Haolun Cheng
1882563827

(Written answers start at page 4)

EE 559 | **Homework 3 (Week 5)** | Posted: Fri., 2/11/2022
Jenkins | | Due: Fri., 2/18/2022, 11:59 PM PST

1. You are given the following training data in a 1D (1 feature), 2-class problem:

$$S_1: x_1 = 0,\ x_2 = 2,\ x_3 = 3$$

$$S_2: x_4 = -2,\ x_5 = -3,\ x_6 = -4$$

(a) (i) Plot the data in non-augmented feature space. Draw a linear decision boundary H (a point) that correctly classifies them, showing which side is positive.

(ii) Give the weight vector $\underline{w}$ that corresponds to your H. Because the length of $\underline{w}$ is not determined by the given quantities, choose $\underline{w}$ so that $||\underline{w}|| = 1$.

Hint: the boundary is given by $g(x) = 0$, and the decision region for Γ_1 is $g(x) > 0$.

(b) Plot the data points (in augmented form) in augmented feature space. Also show the decision boundary and which side is positive.

(c) Plot the augmented data points as lines $g_{\underline{x}_n}(\underline{w})$ in weight space. Show the positive side of each such line with a small arrow. Show the solution region.

(d) Plot the weight vector $\underline{w}$ of H from part (a) as a point in weight space. Is $\underline{w}$ in the solution region?

For parts (e)-(f) below, you will use the same dataset of points $\underline{x}_n$ given above, except in augmented form as in (b), but you will reflect them and use the reflected data points $z_n \underline{x}_n$ for your plots.

(e) Plot the reflected data points $z_n \underline{x}_n$ in augmented feature space. Draw the linear decision boundary H from your part (b) and its weight vector. Does H correctly classify all the data points?

Hint: write down the general condition for correct classification of reflected data points.

(f) Plot the data points $z_n \underline{x}_n$ as lines in 2D weight space, showing the positive side of each with a small arrow. Show the final solution region.

2. This problem uses the notation we used in Lecture 6, and m is a positive integer. For the following computational complexity:

$$p(m) = 2m + 1000$$

(a) Is $p(m) = O(m)$?

If yes, prove your answer by letting $a = 1$, and solve for what m_0 we have $m \geq p(m)\ \ \forall m \geq m_0$. If you need a larger a, then state what value of a will work. Find the smallest such integer m_0 for the value of a you used.

If no, justify why not.

(b) Is $p(m) = \Omega(m)$?

If yes, prove your answer by letting $b = 1$, and solve for what m_1 we have $m \le p(m) \quad \forall m \ge m_1$. If you need a smaller *b*, then state what value of *b* will work. Find the smallest such integer m_1 for the value of *b* you used.

If no, justify why not.

(c) Is $p(m) = \Theta(m)$?

Justify your answer.

3. This problem also uses the notation of Lecture 6.

(a) Suppose we have a function $p(m)$ that can be expressed as:

$$p(m) = p_1(m) + p_2(m) + p_3(m)$$

and we have:

$$p_k(m) = O(q_k(m)), \quad k = 1,2,3 \tag{i}$$

Prove that:

$$p(m) = O(q_1(m) + q_2(m) + q_3(m)) \tag{ii}$$

Hints: (i) Use the definition of big-O.

(ii) If you find the problem statement unclear or confusing, try looking at the example in the appendix below.

(b) Is a similar statement to (a) true for $\Omega(..)$? (That is, if you replace each $O(..)$ in part (a) with $\Omega(..)$, would the last equation be true?) Justify your answer.

4. Consider the following computational complexity:

$$p(m) = m^2 \log_2 m + \frac{2m^3}{\log_2 m} + (\log_2 m)^3$$

(a) Is $p(m) = O(m^3)$? If yes, justify it by showing it satisfies the definition. If no, justify why not.

Hint: $p(m)$ is the sum of 3 terms. Try setting $a = 1$, and check each term individually to see if it is $O(m^3)$ (if it is, then give the value of each m_0). Then use the result of Problem 3(a).

(b) Is $p(m) = \Omega(m^3)$? Justify your answer.

5. For a nearest-means classifier, with $C = 2$ classes (held constant), $\frac{1}{2}N$ data points in each class (assume N is even), and D features, find the computational time complexity and space complexity, for the operations given in (a) and (b) below.

