EE 559 **Homework 3 (Week 5)** Posted: Tuesday, 2/16/2021
Jenkins Due: Tues., 2/23/2021, 11:59 PM PST

1. In Lecture 8 (using a somewhat different approach than in the reading), we derived an expression for the signed distance $r = d(\mathsf{H}, \underline{x})$ between an arbitrary point $\underline{x}$ and a hyperplane H given by $g(\underline{x}) = w_0 + \underline{w}^T \underline{x} = 0$, all in *nonaugmented* feature space. This question explores this topic further.

    (a) Derive, an expression for the signed distance $r = d\left(\mathsf{H}, \underline{x}^{(+)}\right)$ between an arbitrary point $\underline{x}^{(+)}$ and a hyperplane H: $g\left(\underline{x}^{(+)}\right) = \underline{w}^{(+)T} \underline{x}^{(+)} = 0$ in *augmented* feature space. Set up the sign of your distance so that $\underline{w}^{(+)}$ points to the positive-distance side of H.

    (b) In *weight* space, using augmented quantities, derive an expression for the signed distance $r_{\underline{w}}$ between an arbitrary point $\underline{w}^{(+)}$ and a hyperplane $g\left(\underline{w}^{(+)}\right) = \underline{w}^{(+)T} \underline{x}^{(+)} = 0$. Set up the sign of your distance so that $\underline{x}^{(+)}$ points to the positive-distance side of the hyperplane.

2. For a 2-class learning problem with one feature, you are given four training data points (in augmented space):

$$\underline{x}_1^{(1)} = (1, -2);\ \underline{x}_2^{(1)} = (1,1);\ \underline{x}_1^{(2)} = (1,3);\ \underline{x}_2^{(2)} = (1,5)$$

    (a) Plot the data points in 2D feature space. Draw a linear decision boundary H that correctly classifies them, showing which side is positive. Give the weight vector $\underline{w}$ that corresponds to your H.

    (b) Plot the data points $\underline{x}_n$ as lines in 2D weight space, showing the positive side ($g_{\underline{x}_n}(\underline{w}) > 0$) of each with a small arrow. Also show the side of each line that corresponds to correct classification with shading or cross-hatching (similar to plot in Lecture 6 notes, p. 9). Show the final solution region.

    (c) 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 (d)-(f) below, you will use the same dataset of points $\underline{x}_n$ given above, but you will first reflect them and use the reflected data points $z_n \underline{x}_n$ for your plots.

    (d) Plot the reflected data points $z_n \underline{x}_n$ in 2D feature space. Draw the linear decision boundary H from your part (a) 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.

(e) 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. Also show the side of each line that corresponds to correct classification with shading or cross-hatching. (**Same hint** applies as for part (d).) Show the final solution region.

(f) Also, plot the weight vector $\underline{w}$ of H from part (a) as a point in your weight space plot of part (e). Is $\underline{w}$ in the solution region of part (e)?

3. This problem uses the notation we used in Lecture 5, 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.

(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.

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

Justify your answer.

4. This problem also uses the notation of Lecture 5.

(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.

5. 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 4(a).

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

6. 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, 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) is on the next page.

Appendix - Example (relates to Problem 4)

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

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

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 4a, to apply the big-O bound to each term independently:

$$3m^3 = O\left(m^3\right)$$

$$100m^2 \log_2 m = O\left(m^2 \log m\right)$$

$$0.1\left(2^m\right) = O\left(2^m\right)$$

Then using Problem 4a equation (ii), we can conclude:

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