LUMS
A Not-for-Profit University

Department of Electrical Engineering
School of Science and Engineering

EE514/CS535 Machine Learning

HOMEWORK 3

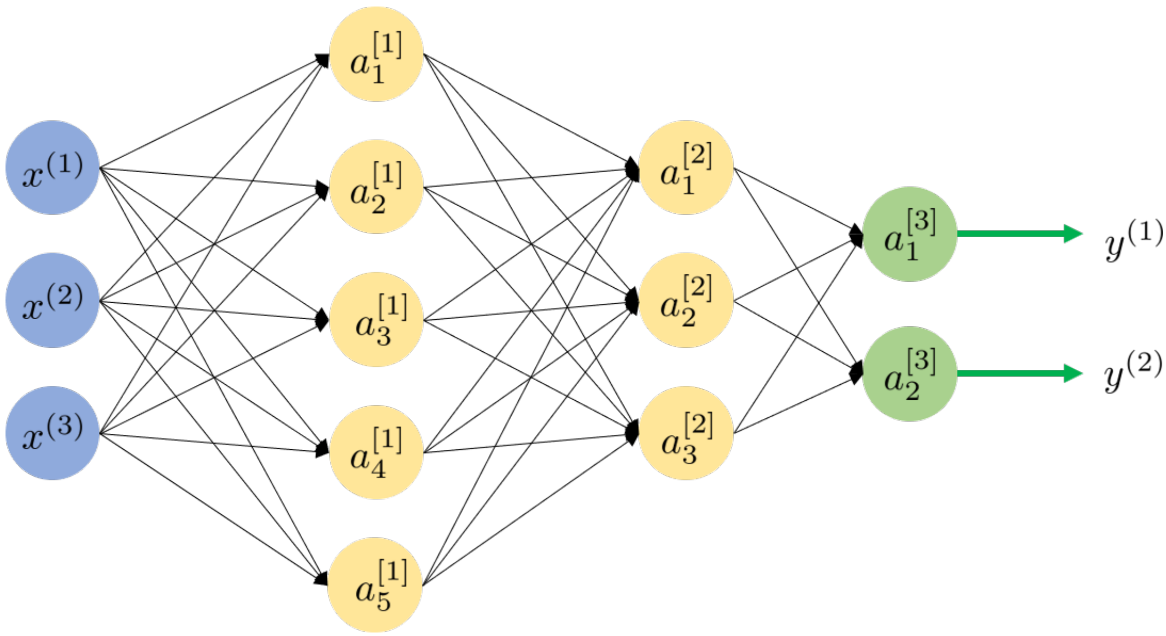
Due Date: 23:55, Saturday, May 07, 2022 (Submit online on LMS)
Format: 6 problems, for a total of 100 marks

Instructions:

- Each student must submit his/her own hand-written assignment, scanned in a single PDF document.
- You are allowed to collaborate with your peers but copying other's solution is strictly prohibited. Anybody found guilty would be subjected to disciplinary action in accordance with the university rules and regulations.
- Note: Vectors are written in lowercase and bold in the homework. For your written submission, kindly use an underline instead. In addition, use capital letters for matrices and lowercase for scalars.

Problem 1 (30 marks)

Consider a neural network shown in the figure below. We have three inputs and two outputs. The activation function used in hidden layer is the ReLU function and no activation function is used at the output layer. The loss function chosen is given by: $J = \frac{1}{2}|\boldsymbol{y} - \hat{\boldsymbol{y}}|$, where $\boldsymbol{y}$ is the true training label vector, and $\hat{\boldsymbol{y}}$ is the predicted output vector. You are given the following transformations:



(a) **[3 marks]** Define the weight matrices and bias vectors for each layer using the appropriate notation and specify their sizes.

(b) **[5 marks]** Formulate a set of equations for Forward pass.

(c) **[2 marks]** Calculate the total number of trainable parameters of the neural network.

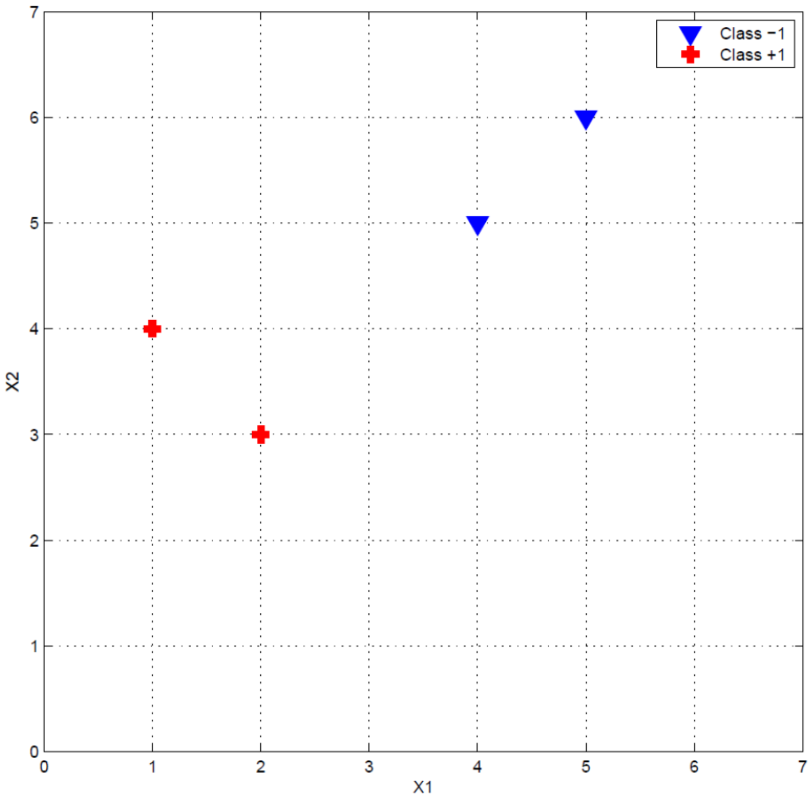
(d) **[5 marks]** Derive $\frac{\partial J}{\partial W_{ij}^{[3]}}$

