ECE/CS 559 - Fall 2016 - Midterm Part #2.

Full Name: **ID Number:**

In the following, $\log(\cdot)$ is the natural logarithm, e is the base of the natural logarithm, $|\cdot|$ is the absolute value ($|x| = x$ if $x \geq 0$, and $|x| = -x$ if $x < 0$), and $\mathbb{R}$ is the set of real numbers. Recall $\frac{\partial e^x}{\partial x} = e^x$, and $e^{\log x} = x$, $x > 0$.

- **Q3 (25 pts):** Consider the activation function

$$f(v) = \begin{cases} 1 - e^{-v}, & v \geq 0, \\ -1 + e^{-|v|}, & v < 0. \end{cases}$$

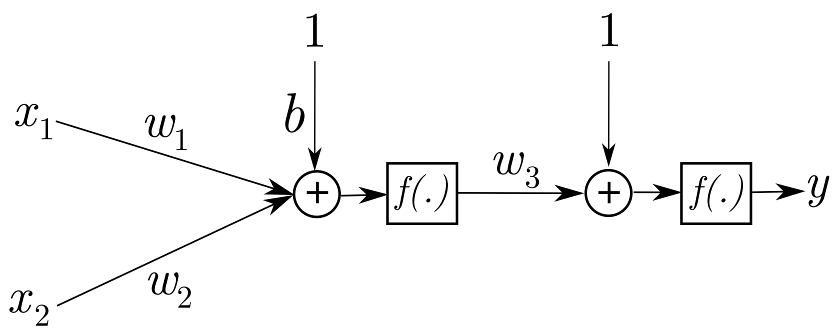
For (a)-(c), consider a single-neuron network with input-output relationship $y = f(b + \mathbf{w}^T\mathbf{x})$, where y is the network output, b is the bias term, $\mathbf{w} = \left[\begin{smallmatrix} w_1 \\ w_2 \end{smallmatrix}\right]$ are the synaptic weights, and $\mathbf{x} = \left[\begin{smallmatrix} x_1 \\ x_2 \end{smallmatrix}\right]$ is the network input.

(a) **(3 pts):** Calculate the derivative $f'(v)$ in closed form.

(b) **(7 pts):** Let $E = (d - y)^2$, where d is a constant (a generic desired output). Find the delta-learning rule (the gradient-descent update equations) for b, w_1, w_2 given learning parameter $\eta = \frac{1}{2}$. Remember that you can write a single (vectorized) update equation that can handle all variables. The final update expression(s) may however contain only the terms/functions $d, y, f', f, \mathbf{w}, w_1, w_2, b$.

(c) **(6 pts):** Consider the same delta-learning setup as in (b). Consider the training vectors $\mathbf{x}_1 = \left[\begin{smallmatrix} \log 2 \\ \log 3 \end{smallmatrix}\right]$, $\mathbf{x}_2 = \left[\begin{smallmatrix} 0 \\ \log 2 \end{smallmatrix}\right]$, with desired outputs $d_1 = \frac{2}{3}$, $d_2 = \frac{5}{2}$, respectively. Find the updated bias and the updated synaptic weights after one epoch of online learning given initial conditions $b = 0$, $\mathbf{w} = \left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right]$.

(d) **(9 pts)** Consider now a multi-layer network as shown below.



Let $E = e^{(d-y)^4}$. Find the gradient-descent update equations for b, w_1, w_2, w_3 given learning parameter $\eta = \frac{1}{4}$. The final expression may only contain the terms/functions $d, y, f', f, \mathbf{w}, w_1, w_2, b, w_3$.
Hint: You do not have to apply some algorithm. Just write the input-output relationship of the network, and use chain rule. This is more of a calculus problem than a neural network problem!

- **Q4 (25 pts):** Design a (possibly multi-layer) neural network with two inputs $x_1, x_2 \in \mathbb{R}$ and a single output y such that $y = 1$ if $x_1 = 3$ and $x_2 = 2$, and $y = 0$, otherwise. In other words, the network output should assume a value of 1 only at the single point $(x_1, x_2) = (3, 2)$ of the input space $\mathbb{R}^2$. Note that the inputs can be any real numbers. The neuron(s) of the network should use the step-activation function $u : \mathbb{R} \to \{0, 1\}$ given by $u(v) = 1$ if $v \geq 0$, and $u(v) = 0$, otherwise. The network should consist of **5 neurons at most**. Spoilers and their partial credits:

(a) **(3 pts):** Design a network with input x_1 and output y such that $y = 1$ if $x_1 \geq 3$, and $y = 0$, otherwise.

(b) **(4 pts):** Design a network with input x_1 and output y such that $y = 1$ if $x_1 \leq 3$, and $y = 0$, otherwise.

(c) **(6 pts):** Design a network with input x_1 and output y such that $y = 1$ if $x_1 = 3$, and $y = 0$, otherwise.

(d) **(6 pts):** Solve the original problem without any restrictions on the number of neurons.

(e) **(6 pts):** Solve the original problem.

You are not required to do (a)-(d) provided you can provide a correct solution to (e).