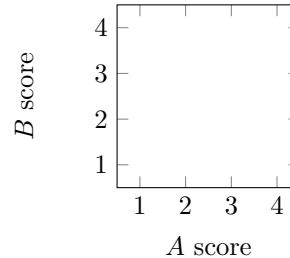


1 Perceptron

You want to predict if movies will be profitable based on their screenplays. You hire two critics A and B to read a script you have and rate it on a scale of 1 to 4. The critics are not perfect; here are five data points including the critics' scores and the performance of the movie:

#	Movie Name	A	B	Profit?
1	Pellet Power	1	1	-
2	Ghosts!	3	2	+
3	Pac is Bac	2	4	+
4	Not a Pizza	3	4	+
5	Endless Maze	2	3	-



1. First, you would like to examine the linear separability of the data. Plot the data on the 2D plane above; label profitable movies with + and non-profitable movies with - and determine if the data are linearly separable.
2. Now you decide to use a perceptron to classify your data. Suppose you directly use the scores given above as features, together with a bias feature. That is $f_0 = 1$, $f_1 = \text{score given by A}$ and $f_2 = \text{score given by B}$.

Run one pass through the data with the perceptron algorithm, filling out the table below. Go through the data points in order, e.g. using data point #1 at step 1.

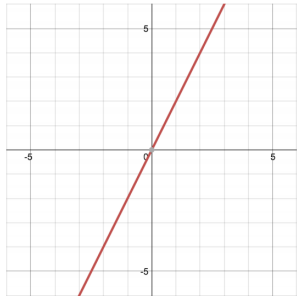
step	Weights	Score	Correct?
1	$[-1, 0, 0]$	$-1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 = -1$	yes
2			
3			
4			
5			

Final weights:

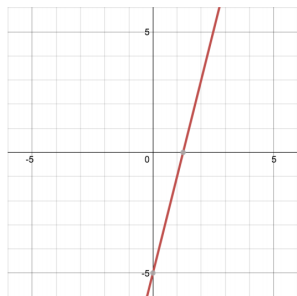
3. Have weights been learned that separate the data?
4. More generally, irrespective of the training data, you want to know if your features are powerful enough to allow you to handle a range of scenarios. Circle the scenarios for which a perceptron using the features above can indeed perfectly classify movies which are profitable according to the given rules:
 - (a) Your reviewers are awesome: if the total of their scores is more than 8, then the movie will definitely be profitable, and otherwise it won't be.
 - (b) Your reviewers are art critics. Your movie will be profitable if and only if each reviewer gives either a score of 2 or a score of 3.
 - (c) Your reviewers have weird but different tastes. Your movie will be profitable if and only if both reviewers agree.

2 Neural Network Representations

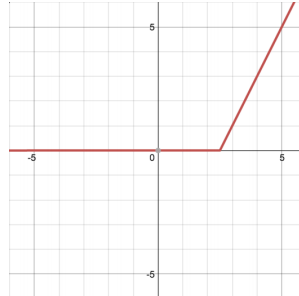
You are given a number of functions (a-h) of a single variable, x , which are graphed below. The computation graphs on the following pages will start off simple and get more complex, building up to neural networks. For each computation graph, indicate which of the functions below they are able to represent.



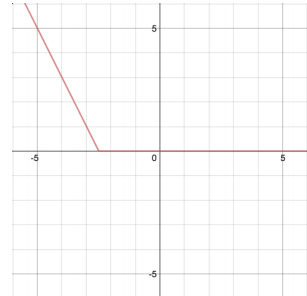
(a) $2x$



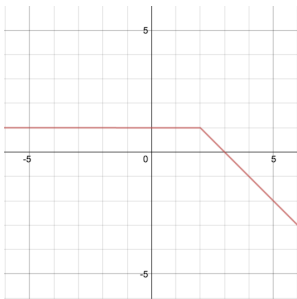
(b) $4x - 5$



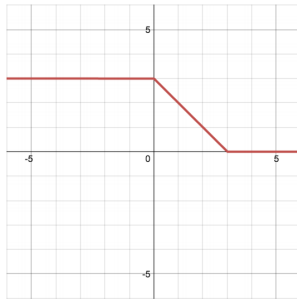
(c) $\begin{cases} 2x - 5 & x \geq 2.5 \\ 0 & x < 2.5 \end{cases}$



