

# Programming Project 1

## A. Implement Perceptron Learning Algorithm (PLA)

1. Implement the full PLA algorithm discussed in the class. Do not use the "perceptron" classifier already implemented in some programming languages.
2. It is recommended to use MATLAB or Python programming languages, where working with datasets as matrices is easier.
3. Use **Bankruptcy data** provided in this document to report the requested results.
4. Use the following two different initial weights separately to learn two perceptrons on the same data.
  - (a)  $\mathbf{w} = \langle w_0, w_1, w_2 \rangle = \langle 0.2, 5.1, 2.3 \rangle$
  - (b)  $\mathbf{w} = \langle w_0, w_1, w_2 \rangle = \langle 1.0, 1.0, 1.0 \rangle$
5. Do not forget to add the bias feature ( $x_0$ ) to the data.
6. Consider learning rate  $\eta = 1$ .
7. For each initial weight vector report two following results:
  - (a) the final updated weight vector that can classify the whole data set without error,
  - (b) the number of iterations through which the algorithm converged to the final weights.

## B. Implement Pocket Algorithm

1. Add two data points  $\langle 5, 1, -1 \rangle$  and  $\langle 3, 1.5, -1 \rangle$  to make data not linearly separable.
2. Modify your PLA implementation to work for non-separable data as well.
3. Set the number of iterations to 2000 and report:
  - (a) the best weight vector
  - (b) the accuracy of the best perceptron you could learn

$$\text{Accuracy} = \frac{\text{\# of correctly classified samples}}{\text{total \# of samples}}$$

x1	x2	Class/Label
3	0.2	-1
1	0.3	-1
4	0.5	-1
2	0.7	-1
0	1	-1
1	1.2	-1
1	1.7	-1
6	0.2	1
7	0.3	1
6	0.7	1
3	1.1	1
2	1.5	1
4	1.7	1
2	1.9	1