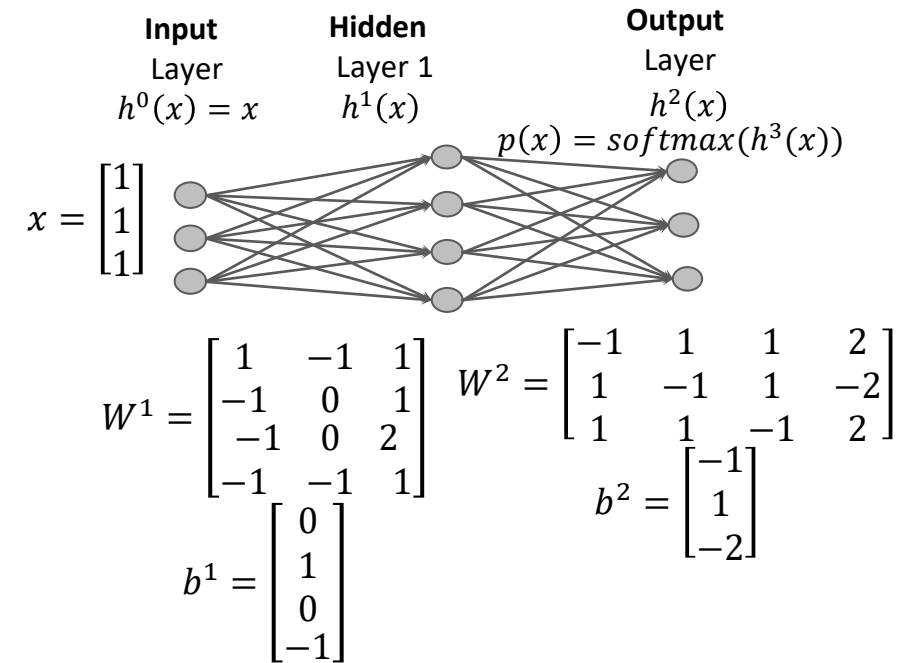
- ☐ A. $h^1 = [0 \ 0 \ 0 \ -4]^T$
- ☒ B. $h^1 = [4 \ 3 \ 1 \ 0]^T$ [x]
- ☐ C. $h^1 = [0 \ 0 \ 0 \ -3]^T$
- ☐ D. $h^1 = [4 \ 2 \ 1 \ -4]^T$



Question 4

Given the following feed-forward neural network. Assume that we input to the network feature vector $x = [1 \ 1 \ 1]^T$. What are the correct statements if we use ReLU activation function? (MC)

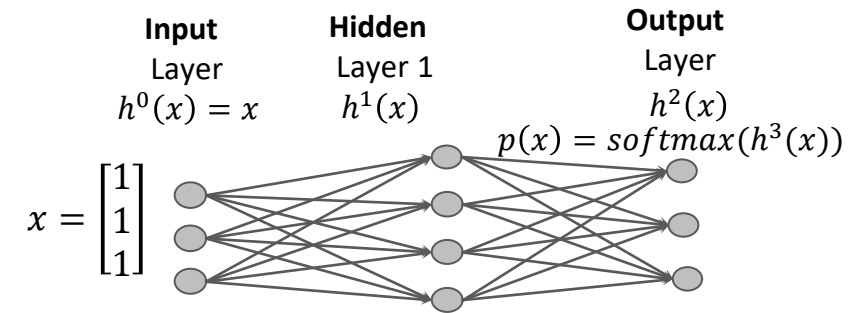
- ☐ A. $h^1 = [1 \ 1 \ 1 \ 0]^T$
- ☐ B. $h^1 = [1 \ 1 \ 1 \ 1]^T$
- ☐ C. Logit $h^2 = [0 \ 2 \ -1]^T$
- ☐ D. Logit $h^2 = [0 \ 2 \ 0]^T$



Question 4

Given the following feed-forward neural network. Assume that we input to the network feature vector $x = [1 \ 1 \ 1]^T$. What are the correct statements if we use ReLU activation function? (MC)

- ☒ A. $h^1 = [1 \ 1 \ 1 \ 0]^T$ [x]
- ☐ B. $h^1 = [1 \ 1 \ 1 \ 1]^T$
- ☒ C. Logit $h^2 = [0 \ 2 \ -1]^T$ [x]
- ☐ D. Logit $h^2 = [0 \ 2 \ 0]^T$



$$\begin{aligned}
 W^1 &= \begin{bmatrix} 1 & -1 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 2 \\ -1 & -1 & 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \\
 + \quad b^1 &= \begin{bmatrix} 0 \\ 1 \\ 0 \\ -1 \end{bmatrix} \\
 W^2 &= \begin{bmatrix} -1 & 1 & 1 & 2 \\ 1 & -1 & 1 & -2 \\ 1 & 1 & -1 & 2 \end{bmatrix} \times \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \\
 + \quad b^2 &= \begin{bmatrix} -1 \\ 1 \\ -2 \end{bmatrix}
 \end{aligned}$$

Question 5

Given an implementation as below. Which of following architecture is correct (SC).

```
: dnn_model = Sequential()  
dnn_model.add(Dense(units=32, input_shape=(784,), activation='relu'))  
dnn_model.add(Dense(units=64, activation='relu'))  
dnn_model.add(Dense(units=64, activation='relu'))  
dnn_model.add(Dense(units=32, activation='relu'))  
dnn_model.add(Dense(units=10, activation='softmax'))
```

