Machine Learning Algorithms

Rishabh Pomaje

July 23, 2023

1 Perceptron Learning Algorithm

Consider we have the data points,

$$(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_N, y_N)$$

Let us assume that the given data is linearly separable.We find the hyperplane of separation as follows :

$$h(\mathbf{x}) = sgn(\sum_{i=1}^{d} w_i x_i - threshold)$$

Here the input($\mathbf{x}$) and weight vector($\mathbf{w}$) is has dimension 'd' and thus,

$$\mathbf{x} = (x_1, x_2, \cdots, x_d)$$

$$\mathbf{w} = (w_1, w_2, ..., w_d)$$

To simplify the expression for the hypothesis,let $-threshold = w_0$.We get,

$$h(\mathbf{x}) = sgn(\sum_{i=1}^{d} w_i x_i + w_0(1))$$

Thus, we incorporate this w_0 and 1 in weight vector and input vector respectively.

$$h(\mathbf{x}) = sgn(\sum_{i=0}^{d} w_i x_i)$$

$$\therefore h(\mathbf{x}) = sgn(\mathbf{w}^T\mathbf{x})$$

1.1 Selection and updating the weights

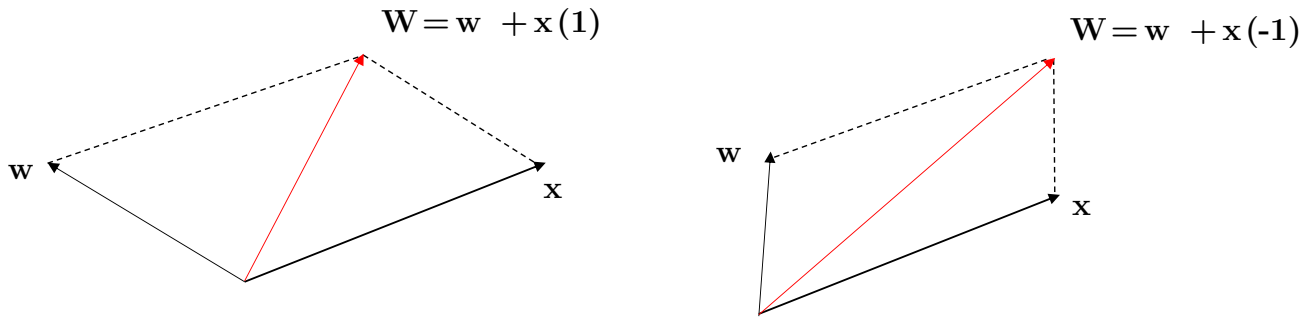
In this algorithm, we start with random weights. Say the weight vector in the i^{th} iteration is $\mathbf{w}_i$.

Definition : A point $\mathbf{x}_k$ is called misclassified if $h(\mathbf{x}_k) \neq y_k$.

For the $(i+1)^{th}$ iteration, we choose a misclassified point and do the following changes :