For all parts of this problem, assume a completely serial computer. Your answer may be in the form of a big-O upper bound; express the big-O bound in simplest form, without making it looser (higher) than necessary. As part of your answer to each part, please:

(i) Give a brief statement of the algorithm you are analyzing

(ii) Show your reasoning in calculating the complexity

(a) Computing each mean vector from the training data (the training phase)

(b) Classifying M data points, given the mean vectors (the classification phase)

(c), (d) Repeat parts (a) and (b), except for a C-class problem, with $\frac{N}{C}$ data points per class, in which C is a variable.

Hint for (d): you may use without proof that finding the minimum of C values that are unsorted can be done in $O(C)$ time and $O(1)$ space.

Appendix - Example (relates to Problem 3)

Suppose we want to find and prove the (tightest) asymptotic upper bound of $p(m)$, with:

$$p(m) = 3m^3 + 100m^2 \log_2 m + 0.1(2^m)$$

Applying the definition directly to $p(m)$ (especially to prove your bound, including finding m_0) might be difficult. Instead, you could use the result of Problem 3a, to apply the big-O bound to each term independently:

$$3m^3 = O(m^3)$$
$$100m^2 \log_2 m = O(m^2 \log m)$$
$$0.1(2^m) = O(2^m)$$

Then using Problem 3a equation (ii), we can conclude:

$$\begin{aligned} 3m^3 + 100m^2 \log_2 m + 0.1(2^m) &= O(m^3 + m^2 \log m + 2^m) \\ &= O(2^m) \end{aligned}$$

1. You are given the following training data in a 1D (1 feature), 2-class problem:

$$S_1: x_1 = 0,\ x_2 = 2,\ x_3 = 3$$

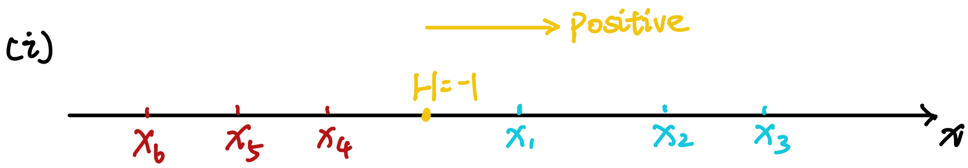
$$S_2: x_4 = -2,\ x_5 = -3,\ x_6 = -4$$

(a) (i) Plot the data in non-augmented feature space. Draw a linear decision boundary H (a point) that correctly classifies them, showing which side is positive.

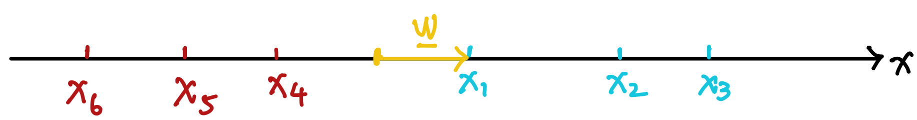
(ii) Give the weight vector $\underline{w}$ that corresponds to your H. Because the length of $\underline{w}$ is not determined by the given quantities, choose $\underline{w}$ so that $||\underline{w}|| = 1$.

Hint: the boundary is given by $g(x) = 0$, and the decision region for Γ_1 is $g(x) > 0$.

(i)



(ii)



$||\underline{w}|| = 1$

$g(x) = w_1 x + w_0 = 0$

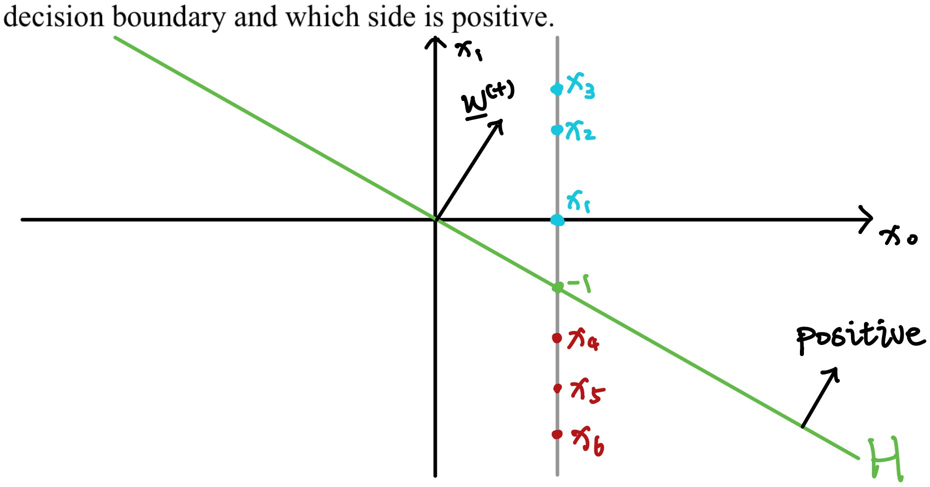
$\underline{w} = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}$

