James Rider

Homework 3
Perceptron

CS 6350 - Fall 2022
November 5, 2022

Paper Problems

1.

a.

$||\mathbf{w}||_2 = \sqrt{29}$

Utilizing the formula from class we find that there is a distance from each point to the hyperplane thus a margin does exist and it is the smallest of these which means that the hyperplane has a margin with this dataset of $\frac{1}{\sqrt{29}}$. I was trying to include a table with all of the distances but LaTeX didnt like my table for some reason so I gave up. I think it was me trying to put square roots in it.

b.

This new table only adds one additional data point, [-1, -1]. Evaluating the distance to the hyperplane for this point we find that it has a distance of $\frac{9}{\sqrt{29}}$. Thus it still has the same margin as the previous table even if it is incorrectly labeling that point. Now if we had added the point [6, -3] we would have a distance of 0 and thus not have a margin.

2.

a.

This dataset is linearly separable thus we can find its margin. In this case the hyperplane with the largest margin is going to be $x_1 + x_2 = 0$ thus we will have a margin of $\frac{1}{\sqrt{2}}$.

b.

We cannot calculate the margin of this dataset because the data is not linearly separable.

3.

Extra credit, skipping for now.

4.

Step One: Find R

The vector that give the maximum length is the vector that has 0 up to x_k and 1 from x_k to x_{2k}. Thus will have 2k features that result in 1 and then we augment a constant feature. That means that we find that $R = \sqrt{1+2k}$.

Step Two: Find a separating hyperplane with a nonzero margin.

There exists a hyperplane with a non zero margin as follows:

$$-x_1 + ... + -x_k + x_{k+1} + ...k_{2k} + \left(k - \frac{1}{2}\right) \geq 0$$

The maximum distance of this hyperplane is found using the same vector outlined previously, lets call this $\mathbf{w}$. Recall that is 0s up to x_k then 1s up to x_{2k}. Meaning that the norm of this vector in the augmented space is going to be,

$$\begin{aligned} ||\mathbf{w}|| &= \sqrt{k + \left(k - \frac{1}{2}\right)^2} \\ &= \sqrt{k + \left(k^2 - k + \frac{1}{4}\right)} \\ &= \sqrt{k^2 + \frac{1}{4}} \end{aligned}$$

Meaning that the normalized margin is going to be,

$$\gamma = \frac{\left(k - \frac{1}{2}\right)}{\sqrt{k^2 + \frac{1}{4}}}$$

Step Three: Number of mistakes

Thus the number of mistakes has an upper bound of

$$\begin{aligned} \frac{R^2}{\gamma^2} &= (1+2k)^2 / \left(\frac{\left(k - \frac{1}{2}\right)}{\sqrt{k^2 + \frac{1}{4}}}\right)^2 \\ &= \left(k^2 + \frac{1}{4}\right)(1+2k)^2 / \left(k - \frac{1}{2}\right)^2 \\ &= \frac{\left(4k^2+1\right)(2k+1)^2}{(2k-1)^2} \end{aligned}$$

Alright I tried simplifying that into something nice and it ain't happening. At least I got rid of the fractions, right? I'm just hoping its right.

5.

I hoping an examples of a labelings that cannot be shattered for any possible configuration of points is sufficient to prove this.

Consider 4 distinct points x1, x2, x3, x4. Also, assume our labels are x and o. There are four possible ways that these points can be laid out.

1. **All four points aligned.**
   In this case it should be pretty obvious that any labeling in which the labels are split cannot be classified by a line. Consider the ordering of labels along this line of x, o, o, x. It should be apparent that there are no ways to separate these with a linear classifier. Any line that isolates x1 would group x2, x3, and x4 and thus misclassify x4.
2. **Three points aligned**
   The next case to consider is that if x1, x2, and x3 are aligned with x4 off on its own. Assume that the labels for the x1, x2, and x3 alternate. In this instance it does not matter what the label of x4 is because there is no way to label x1 and x3 with the same label without giving x2 that label as well.
3. **Triangle with 4th point inside**
   This example is one where 3 of the points, lets chose x1, x2, x3 make up the corners of a triangle with x4 inside it. It should be pretty obvious that if the label of x4 is different from the labels of x1, x2, and x3 that there is no way to linearly separate it without catching one of the other points along with it.
4. **Four vertex shape**
   In this final case consider some shape where each of the points is used as corners of the shape. A labelling where each point has a different label than its neighbors would be impossible to classify. This labelling would result in opposite corners sharing a label. No matter how you draw a line through this shape it would not be possible to assign distinct label to one corner without misclassifying the opposite corner. It should be noted that there is obviously a case in which one of the corners is inside the other three points and then you could classify the opposing corners but this instance is covered by my case three outlined above.

Honestly, I should probably have included pictures but I really didn't feel like and I feel like my descriptions of each case should be sufficient.

6.

VC($\mathcal{H}$) = 4.

Given a set of five points in the plane, one of the points must lie inside a rectangle bounded by the other four. In this case if we label this interior point negative and the external points positive then there is no rectangle that can correctly label the points. I feel like this was too easy and am probably missing something.

Practice Problem 2

(c)

Not entirely sure what I'm supposed to be observing here. It would appear that the learned weight vector of the average Perceptron roughly lines up with what the individual weight vectors from voted Perceptron gives. In that I mean that the values relative to each other are similar as are their signs.

(d)

I ran them all several times to observe their averages. The standard Perceptron algorithm had the most variance going as high as 10% in one of the runs but most of the time performing the same as the other two and occasionally better. Whereas the other two were very consistent and usually matching each other at 1.4%. There was slight variation in them but not nearly as much as the standard algorithm. Therefore we can conclude that these other methods reduce the variance of the output, which makes sense since they're similar to ensemble methods.