EE514 & CS535 - Machine Learning
Final Examination Spring 2021
Part - 2 (50 pts)

Note:
There are seven questions worth 60 points. We require you to attempt questions with cumulative worth of 50 points. Do not over attempt; we will not grade the one with the highest marks.

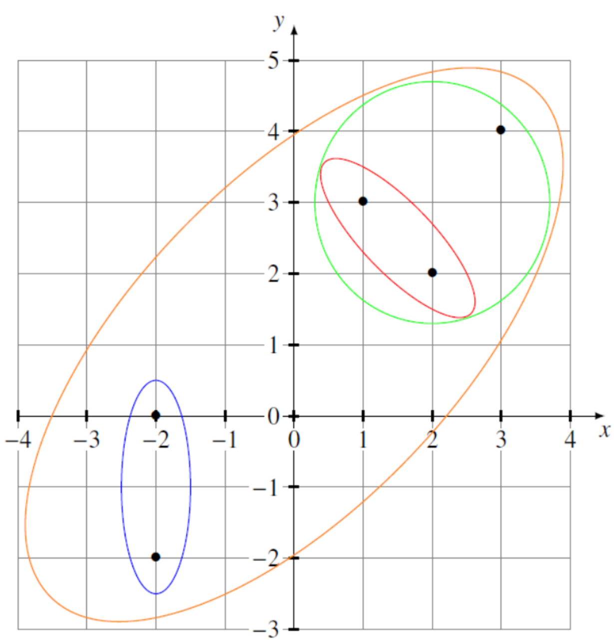
Problem 1. (**10 pts**)

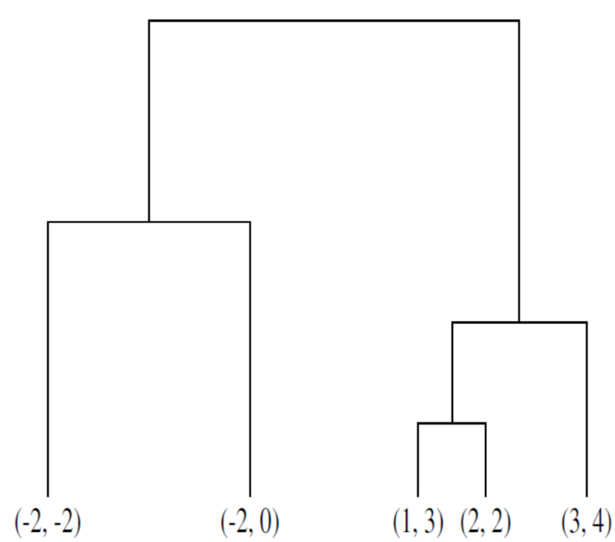
For the data-set given in Table 1, carry out agglomerative clustering to obtain a set of nested clusters and a dendrogram. Use centroid linkage for the merging of the clusters.

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

Table 1: Data points for agglomerative clustering

Solutions:
Nested Clusters and Associated Dendrogram:





Problem 2. (**10 pts**) This problem is related to a perceptron classifier with d number of input features (inputs) and a binary output.

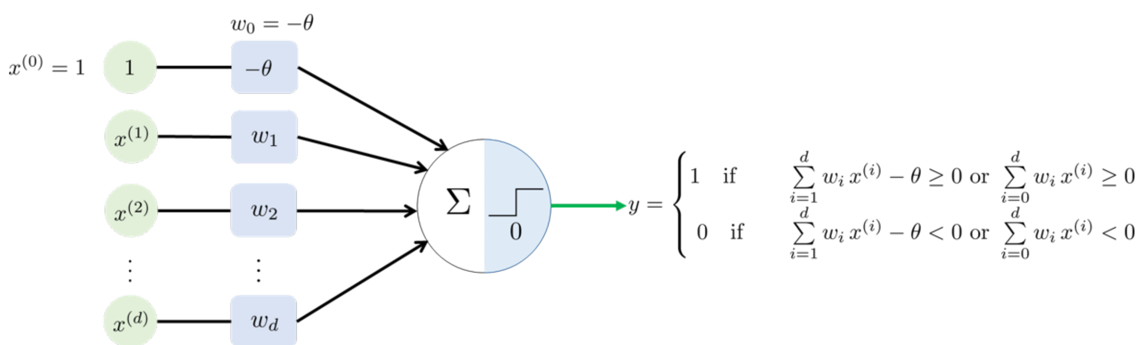
(a) (**2 pts**) Provide the mathematical formulation and a diagram of the perceptron classifier. Indicate each term in your formulation on the diagram.

(b) (**5 pts**) How do we learn the parameters of the perceptron classifier using the training data? Provide pseudo-code of the the perceptron learning algorithm.