$-\dfrac{w_0}{w_1} = -1$

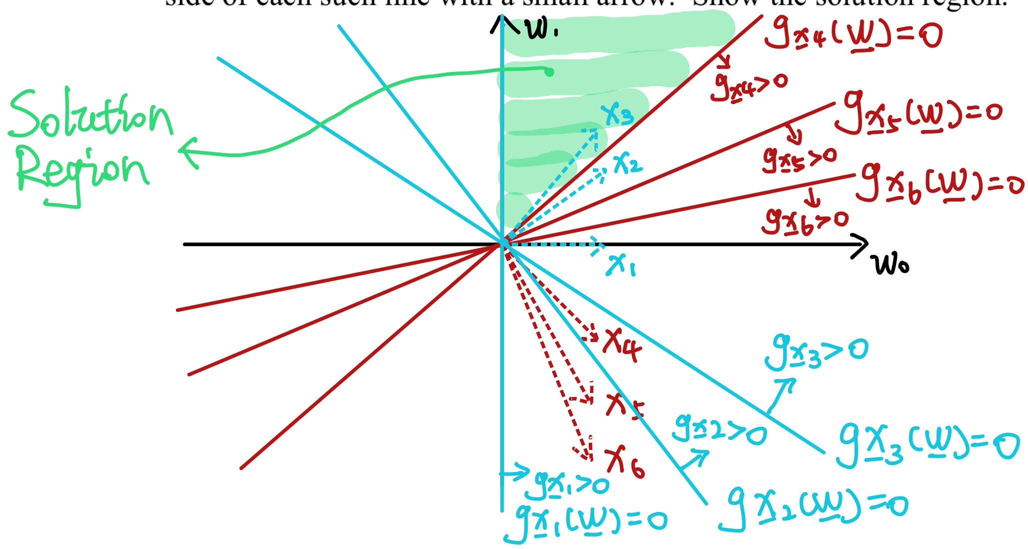
$w_0^2 + w_1^2 = 1$

$w_0 = w_1 = \dfrac{\sqrt{2}}{2} \Rightarrow \boxed{\underline{w} = \begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix}}$

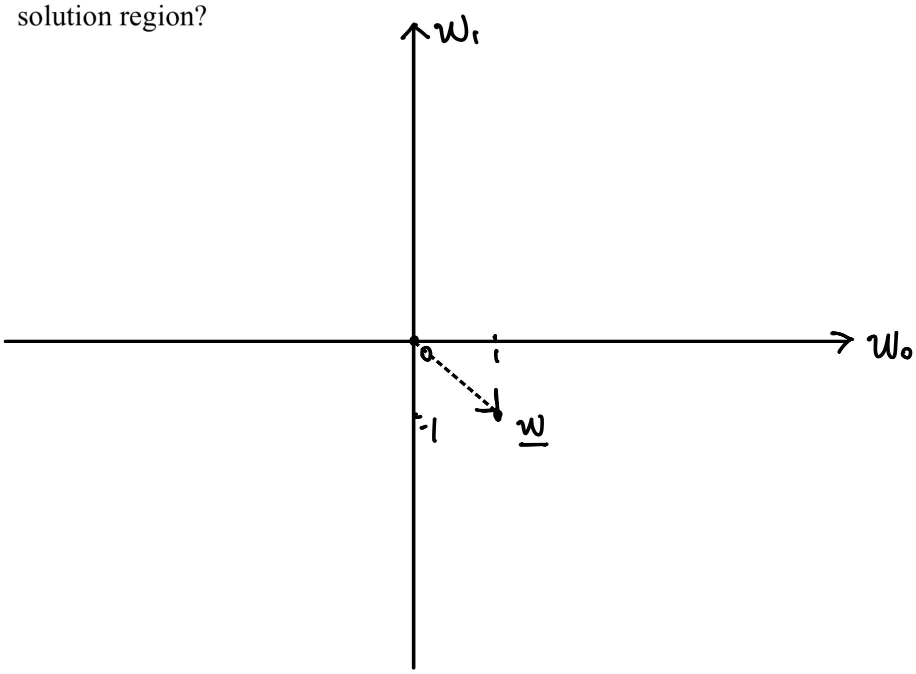
(b) Plot the data points (in augmented form) in augmented feature space. Also show the decision boundary and which side is positive.