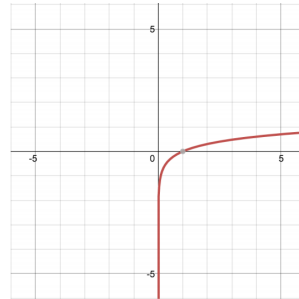
(d) $\begin{cases} -2x - 5 & x \leq -2.5 \\ 0 & x > -2.5 \end{cases}$



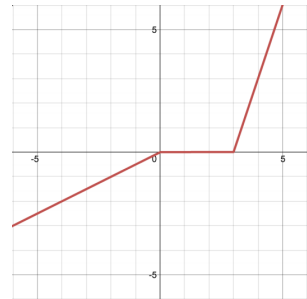
(e) $\begin{cases} -x + 3 & x \geq 2 \\ 1 & x < 2 \end{cases}$



(f) $\begin{cases} 3 & x \leq 0 \\ 3 - x & 0 < x \leq 3 \\ 0 & x > 3 \end{cases}$

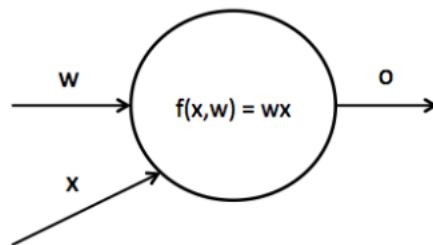


(g) $\log(x)$

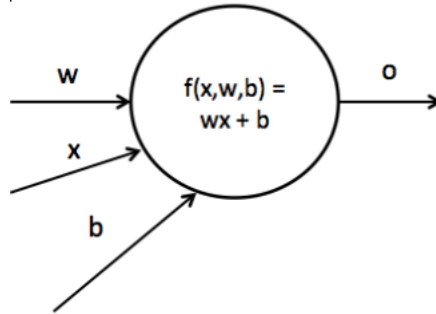


(h) $\begin{cases} 0.5x & x \leq 0 \\ 0 & 0 < x \leq 3 \\ 3x - 9 & x > 3 \end{cases}$

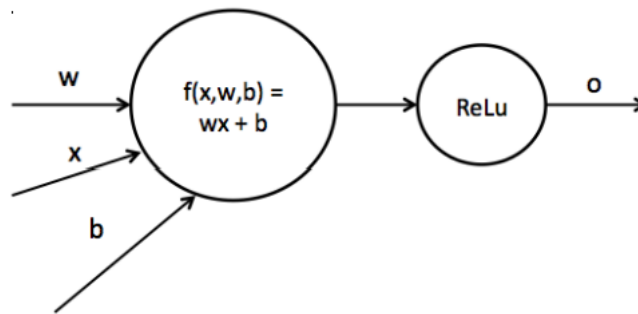
1. Consider the following computation graph, computing a linear transformation with scalar input x , weight w , and output o , such that $o = wx$. Which of the functions can be represented by this graph? For the options which can, write out the appropriate value of w .



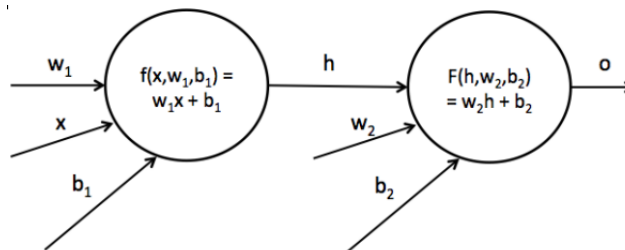
2. Now we introduce a bias term b into the graph, such that $o = wx + b$ (this is known as an *affine* function). Which of the functions can be represented by this network? For the options which can, write out an appropriate value of w, b .



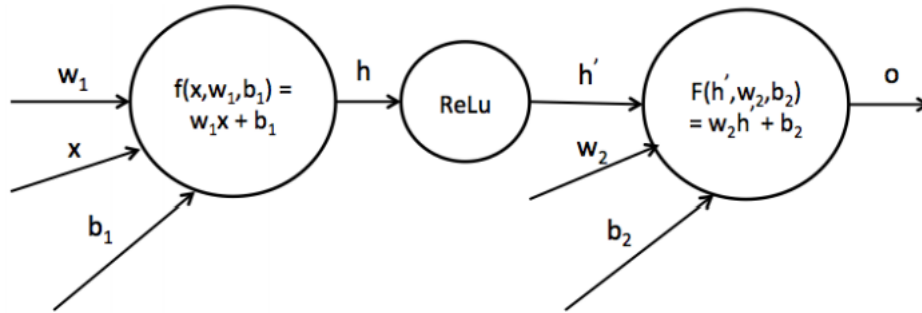
3. We can introduce a non-linearity into the network as indicated below. We use the ReLU non-linearity, which has the form $ReLU(x) = \max(0, x)$. Now which of the functions can be represented by this neural network with weight w and bias b ? For the options which can, write out an appropriate value of w, b .