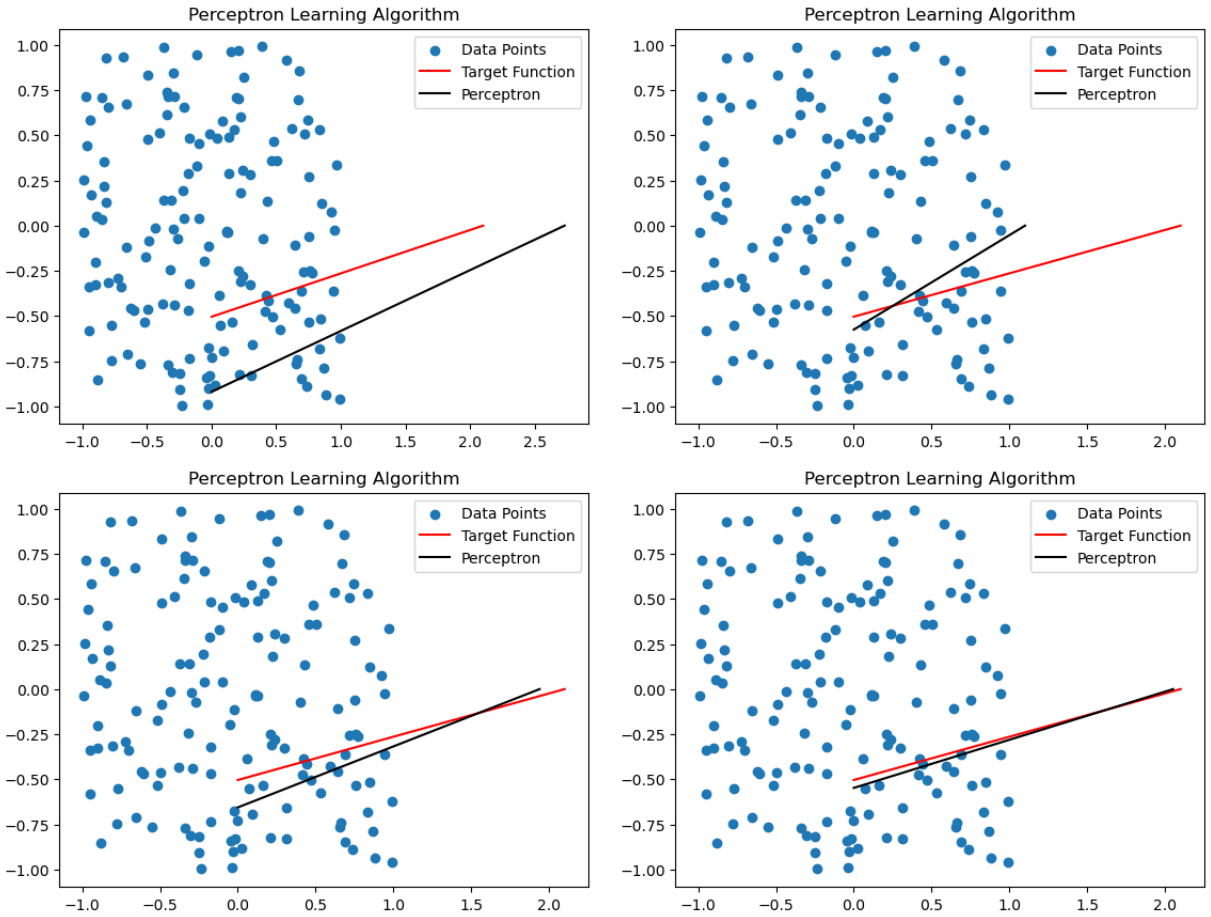
$$\mathbf{w}_{i+1} = \mathbf{w}_i + y_k\mathbf{x}_k$$

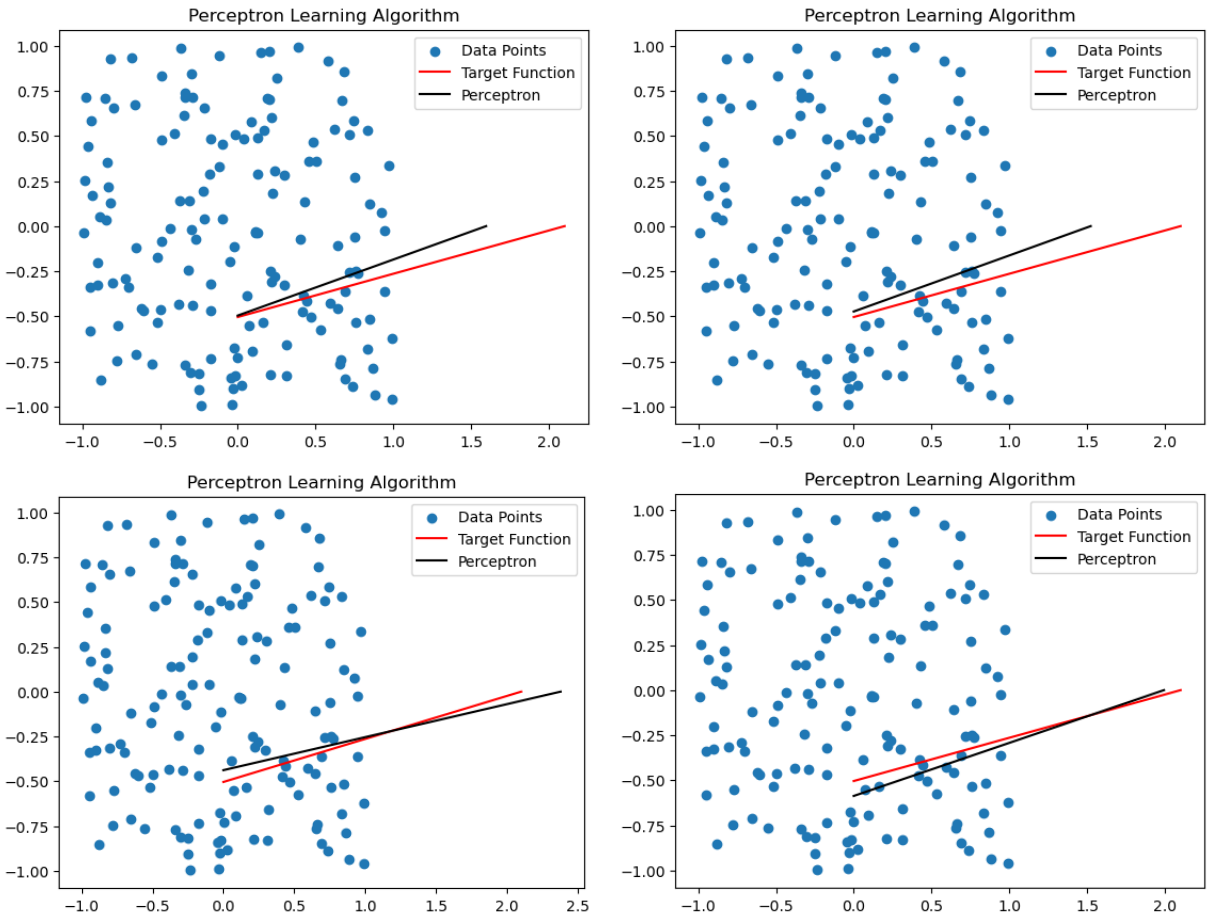
Q.Why does this work ?
Ans.