(c) (**3 pts**) We have a theorem (proof of convergence) associated with the perceptron learning algorithm. Briefly explain this theorem, i.e., state the assumptions for the convergence of learning algorithm and the speed of convergence.

Solutions:

(a) Perceptron Model:



(b) Learning Algorithm Pseudo-code:

Initialize $\mathbf{w} = 0$

while TRUE do

  $m = 0$

  for $(\mathbf{x}_i, y_i) \in \mathcal{D}$ **do**

    if $y_i(\mathbf{w}^T\mathbf{x}_i) \leq 0$

      $\mathbf{w} \leftarrow \mathbf{w} + y_i\mathbf{x}_i$

      $m \leftarrow m + 1$

    end if

  end for

  if $m = 0$

    break

  end if

end while

(c) Assumptions and Convergence Theorem:

Assumptions:

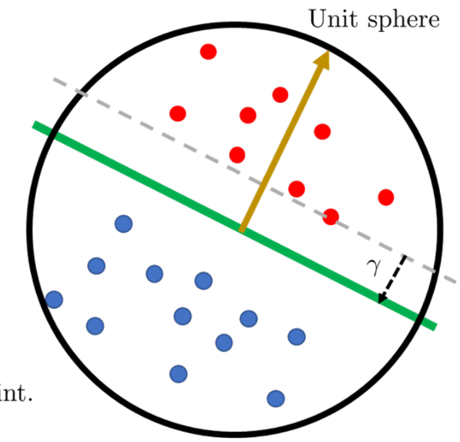
- Data is lineary separable: $\exists\, \mathbf{w}^*$ such that $y_i(\mathbf{x}_i^T \mathbf{w}^*) > 0\ \forall (\mathbf{x}_i, y_i) \in D$.
- We rescale each data point and the $\mathbf{w}^*$ such that

$$||\mathbf{w}^*|| = 1 \qquad \text{and} \qquad ||\mathbf{x}_i|| \leq 1 \quad i = 1, 2, \ldots, n$$

  - All inputs $\mathbf{x_i}$ live within the unit sphere
  - $\mathbf{w}^*$ lies on the unit sphere
- We define the margin of a hyper-plane, denoted by γ, as

$$\gamma = \min_{(\mathbf{x}_i, y_i) \in D} |\mathbf{x}_i^\top \mathbf{w}^*|$$

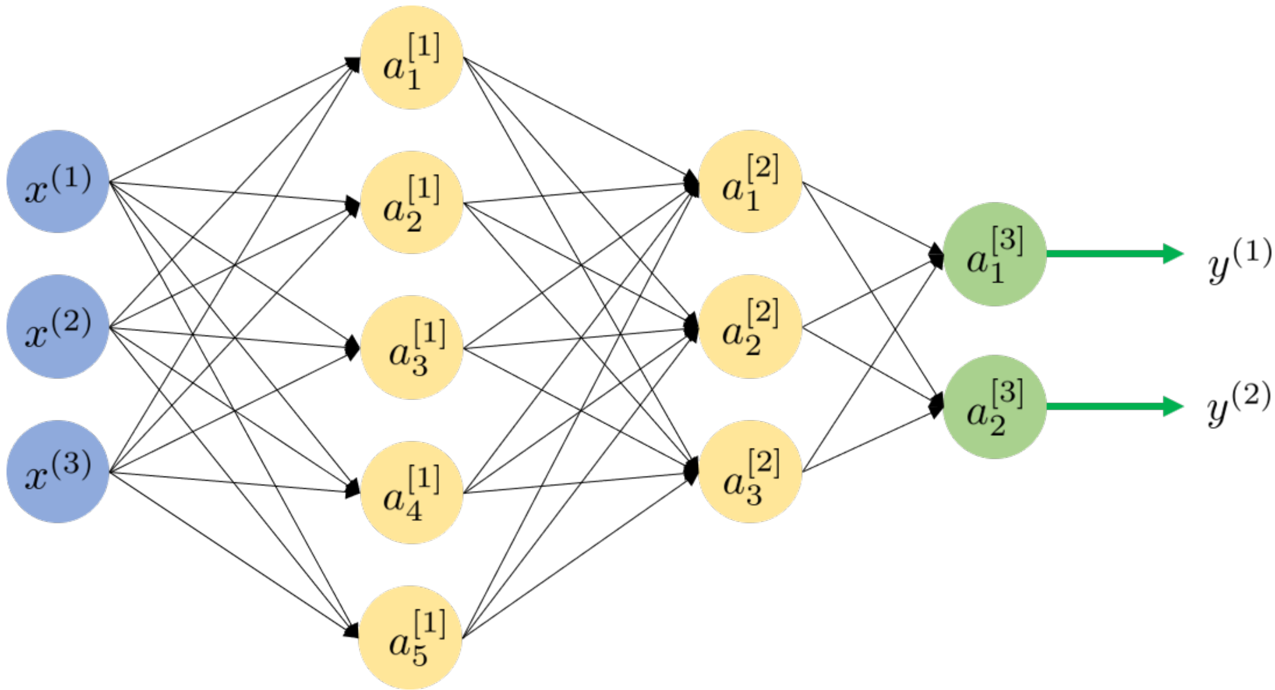
.
  - γ is the distance from the hyperplane to the closest data point.