- ☐ A. 784 → 32(ReLU) → 64(ReLU) → 64(ReLU) → 32(ReLU) → 10(ReLU)
- ☐ B. 784 → 32(ReLU) → 64(ReLU) → 64(ReLU) → 32(ReLU) → 10(Softmax)
- ☐ C. 32(ReLU) → 64(ReLU) → 64(ReLU) → 32(ReLU) → 10(ReLU)
- ☐ D. 784(ReLU) → 32(ReLU) → 64(ReLU) → 64(ReLU) → 32(ReLU) → 10(Softmax)

Question 5

Given an implementation as below. Which of following architecture is correct (SC).

```
: dnn_model = Sequential()  
dnn_model.add(Dense(units=32, input_shape=(784,), activation='relu'))  
dnn_model.add(Dense(units=64, activation='relu'))  
dnn_model.add(Dense(units=64, activation='relu'))  
dnn_model.add(Dense(units=32, activation='relu'))  
dnn_model.add(Dense(units=10, activation='softmax'))
```

- ☐ A. 784 → 32(ReLU) → 64(ReLU) → 64(ReLU) → 32(ReLU) → 10(ReLU)
- ☒ B. 784 → 32(ReLU) → 64(ReLU) → 64(ReLU) → 32(ReLU) → 10(Softmax) [x]
- ☐ C. 32(ReLU) → 64(ReLU) → 64(ReLU) → 32(ReLU) → 10(ReLU)
- ☐ D. 784(ReLU) → 32(ReLU) → 64(ReLU) → 64(ReLU) → 32(ReLU) → 10(Softmax)

Question 6

Given an implementation as below. Which of following statements are correct (MC).

```
: dnn_model = Sequential()  
  dnn_model.add(Dense(units=32, input_shape=(784,), activation='relu'))  
  dnn_model.add(Dense(units=64, activation='relu'))  
  dnn_model.add(Dense(units=64, activation='relu'))  
  dnn_model.add(Dense(units=32, activation='relu'))  
  dnn_model.add(Dense(units=10, activation='softmax'))
```

- ☐ A. The model has 5 Fully Connected layers
- ☐ B. The batch size is 784
- ☐ C. The model can work with an arbitrary batch size
- ☐ D. The model's output is a logit value and in range $[-\infty, +\infty]$

Question 6

Given an implementation as below. Which of following statements are correct (MC).

```
: dnn_model = Sequential()  
  dnn_model.add(Dense(units=32, input_shape=(784,), activation='relu'))  
  dnn_model.add(Dense(units=64, activation='relu'))  
  dnn_model.add(Dense(units=64, activation='relu'))  
  dnn_model.add(Dense(units=32, activation='relu'))  
  dnn_model.add(Dense(units=10, activation='softmax'))
```

- ☒ A. The model has 5 Fully Connected layers [x]
- ☐ B. The batch size is 784
- ☒ C. The model can work with an arbitrary batch size [x]
- ☐ D. The model's output is a logit value and in range $[-\infty, +\infty]$

the output is a probability

Question 7

Given an implementation as below. What are outputs of the two print functions (SC).

```
[5]: dnn_model = Sequential()  
dnn_model.add(Dense(units=32, input_shape=(784,), activation='relu'))  
dnn_model.add(Dense(units=64, activation='relu'))  
dnn_model.add(Dense(units=64, activation='relu'))  
dnn_model.add(Dense(units=32, activation='relu'))  
dnn_model.add(Dense(units=10, activation='softmax'))
```

```
[8]: hidden1 = dnn_model.layers[0]  
weights, biases = hidden1.get_weights()  
print('shape W=', weights.shape)  
print('shape b=', biases.shape)
```

- ☐ A. (32, 32), (32,)
- ☐ B. (32, 784), (784,)
- ☐ C. (784, 32), (32,)
- ☐ D. (784, 32), (784,)

Question 7

Given an implementation as below. What are outputs of the two print functions (SC).

```
[5]: dnn_model = Sequential()
      dnn_model.add(Dense(units=32, input_shape=(784,), activation='relu'))
      dnn_model.add(Dense(units=64, activation='relu'))
      dnn_model.add(Dense(units=64, activation='relu'))
      dnn_model.add(Dense(units=32, activation='relu'))
      dnn_model.add(Dense(units=10, activation='softmax'))
```

```
[8]: hidden1 = dnn_model.layers[0]
      weights, biases = hidden1.get_weights()
      print('shape W=', weights.shape)
      print('shape b=', biases.shape)
```

- ☐ A. (32, 32), (32,) $Wx + b$
- ☐ B. (32, 784), (784,)
- ☒ C. (784, 32), (32,) [x] why not (32,784) , (32,)? this is tensorflow specific
- ☐ D. (784, 32), (784,)

Question 8

Given an implementation as below. What is the total parameters of the model (SC).

```
dnn_model = Sequential()  
dnn_model.add(Dense(units=20, input_shape=(10,), activation='relu'))  
dnn_model.add(Dense(units=20, activation='relu'))  
dnn_model.add(Dense(units=10, activation='softmax'))
```

- ☐ A. 800
- ☐ B. 830
- ☐ C. 840
- ☐ D. 850

Question 8

Given an implementation as below. What is the total parameters of the model (SC).

```
dnn_model = Sequential()  
dnn_model.add(Dense(units=20, input_shape=(10,), activation='relu'))  
dnn_model.add(Dense(units=20, activation='relu'))  
dnn_model.add(Dense(units=10, activation='softmax'))
```

☐ A. 800

☐ B. 830

☐ C. 840

☒ D. 850 [x]

Architecture **10 → 20 (ReLU) → 20 (ReLU) → 10 (softmax)**
 $(20 \times 10 + 20) + (20 \times 20 + 20) + (10 \times 20 + 10) = 850$