(c) Plot the augmented data points as lines $g_{\underline{x}_n}(\underline{w})$ in weight space. Show the positive side of each such line with a small arrow. Show the solution region.



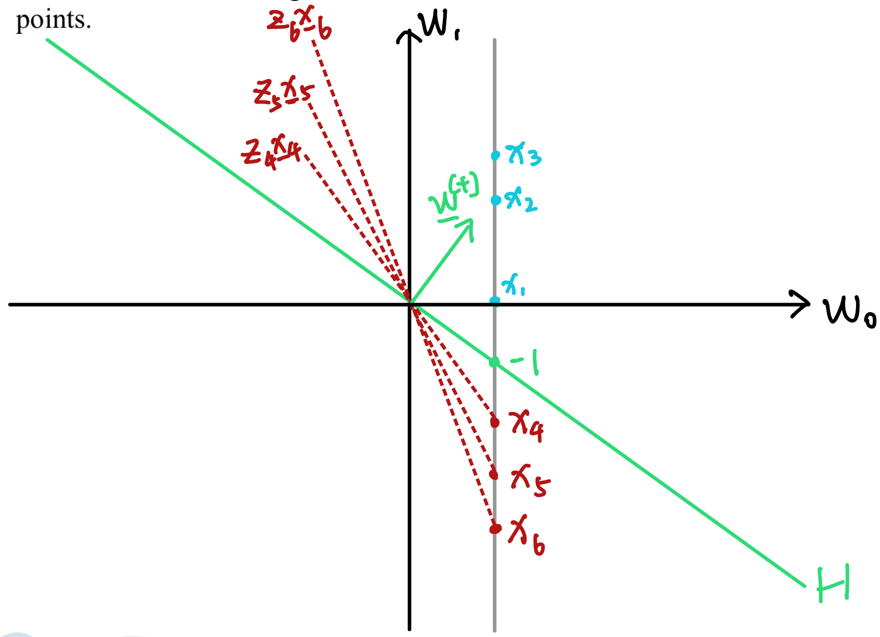
(d) Plot the weight vector $\underline{w}$ of H from part (a) as a point in weight space. Is $\underline{w}$ in the solution region?



According to part (c), $\underline{w}$ is not in the solution region.

(e) Plot the reflected data points $z_n \underline{x}_n$ in augmented feature space. Draw the linear decision boundary H from your part (b) and its weight vector. Does H correctly classify all the data points?

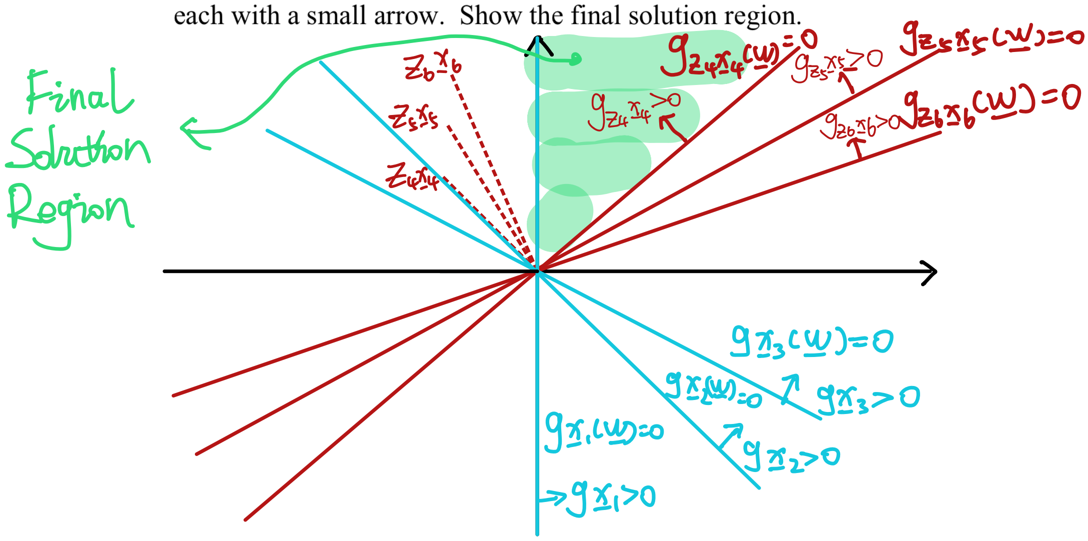
Hint: write down the general condition for correct classification of reflected data points.