Theorem: Under these assumptions, the perceptron algorithm makes at most $1/\gamma^2$ misclassifications.

Interpretation: The number of updates is equal to the number of misclassifications!

Problem 3. (**10 pts**) Consider a neural network shown in the figure below. We have three inputs and two outputs.



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

(b) (**5 pts**) Formulate a set of equations for Forward pass.

(c) (**2 pts**) Calculate the total number of trainable parameters of the neural network.

Solutions:

Layer 1 Parameters: $\mathbf{W}^{[1]}$ – 5×3, $\mathbf{b}^{[1]}$ – 5×1

Layer 2 Parameters: $\mathbf{W}^{[2]}$ – 3×5, $\mathbf{b}^{[2]}$ – 3×1

Layer 2 Parameters: $\mathbf{W}^{[3]}$ – 2×3, $\mathbf{b}^{[3]}$ – 2×1

Forward-pass Equations:

$$\mathbf{a}^{[1]} = g(\mathbf{z}^{[1]}), \quad \mathbf{z}^{[1]} = \mathbf{W}^{[1]}\mathbf{x} + \mathbf{b}^{[1]}$$

$$\mathbf{a}^{[2]} = g(\mathbf{z}^{[2]}), \quad \mathbf{z}^{[2]} = \mathbf{W}^{[2]}\mathbf{a}^{[1]} + \mathbf{b}^{[2]}$$

$$\mathbf{a}^{[3]} = g(\mathbf{z}^{[3]}), \quad \mathbf{z}^{[3]} = \mathbf{W}^{[3]}\mathbf{a}^{[2]} + \mathbf{b}^{[3]}$$

Number of parameters: $(5 \times 3) + 5 + (3 \times 5) + 3 + (2 \times 3) + 2 = 46$

Problem 4. (**5 pts**) SVM is inherently defined for binary classification problems. For an M class multi-class classification problem, build a one-vs-rest (one-vs-all) classifier using M number of binary SVM classifiers. We only require you to briefly explain on the training of each classifier and the prediction for a new test-point.

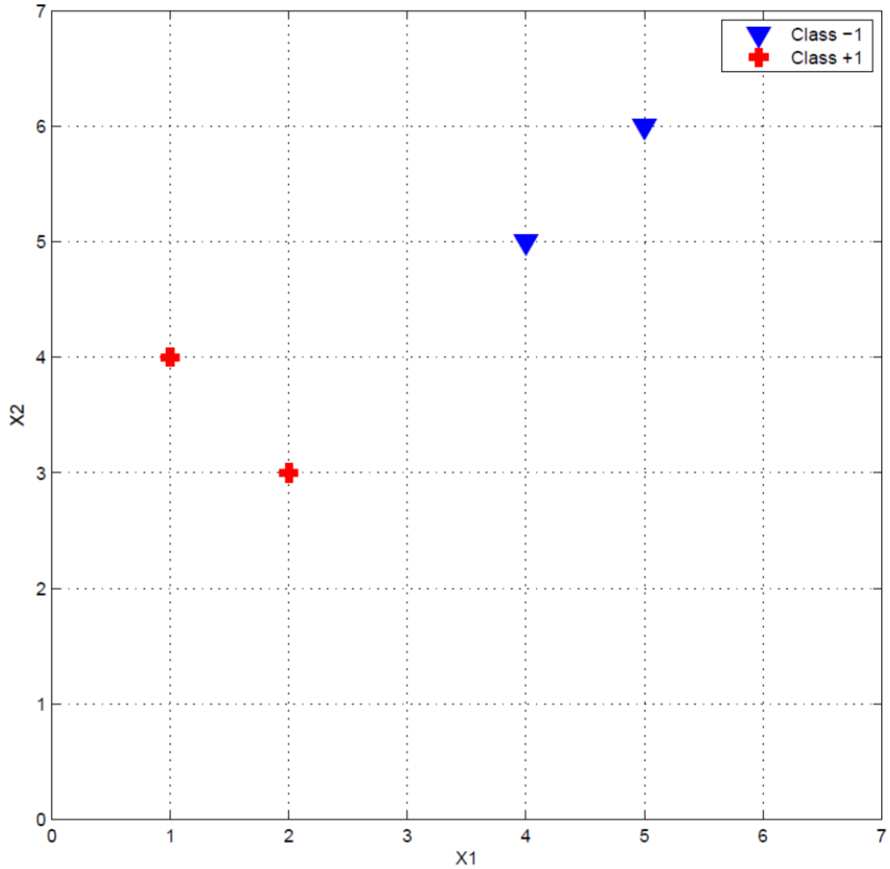
Solutions:

