1. Which of the following are true? (Check all that apply.)

1 / 1 point

- [ ] $W^{[1]}$ is a matrix with rows equal to the parameter vectors of the first layer.
- [ ] $w_3^{[4]}$ is the column vector of parameters of the third layer and fourth neuron.
- [x] $W^{[1]}$ is a matrix with rows equal to the transpose of the parameter vectors of the first layer.

  ✓ **Correct**

  Yes. We construct $W^{[1]}$ stacking the parameter vectors $w_j^{[1]}$ of all the neurons of the first layer.

- [x] $w_3^{[4]}$ is the column vector of parameters of the fourth layer and third neuron.

  ✓ **Correct**

  Yes. The vector $w_j^{[i]}$ is the column vector of parameters of the i-th layer and j-th neuron of that layer.

- [ ] W_1 is a matrix with rows equal to the parameter vectors of the first layer.

Expand

✓ **Correct**

Great, you got all the right answers.

2. In which of the following cases is the linear (identity) activation function most likely used?

1 / 1 point

- For binary classification problems.
- When working with regression problems.
- The linear activation function is never used.
- As activation function in the hidden layers.

Expand

Correct

Yes. In problems such as predicting the price of a house it makes sense to use the linear activation function as output.

3. Which of the following is a correct vectorized implementation of forward propagation for layer 2? **1 / 1 point**

- $Z^{[2]} = W^{[2]} A^{[1]} + b^{[2]}$
  $A^{[2]} = g(Z^{[2]})$
- $Z^{[1]} = W^{[1]} X + b^{[1]}$
  $A^{[1]} = g^{[1]}(Z^{[1]})$
- $Z^{[2]} = W^{[2]} X + b^{[2]}$
  $A^{[2]} = g^{[2]}(Z^{[2]})$
- $Z^{[2]} = W^{[2]} A^{[1]} + b^{[2]}$
  $A^{[2]} = g^{[2]}(Z^{[2]})$

Expand

Correct
Yes. The elements of layer two are represented using a superscript in brackets.

4. When building a binary classifier for recognizing cats (y=1) vs raccoons (y=0). Is better to use the sigmoid function as activation function for the hidden layers. True/False

1 / 1 point

- (●) False
- ( ) True

Expand

Correct

Yes. Using tanh almost always works better than the sigmoid function for hidden layers.

5. Consider the following code:

1 / 1 point

```
#+begin_src python

x = np.random.rand(4, 5)

y = np.sum(x, axis=1)

#+end_src
```

What will be y.shape?

- (4,)
- (5,)
- (1, 5)
- (4, 1)

Expand

Correct

Yes. By using axis=1 the sum is computed over each row of the array, thus the resulting array is a column vector with 4 entries. Since the option keepdims was not used the array doesn't keep the second dimension.

6. Suppose you have built a neural network with one hidden layer and tanh as activation function for the hidden layer. You decide to initialize the weights to small random numbers and the biases to zero. The first hidden layer's neurons will perform different computations from each other even in the first iteration. True/False?

1 / 1 point

- False No. Since the weights are most likely different, each neuron will do a different computation.
- True Yes. Since the weights are most likely different, each neuron will do a different computation.

Expand

Correct

7. Using linear activation functions in the hidden layers of a multilayer neural network is equivalent to using a single layer. True/False?

1 / 1 point

- False
- True

Expand

Correct

Yes. When the identity or linear activation function $g(c) = c$ is used the output of composition of layers is equivalent to the computations made by a single layer.

8. You have built a network using the tanh activation for all the hidden units. You initialize the weights to relatively large values, using np.random.randn(..,..)*1000. What will happen?

1 / 1 point

- (●) This will cause the inputs of the tanh to also be very large, thus causing gradients to be close to zero. The optimization algorithm will thus become slow.
- ( ) This will cause the inputs of the tanh to also be very large, causing the units to be "highly activated" and thus speed up learning compared to if the weights had to start from small values.
- ( ) So long as you initialize the weights randomly gradient descent is not affected by whether the weights are large or small.
- ( ) This will cause the inputs of the tanh to also be very large, thus causing gradients to also become large. You therefore have to set α to a very small value to prevent divergence; this will slow down learning.

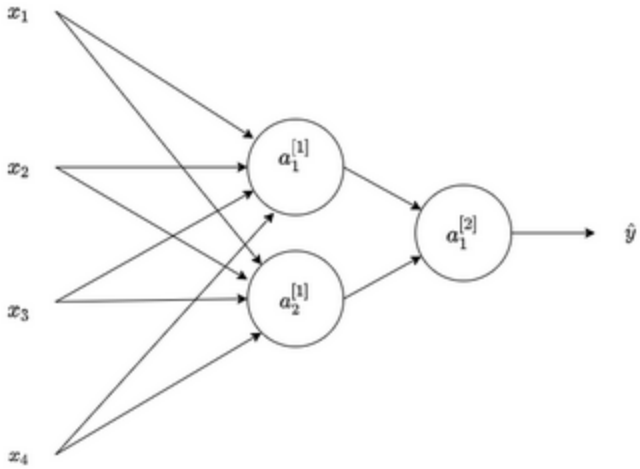
Expand

Correct

Yes. tanh becomes flat for large values; this leads its gradient to be close to zero. This slows down the optimization algorithm.

9. Consider the following 1 hidden layer neural network:

1 / 1 point



Which of the following statements are True? (Check all that apply).

- [ ] $W^{[1]}$ will have shape (4, 2).
- [ ] $W^{[2]}$ will have shape (2, 1)
- [x] $W^{[1]}$ will have shape (2, 4).

  ✓ Correct
  Yes. The number of rows in $W^{[k]}$ is the number of neurons in the k-th layer and the number of columns is the number of inputs of the layer.

- [x] $W^{[2]}$ will have shape (1, 2)

  ✓ Correct
  Yes. The number of rows in $W^{[k]}$ is the number of neurons in the k-th layer and the number of columns is the number of inputs of the layer.

- [ ] $b^{[1]}$ will have shape (4, 2)
- [x] $b^{[1]}$ will have shape (2, 1).

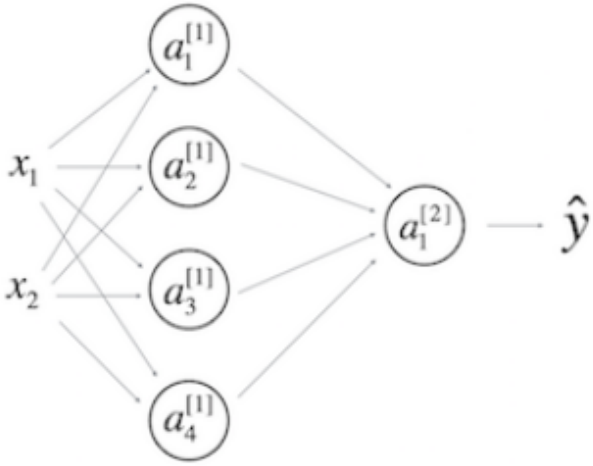
  ✓ Correct
  Yes. $b^{[k]}$ is a column vector and has the same number of rows as neurons in the k-th layer.

Expand

✓ Correct
Great, you got all the right answers.

10. What are the dimensions of $Z^{[1]}$ and $A^{[1]}$?

1 / 1 point



- $Z^{[1]}$ and $A^{[1]}$ are (4,2)
- $Z^{[1]}$ and $A^{[1]}$ are (1,4)
- $Z^{[1]}$ and $A^{[1]}$ are (4,m)
- $Z^{[1]}$ and $A^{[1]}$ are (4,1)

Expand

Correct