CSE 151A Winter 2021 - Take Home Exam 4

1 hour **Name:** **PID:**

---

(1) This is an open book, open notes exam. You are free to consult any text book or notes. **You are not allowed to consult with any other person.**

(2) If you need any clarification, please post a private message to the instructors on Piazza.

(3) Remember that your work is graded on the *clarity* of your writing and explanation as well as the validity of what you write.

(4) This is a one-hour exam.

(1) We are given a training set $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ where $y_i \in \{-1, 1\}$ and each x_i is a $d \times 1$ vector. Suppose we know that the feature vectors $x_1, \ldots, x_n$ lie on a k-dimensional subspace T of $\mathbb{R}^d$. State whether the following statements are true or false. Justify your answer.

(a) (5 points) Suppose we run Perceptron for a single pass on S starting with an initial point $w_0 = 0$ (the all zeros vector). Does the output w_P lie in T? Justify your answer.

Solution

Yes, the output weights w_P will lie in T. This can be explained by looking at the perceptron algorithm. In perceptron algorithm, we initialize our weights with some initial value w_0 which is the zero vector in our case. Then, for every data-point x_t, we update the weights as per the equation $w_{t+1} = w_t + y_t x_t$ if the point is wrongly classified (i.e. $sign(< w_t.x_t >) \neq y_t$) and the weights remain the same if the point is correctly classified. Thus, w_P can be written as

$$w_P = \sum_{i=1}^{n} \delta_i x_i \text{ where } \delta_i \in \{-1, 0, 1\}$$

Here, $\delta_i = y_i$ for wrongly classified data-points and $\delta_i = 0$ for points which get correctly classified. Thus, w_P is a linear combination of the feature vectors x_is all lying in the subspace T which implies that w_P will also lie in T.

(b) (5 points) Now suppose we run gradient descent for logistic regression on S for a 100 iterations starting with an initial point $w_0 = 0$. Does the output w_L lie in T? Justify your answer.

Solution

Yes, the output weights w_L lie in T. This can be explained in a similar way as that of perceptron. The update rule for Logistic Regression (From Class Notes) is -

$$w_{t+1} = w_t + \eta_t \sum_{i=1}^{n} \frac{y_i x_i}{1 + e^{y_i w^T x_i}}$$

$$w_{t+1} = w_t + \eta_t \sum_{i=1}^{n} \delta_i x_i \text{ where } \delta_i = \frac{y_i}{1 + e^{y_i w^T x_i}}$$

Here, δ_i is a scalar. Thus, In an iteration, the update to the weights is a linear combination of the feature vectors x_is all lying in the subspace T. This implies that the resulting weights w_L will also be a linear combination of the feature vectors and will lie in T.

(2) (5 points) Write down an example of a dataset that is (a) linearly separable, but (b) where running a single pass of Perceptron does not lead to a classifier with zero training error.

Solution

A 2-D Example with two data points. Let $S = \{(1,2),(1,3)\}$ where (1,2) has a label -1 and (1,3) has a label 1. This is clearly a linearly separable data in 2D space. Let the initial weight vector $w_0 = (0,0)$.

Step: 1 Input x is (1,2). $w^T x = 0$ since $w = 0$. As per the perceptron rule, we'll update the weights. $w_1 = (0,0) + (-1)(1,2) = (-1,-2)$

Step: 2 Input x is (1,3). $w^T x = -7$. As per the perceptron rule, we'll update the weights. $w_1 = (-1,-2) + (1)(1,3) = (0,1)$

This completes a single pass on the dataset. With these weights, the data point (1,2) is still mis-classified as $w^T x = 2 > 0$ and the label for this point is -1.

(3) (5 points) Suppose Alice and Bob are given the same training dataset S. Alice finds a classifier w_A that exactly minimizes the logistic regression loss function. Bob finds a classifier w_B that minimizes the loss function:

$$w_B = \mathbf{argmin}_w \exp(\sum_{i=1}^{n} \log(1 + e^{-y_i w^\top x_i}))$$

Is $w_A = w_B$ for all training sets S? Justify your answer.

Solution

Yes, $w_A = w_B$ for all training sets S. This is because, in both cases, the loss function is equivalent. From lecture notes, we have the loss function for w_A is

$$w_A = \mathbf{argmin}_w \sum_{i=1}^{n} \log(1 + e^{-y_i w^\top x_i})$$

Just like log, exp is also a monotonic function and thus minimizing/maximizing x and $exp(x)$ are equivalent. Thus loss function for w_A can also be written as

$$w_A = \mathbf{argmin}_w \exp\left[\sum_{i=1}^{n} \log(1 + e^{-y_i w^\top x_i})\right]$$

which is same as the loss function for w_B. Hence, these classifiers are equivalent.