$\because\ g(\underline{x}) = \underline{w}^t z_n \underline{x}_n > 0$

$\therefore\ \underline{x}^{(1)}$ correctly classified.

(f) Plot the data points $z_n \underline{x}_n$ as lines in 2D weight space, showing the positive side of each with a small arrow. Show the final solution region.



2. This problem uses the notation we used in Lecture 6, and m is a positive integer. For the following computational complexity:

$$p(m) = 2m + 1000$$

(a) Is $p(m) = O(m)$?

If yes, prove your answer by letting $a = 1$, and solve for what m_0 we have $m \geq p(m) \quad \forall m \geq m_0$. If you need a larger a, then state what value of a will work. Find the smallest such integer m_0 for the value of a you used.

If no, justify why not.

Yes, $p(m) = O(m)$.

Let $a=1$, we have $0 \leq p(m) \leq g(m) \ \forall m \geq m_0$

$\because$ Since $p(m) = 2m + 1000$.

$\therefore 0 \leq 2m + 1000 \leq m$

BUT, $2m+1000 \leq m$ is incorrect.

$\because$ We need an $a > 2$, and we will take $a=3$ for the purpose of finding a tight bond.

$\therefore 2m + 1000 \leq 3m$

$\Longrightarrow m \geq 1000$

Therefore we have $a=3$ and $m_0 = 1000$.

(b) Is $p(m) = \Omega(m)$?

If yes, prove your answer by letting $b = 1$, and solve for what m_1 we have $m \leq p(m) \quad \forall m \geq m_1$. If you need a smaller b, then state what value of b will work. Find the smallest such integer m_1 for the value of b you used.

If no, justify why not.

Yes, $p(m) = \Omega(m)$.

Let $b=1$, we have $0 \leq g(m) \leq p(m) \ \forall m \geq m_1$.

$\because$ Since $p(m) = 2m + 1000$

$\therefore 0 \leq m \leq 2m + 1000$

We have $m \geq -1000$

Since $m > 0$, for any integer $m_1 > 0$ and $0 < b < 2$ will work.

(c) Is $p(m) = \Theta(m)$?

Justify your answer.

Yes, since, from part (a) and (b), $p(m) = O(q(m))$ and $p(m) = \Omega(q(m))$, we have $p(m) = \Theta(q(m))$, which is $p(m) = \Theta(m)$.

3. This problem also uses the notation of Lecture 6.

(a) Suppose we have a function $p(m)$ that can be expressed as:

$$p(m) = p_1(m) + p_2(m) + p_3(m)$$

and we have:

$$p_k(m) = O(q_k(m)), \quad k = 1,2,3 \qquad \text{(i)}$$

Prove that:

$$p(m) = O(q_1(m) + q_2(m) + q_3(m)) \qquad \text{(ii)}$$

Hints: (i) Use the definition of big-O.

(ii) If you find the problem statement unclear or confusing, try looking at the example in the appendix below.

$\because$ Suppose for $p_k(m) = O(q_k(m))$, $k=1,2,3$, we have constants a_k, $k=1,2,3$.

$\therefore p_1(m) + p_2(m) + p_3(m) \le a_1 q_1(m) + a_2 q_2(m) + a_3 q_3(m)$

$\Rightarrow p(m) \le a_1 q_1(m) + a_2 q_2(m) + a_3 q_3(m)$

$\because$ a_1, a_2, a_3 might be different.

$\therefore p(m) \le \max(a_k)(q_1(m) + q_2(m) + q_3(m))$, for $k=1,2,3$

$\therefore p(m) = O(q_1(m) + q_2(m) + q_3(m))$.

(b) Is a similar statement to (a) true for $\Omega(..)$? (That is, if you replace each $O(..)$ in part (a) with $\Omega(..)$, would the last equation be true?) Justify your answer.

