Which of the following statements are true? Check all that apply.

1. Any logical function over binary-valued (0 or 1) inputs x_1 and x_2 can be (approximately) represented using some neural network.
2. A two layer (one input layer, one output layer; no hidden layer) neural network can represent the XOR function.
3. The activation values of the hidden units in a neural network, with the sigmoid activation function applied at every layer, are always in the range (0, 1).
4. Suppose you have a multi-class classification problem with three classes, trained with a 3 layer network. Let $a_1^{(3)} = (h_\Theta(x))_1$ be the activation of the first output unit, and similarly $a_2^{(3)} = (h_\Theta(x))_2$ and $a_3^{(3)} = (h_\Theta(x))_3$. Then for any input x, it must be the case that $a_1^{(3)} + a_2^{(3)} + a_3^{(3)} = 1$.

1 and 4

Basic AND, OR, and NOT functions have two layer network.

Consider the following neural network which takes two binary-valued inputs $x_1, x_2 \in \{0, 1\}$ and outputs $h_\Theta(x)$. Which of the following logical functions does it (approximately) compute? -30, 20, 20

1. OR
2. AND

1 because table gives 0111

Consider the neural network given below. Which of the following equations correctly computes the activation $a_1^{(3)}$? Note: $g(z)$ is the sigmoid activation function.

1. $a_3^{(2)} = g(\Theta_{30}^{(1)} x_0 + \Theta_{31}^{(1)} x_1 + \Theta_{32}^{(1)} x_2 + \Theta_{33}^{(1)} x_3)$
2. 1 but with some random changes

1 because this correctly uses the first row of $\Theta^{(2)}$ and includes the "+1" term of $a_0^{(2)}$

You have the following neural network:

$$\begin{bmatrix} +1 \\ x_1 \\ x_2 \end{bmatrix} \rightarrow \begin{bmatrix} . \\ . \\ . \end{bmatrix} \rightarrow [.] \rightarrow h_\Theta(x)$$

You'd like to compute the activations of the hidden layer $a^{(2)} \in \mathbb{R}^3$. One way to do so is the following Octave code:

..

You want to have a vectorized implementation of this (i.e., one that does not use for loops). Which of the following implementations correctly compute $a^{(2)}$? Check all that apply.

1. z = Theta1 * x; a2 = sigmoid (z);
2. a2 = sigmoid (x * Theta1);

3. a2 = sigmoid (Theta2 * x);
4. z = sigmoid(x); a2 = sigmoid (Theta1 * z);

3 because in the lecture's notation $a^{(2)} = g(\Theta^{(1)}x)$, so this version computes it directly, as the sigmoid function will act element-wise.