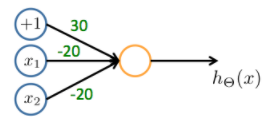
***Neural Networks – Applications***

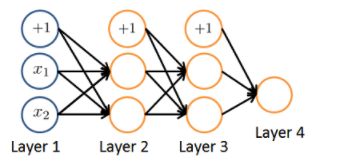
* Which of the following statements are true? Check all that apply.
* **The activation values of the hidden units in a NN, w/ the sigmoid activation function applied at every layer, are always in the range (0, 1).**
* **A 2 layer (1 input, 1 output; no hidden NN cannot represent the XOR function.**
* **Any logical function over binary-valued (0 or 1) inputs x1 and x2 can be (approximately) represented using some neural network.**

.

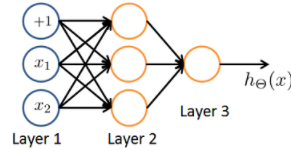
* Consider the following NN which takes 2 binary-valued inputs x1, x2∈{0,1} and outputs hΘ(x). Which of the following logical functions does it (approximately) compute?



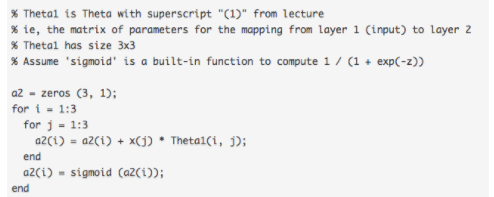
* **NAND (meaning "NOT AND")**
* Consider the NN given below. Which of the following correctly computes the activation a1(3)?



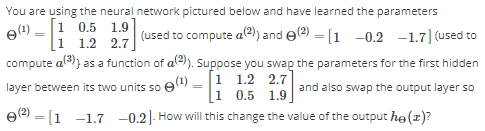
* 
* You have the following neural network:

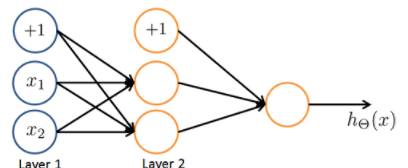


You'd like to compute the activations of the hidden layer a(2) ∈ R3. 1 way to do so is the following Octave code:



* You want to have a *vectorized* implementation of this (i.e., does not use for loops). Which of the following implementations correctly compute a(2)? Check all that apply.
* **a2 = sigmoid (Theta1 \* x);**
* **z = Theta1 \* x; a2 = sigmoid (z);**





* **It will stay the same.**