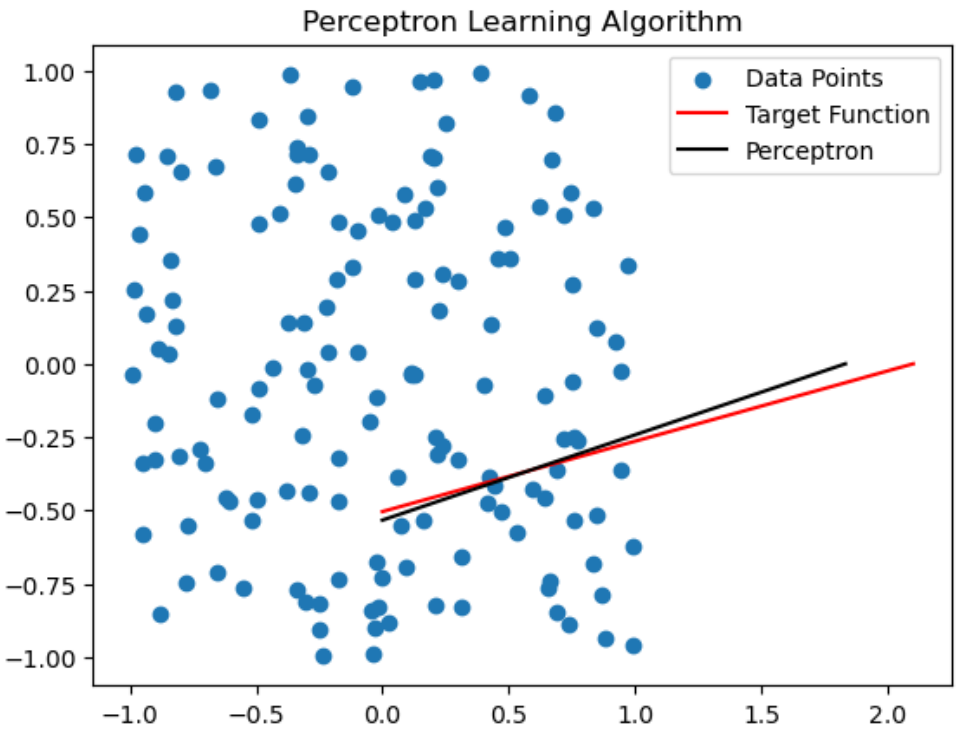
Updating the weight vector-Perceptron Learning Algorithm

1.2 Perceptron in 2-D : Visualized





Iterations in a sample PLA implementation



Final result after 9 iterations