CS5102: Deep Learning
Mid-Term 1 Exam
Fall 2018

Student's ID: _______________

Instructions:

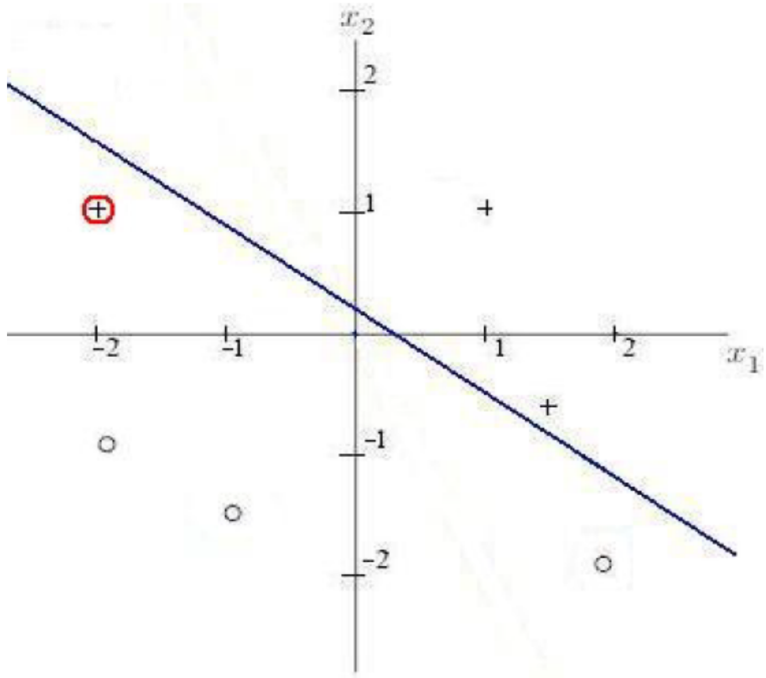
1: This is an open-book exam. You can use any books, helping material and calculators

2: The exam contains three(3) problems. Please record your response to each problem in the space provided.

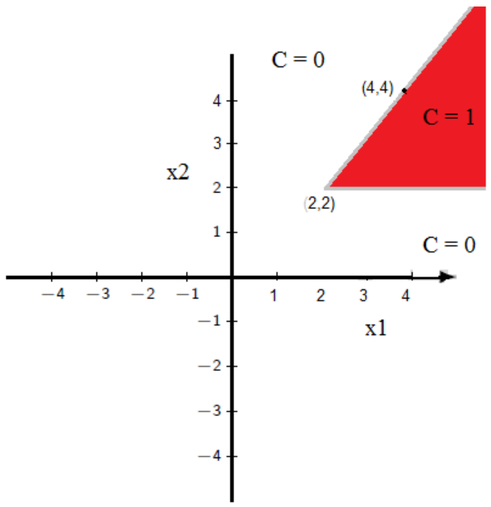
3: Please show all your work.

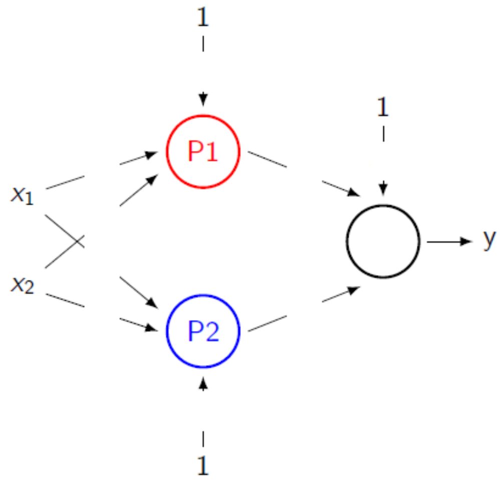
4: All questions are explained clearly. If you find some question's statement ambiguous, you can make reasonable assumptions as long as you state them clearly.

Good luck!



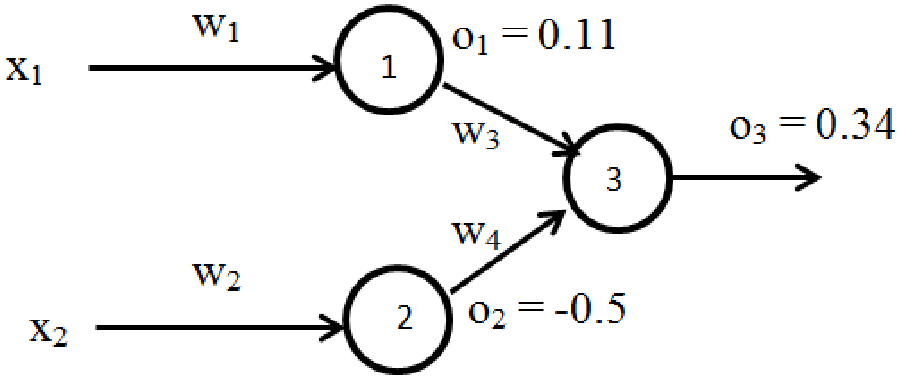
- Problem 1: (40 points, 20+20)
  (a): A single perceptron with two inputs x_1 and x_2 is trained using the data point indicated in figure. Using initial weight vector $w = [-0.2, 0.6, 0.9]^T$, update the weights using the data point (-2,1) and $\eta = 0.5$.





(b): We need to come up with a multilayer perceptron that implements the classification outlined in the accompanying figure. We want to accomplish this task using only 3 perceptron units with the architecture shown above. Each perceptron uses a thresholding activation function.

Find the weights for each connection that would implement this classification boundary.



- Problem 2: (40 points, 20+20)

  Shown in the accompanying figure is a multilayer perceptron we wish to train using back propagation. The outputs of each layer as the result of forward pass are also indicated as o_1, o_2 etc. The perceptrons implement a sigmoidal activation function. With the target value $y = 0.75, \eta = 0.2$:

  (a): Apply the back-propagation algorithm to calculate $\Delta w_1 = -\eta \frac{dE}{dw_1}$

(b):

For the neural network shown above, instead of minimizing the cost function $E(w) = \frac{1}{2}(y-o)^2$ as we were doing before, we want to minimize $E(w) = \frac{1}{2}(y-o)^2 + \sum w_i^2$ using back-propagation, where w_i are the weights in the network. Find Δw_1 with this modified error function.

- Problem 3: (20 points)

  For a linear regression problem, we are given data samples x (n samples, each with d features) and labels y. We are trying to fit the parameters θ to the data as:

$$\begin{bmatrix} --x_1-- \\ --x_2-- \\ \vdots \\ --x_n-- \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_d \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

  What is the constraint on n and d to avoid over-fitting.

  Further, we are given that parameters θ repeat themselves after m values, such that $\theta_1 = \theta_{m+1}, \theta_2 = \theta_{m+2}....$ What is the constraint now to avoid over-fitting.