$\because$ Suppose for $P_k(m) = \Omega(q_k(m))$, $k=1,2,3$, we have constants b_k, $k=1,2,3$.

$\therefore P_k(m) \geq b_1 q_1(m) + b_2 q_2(m) + b_3 q_3(m)$

$\because b_1, b_2, b_3$ might be different.

$\therefore P(m) \geq \min(b_k)(q_1(m) + q_2(m) + q_3(m))$, for $k=1,2,3$

$\therefore P(m) = \Omega(q_1(m) + q_2(m) + q_3(m))$.

Yes, it is still true for $\Omega(\cdots)$.

4. Consider the following computational complexity:

$$p(m) = m^2 \log_2 m + \frac{2m^3}{\log_2 m} + (\log_2 m)^3$$

(a) Is $p(m) = O(m^3)$? If yes, justify it by showing it satisfies the definition. If no, justify why not.

Hint: $p(m)$ is the sum of 3 terms. Try setting $a = 1$, and check each term individually to see if it is $O(m^3)$ (if it is, then give the value of each m_0). Then use the result of Problem 3(a).

Yes, $p(m) = O(m^3)$.

Suppose $a=1$, we have the following:

① $\left.\begin{array}{l} m^2 \log_2 m \leq m^3 \\ \log_2 m \leq m \\ m \geq 1 \end{array}\right\} \Rightarrow m^2 \log_2 m = O(m^3) \quad \forall m \geq m_0$

② $\left.\begin{array}{l} \frac{2m^3}{\log_2 m} \leq m^3 \\ 2 \leq \log_2 m \\ m \geq 4 \end{array}\right\} \Rightarrow \frac{2m^3}{\log_2 m} = O(m^3) \quad \forall m \geq m_0$

③ $(\log_2 m)^3 \leq m^3$, $m \geq 1$ $\Rightarrow c(\log_2 m)^3 = O(m^3)$ $\forall\ m \geq m_0$

Using the result of problem 3(a), we have:

$$p(m) = m^2 \log_2 m + \frac{2m^3}{\log_2 m} + (\log_2 m)^3 = O(m^3)$$

(b) Is $p(m) = \Omega(m^3)$? Justify your answer.

NO, $p(m) \neq \Omega(m^3)$.

Suppose $b = 1$, we have the following:

① $m^2 \log_2 m \geq m^3$, $\log_2 m \geq m$ $\Rightarrow$ We could not find the range of m, and thus cannot prove its relationship with m^3.

② $\frac{2m^3}{\log_2 m} \geq m^3$, $2 \geq \log_2 m$, $0 < m \leq 4$ $\quad \frac{2m^3}{\log_2 m} = \Omega(m^3)$

③ $(\log_2 m)^3 \geq m^3$, $0 < m \leq 1$ $\quad (\log_2 m)^3 = \Omega(m^3)$

$\therefore$ We can't find m_1, b such that $0 \leq b m^3 \leq P(m)$ $\forall m \geq m_1$

$$P(m) = m^2 \log_2 m + \frac{2m^3}{\log_2 m} + c(\log_2 m)^3 \neq \Omega(m^3)$$

5. For a nearest-means classifier, with $C = 2$ classes (held constant), $\frac{1}{2}N$ data points in each class (assume N is even), and D features, find the computational time complexity and space complexity, for the operations given in (a) and (b) below.

For all parts of this problem, assume a completely serial computer. Your answer may be in the form of a big-O upper bound; express the big-O bound in simplest form, without making it looser (higher) than necessary. As part of your answer to each part, please:

(i) Give a brief statement of the algorithm you are analyzing

(ii) Show your reasoning in calculating the complexity

(a) Computing each mean vector from the training data (the training phase)

(b) Classifying M data points, given the mean vectors (the classification phase)

(c), (d) Repeat parts (a) and (b), except for a C-class problem, with $\frac{N}{C}$ data points per class, in which C is a variable.

Hint for (d): you may use without proof that finding the minimum of C values that are unsorted can be done in $O(C)$ time and $O(1)$ space.

$C=2$ classes ; $\frac{1}{2}N$ data points per class ; D features ;

(a) (i) Algorithm: Load each data point to calculate the sample means for each class.

