ML Assignment 1: Perceptron

Anirudh Agrawal	Aviral Agarwal	Vikramjeet Das
2018A7PS0099H	2018A7PS0192H	2018A7PS0280H

1 The Perceptron

The perceptron is used for supervised learning in a binary classification task. It models our data as a linear combination of the inputs, with a hard threshold activation applied on it. Given a set of data

$$X = \begin{bmatrix} | & | & & | \\ X^{(0)} & X^{(1)} & ... & X^{(n)} \\ | & | & & | \end{bmatrix}$$

where each $X^{(i)}$ is defined by

$$X^{(i)} = \begin{bmatrix} x^{(1)} \\ x^{(2)} \\ ... \\ x^{(m)} \end{bmatrix}$$

we want to find

$$W = \begin{bmatrix} w_1 \\ w_2 \\ ... \\ w_m \end{bmatrix}$$

such that $y = W^T X$ correctly classifies as many points as possible.

2 Implementation

The algorithm can be roughly outlined by the following steps:

1. Initialize W
2. For all epochs (e)
3. For all training points ((x, y))
4. Predict $y' = W^T x$
5. If $y' \neq y$ and y is positive
6. $W = W + x$
7. If $y' \neq y$ and y is negative
8. $W = W - x$
9. If accuracy did not change much from previous epoch
10. break;

3 Results

Accuracy on dataset 1 hovers around 0.985.
Accuracy on dataset 2 is over 0.99, and mostly a perfect classification is made with accuracy of 1. **Dataset 2** is more linearly separable. The perceptron consistently performs better on Dataset 2 than Dataset 1.

4 Limitations

1. The hard-threshold activation function in a perceptron forces a 0-1 output. It is only suitable for binary classification, and cannot be easily extended to multi-class tasks.

2. Perceptron only works when the data is linearly separable. For non-linearly separable data, perceptron will never converge to a stable well-performing state.