(e) **[5 marks]** Now write $\frac{\partial J}{\partial W^{[3]}}$, i.e., use vector products.

(f) **[10 marks]** Using your result from previous parts, derive $\frac{\partial J}{\partial W_{ij}^{[2]}}$

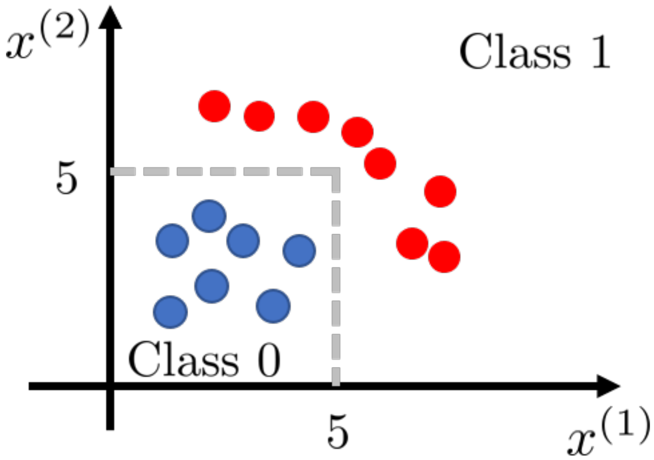
Problem 2 (10 marks)

For the training data plotted below, find the weight vector and bias for the decision boundary $\mathbf{w}^T\mathbf{x} - \theta = 0$ maximizing the classification margin. Also, indicate the support vectors and compute the classification margin.



Problem 3 (15 marks)

For the binary classification problem with two inputs and one output depicted in the figure below, we want to design a neural network with decision boundary indicated by the dashed line.



(a) **[6 marks]** Design a single sigmoid neuron, that is, determine weights and bias, such that the the decision boundary is $x^{(1)} = 5$. Now you are trained to build a network for the problem under consideration.

(b) **[9 marks]** Design a neural network with the dashed line indicated in the figure as its (approximate) decision boundary. You must draw a neural network indicating inputs, output, weights and biases and provide a brief explanation of your design.

Problem 4 (20 marks)

Consider a binary classification problem with two inputs and the following labeled data-set for training.

Label y	Data Point $(x^{(1)}, x^{(2)})$
1	(-2, -2)
1	(-2, 2)
1	(2, 2)
-1	(1, 1)
-1	(1, -1)
-1	(-1, 1)

Table 1: Data points

(a) [**3 marks**] Plot the points on a 2D plane. Can we use hard SVM for this problem? Provide a brief justification to support your answer.

(b) [**6 marks**] Since the data is not linearly separable, we map the 2D feature space to 3D feature space using the mapping function $\boldsymbol{\phi}(\mathbf{x})$ to make it linearly separable. Determine the mapping function that can enable us to use hard SVM in 3D feature space.

(c) [**5 marks**] We have a linear decision boundary (hard SVM) in 3D space to separate the transformed data in 3D (new feature space). Indicate this boundary as a (non-linear) decision boundary on the plot obtained in part (a).

(d) [**6 marks**] Instead of mapping the data into 3D space and using hard SVM to learn the decision boundary in 3D, we can use the kernel trick to learn a non-linear boundary you have plotted in part(c) in the original 2D feature space. Formulate a kernel function associated with the mapping function you used in part (b).

Problem 5 (15 marks)

In this question we will dry-run the k-means clustering algorithm on a two dimensional data-set. The data-set is provided in Table below.

#	Data Point
1	(59, 32)
2	(46, 29)
3	(62, 28)
4	(47, 32)
5	(55, 42)
6	(50, 30)
7	(49, 31)
8	(67, 31)
9	(51, 38)
10	(60, 30)

Table 2: Data points for unsupervised learning

Run the k-means algorithm for $k = 3$ on this data-set for two iterations. Take $\mu_1 = (62, 32)$, $\mu_2 = (66, 37)$ and $\mu_3 = (65, 30)$ as your initial centroids for k = 1, 2, and 3 respectively. Use Euclidean distance where required. Associate each data point with it's corresponding k-number.

Problem 6 (10 marks)

For the data-set given in Table 1, carry out agglomerative clustering to obtain a set of nested clusters and a dendrogram. Use centroid linkage for the merging of the clusters.

#	Data Point
1	(-2, -2)
2	(-2, 0)
3	(1, 3)
4	(2, 2)
5	(3, 4)

Table 5: Data points for agglomerative clustering

— End of Homework —