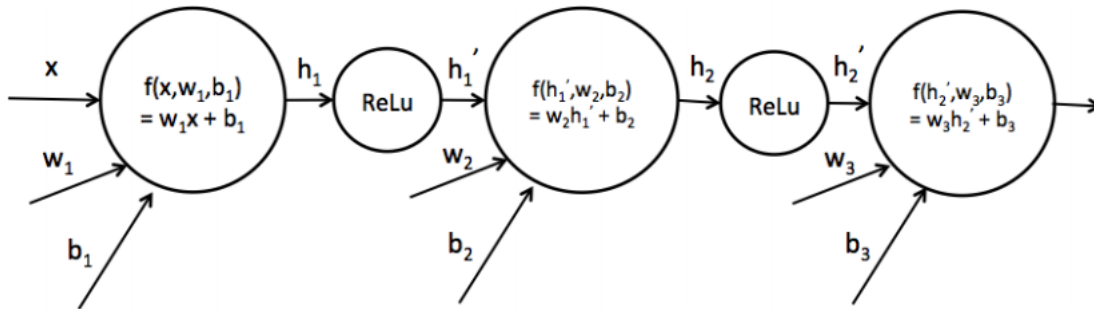
4. Now we consider neural networks with multiple affine transformations, as indicated below. We now have two sets of weights and biases w_1, b_1 and w_2, b_2 . We denote the result of the first transformation h such that $h = w_1x + b_1$, and $o = w_2h + b_2$. Which of the functions can be represented by this network? For the options which can, write out appropriate values of w_1, w_2, b_1, b_2 .



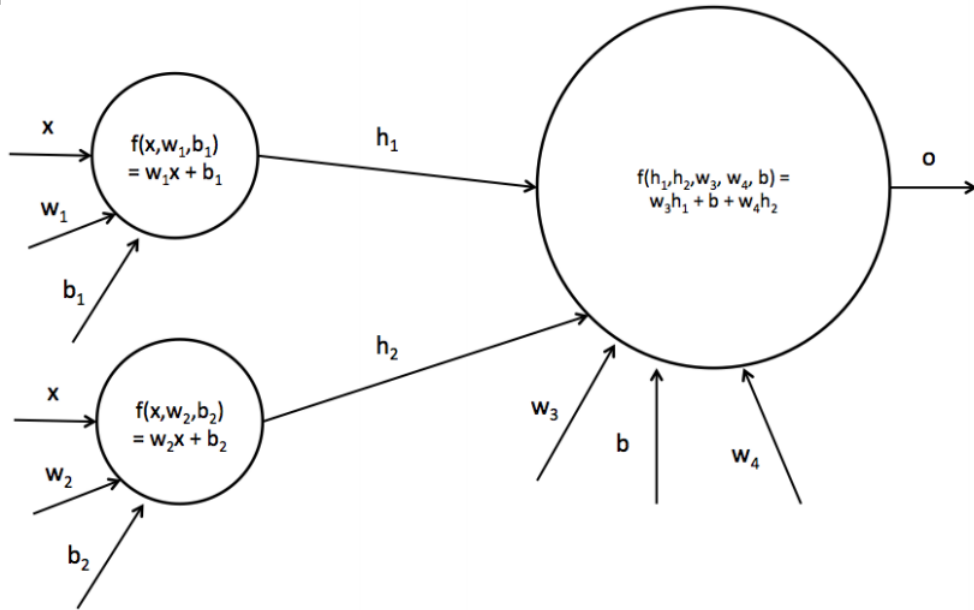
5. Next we add a ReLU non-linearity to the network after the first affine transformation, creating a hidden layer. Which of the functions can be represented by this network? For the options which can, write out appropriate values of w_1, w_2, b_1, b_2 .



6. Now we add another hidden layer to the network, as indicated below. Which of the functions can be represented by this network?



7. We'd like to consider using a neural net with just one hidden layer, but have it be larger – a hidden layer of size 2. Let's first consider using just two affine functions, with no nonlinearity in between. Which of the functions can be represented by this network?



8. Now we'll add a non-linearity between the two affine layers, to produce the neural network below with a hidden layer of size 2. Which of the functions can be represented by this network?