Let $\frac{N_2}{2}$ = # of data points in each class ($\frac{N_1}{2} = \frac{N_2}{2}$)

D = # of features.

$$\underline{M}_k = \frac{1}{\frac{N_2}{2}} \sum_{i=1}^{\frac{N_2}{2}} \underline{x}_i^{(k)} = F_2 \sum_{i=1}^{\frac{N_2}{2}} \underline{x}_i^{(k)} , \quad F_2 = \frac{2}{N_2} .$$

Initialize $(\underline{x}_{sum}^{(k)})_j = 0 \quad \forall j,k$

For each $k = 1, \dots, C$

For each $j = 1, \dots, D$

For each $i = 1, \dots, N_c$

Load $x_{ij}^{(k)}$

$(\underline{x}_{sum}^{(k)})_j = x_{ij}^{(k)} + (\underline{x}_{sum}^{(k)})_j$

} need memory for 2 scalars $(\underline{x}_{sum}^{(k)})_j$, $x_{ij}^{(k)}$

$\mu_j^{(k)} = F_c \cdot (\underline{x}_{sum}^{(k)})_j$

Output $\mu_j^{(k)}$

(ii) # Adds: $2(\frac{N_2}{2}-1)D$

Mult's: $2D$

Div's: 1

Total time complexity: $(\frac{N_2}{2}-1)\cdot D\cdot 2\cdot T_A + 2\cdot D \cdot T_M + 1\cdot T_D$

$$= (N_2-2)DT_A + 2DT_M + T_D$$

$$= N_2DT_A - 2DT_A + 2DT_M + T_D$$

$$\Rightarrow T = O(ND)$$

Total space complexity: $2+4=6=O(1)$

(b) (i) Algorithm: Given the sample means for each class, we want to classify M data points from the test set. [Sample means for class 1 & 2: mean1, mean2]

```
Initialize classify=[]
For each k=1,...,C.
    For each j=1,...,D
        For each i=1,...,M
            Load x_ij^(k)
            d1 = x_ij^(k) - mean1
            d2 = x_ij^(k) - mean2
            d = d1 - d2
            IF d<0
                classify[i-1]=1
            ELSE If d>0
                classify[i-1]=2
```

13 Output classify

(ii) Total time complexity: We have three for loops, but we only have 2 classes.

$\therefore \boxed{T = O(DM)}$

Total space complexity:

Need space for $\underline{x}_{ij}^{(k)}$, d, d_1, d_2, classify[m],

$\Rightarrow \boxed{O(1)}$ (since m is also constant).

(c) C classes ; $\frac{N}{C}$ data points per class; D features

(i) Algorithm: Load each $\frac{N}{C}$ data points to calculate the sample means for each of C classes.

Initialize $(x_{sum}^{(k)})_j = 0 \quad \forall j, k$

For each $k = 1, \dots, C$

 For each $j = 1, \dots, D$

  For each $i = 1, \dots, \frac{N}{2}$

   Load $x_{ij}^{(k)}$

   $(\underline{x}_{sum}^{(k)})_j = x_{ij}^{(k)} + (\underline{x}_{sum})_j$

 $\mu_j^{(k)} = F_c \cdot (\underline{x}_{sum}^{(k)})_j$

Output $\mu_j^{(k)}$

(ii) Total time complexity:

$$\left(\frac{N}{C}-1\right) D\cdot C\cdot T_A + D\cdot C\cdot T_M + 1\cdot T_0$$

$$= ND\cdot T_A + D\cdot C\cdot T_M$$

$$\therefore\ T = O(ND+DC)$$

Total space complexity: $2+4=6=O(1)$

Two space for $x_{ij}^{(k)}$, $(x_{sum}^{(k)})_j$

Four space for constants.

(d) (i) Algorithm: given sample means for C classes, we want to classify N data points [sample mean: $mean_k$, for $k=1,\ldots,C$]

```
Initialize classify = []
For each i = 1, ..., M
    For each j = 1, ..., D
        distances = []
        For each k = 1, ..., C
            Load x_ij^(k)
            d = ||x_ij^(k) - mean_k||
            distance[k-1] = d.
        classify[i-1] = IndexOf(min[distance])+1.
Output classify.
```

(ii) Total time complexity:

We have three nested for loops with the lengths of M, D, C. $\Rightarrow$ $T = O(MDC)$

Total space complexity:

Need space for classify[], distances[], $x_{ij}^{(k)}$, d,

$\Rightarrow 4 \Rightarrow O(1)$