- $\mathcal{Y} = \{0, 1, 2, \ldots, M-1\}$ (M-class classification)

Build a one-vs-rest (OvR) classifier:

- Train M different SVM classifiers $h_0(\mathbf{x}), h_1(\mathbf{x}), \ldots, h_{M-1}(\mathbf{x})$.
- Classifier $h_i(\mathbf{x}) = \mathbf{w}_i^T \mathbf{x} - \theta_i$ is trained to classify if $\mathbf{x}$ belongs to i-th class or not.
- For a new test point $\mathbf{z}$, get scores for each classifier, that is, $s_i = h_i(\mathbf{z})$.
- s_i represents the classification margin of the test point from the boundary separating i-th class and the rest of the classes.
- Predict the label as $\hat{y} = \max\limits_{i=0,1,2,\ldots,M-1} s_i$

Problem 5. (**5 pts**) 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.



Solutions:

The support vectors are $\mathbf{x}^b = (4,5)$ and $\mathbf{x}^r = (2,3)$; we have used b and r to denote blue and red support vectors respectivley. Decision boundary must be passing through (3,4) and perpendicular to the line connecting the support vectors to maximize the classification margin. This yields the decision boundary

$$x_1 + x_2 - 7 = 0.$$

If we compare this with the notation we adopt $w_1x_1 + w_2x_2 - \theta = 0$, we obtain $w_1 = w_2$

In SVM formulation, we also require the following equations to hold

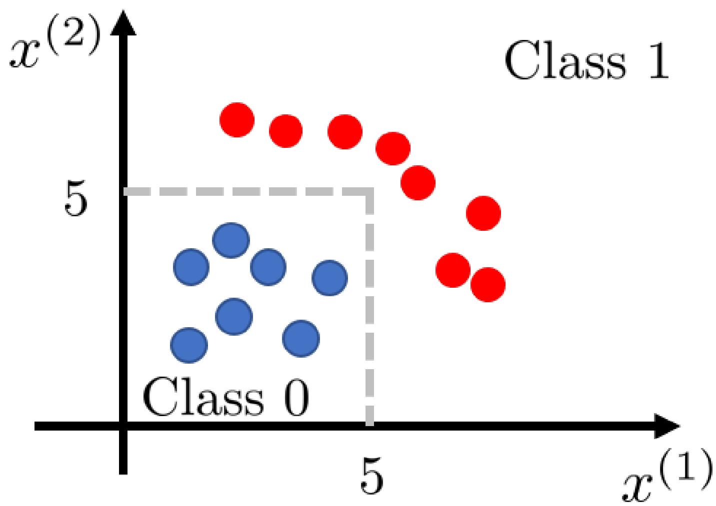
$$\mathbf{w}^T\mathbf{x}^b - \theta = 1, \quad 2w_1 + 3w_2 - \theta = 1,$$

$$\mathbf{w}^T\mathbf{x}^r - \theta = -1, \quad 4w_1 + 5w_2 - \theta = -1,$$

which yields $w_1 = w_2 = \frac{-1}{2}$, and $\theta = -7/2$.

Classification margin is given by $\frac{2}{\|\mathbf{w}\|} = 2\sqrt{2}$.

Problem 6. (**10 pts**) 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) (**3 pts**) 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) (**7 pts**) 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 7. (**10 pts**) 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 2: Data points for agglomerative clustering

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

(b) (**3 pts**) 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) (**2 pts**) 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) (**3 pts**) 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).

Solutions:

(a) We cannot use hard-SVM as the classes are not linearly separable.

(b) We simply need to extend dimension by 1, that is, $x_3 = \Phi(\mathbf{x}) = x_1^2 + x_2^2$.

(c) For class 1, we have $x_1^2 + x_2^2 = 8$ for all points. For class 0, we have $x_1^2 + x_2^2 = 2$ for all points. Maximum margin SVM decision boundary will be $x_1^2 + x_2^2 = 5$, indicated on the plot.

(d) $K(\mathbf{x}, \mathbf{x}') = \Phi^T(\mathbf{x})\Phi(\mathbf{x}) = (x_1^2 + x_2^2)({x'_1}^2 + {x'_2}^2)$