"Machine Learning and Computational Statistics"

9th Homework (Part B)

Exercise 4:

Consider the lines (ε1) x_2=0, (ε2) x_1=0 and (ε3) x_1+x_2=2 in the two-dimensional space that all leave the point (4,4) on their positive side. Consider a two-class classification problem where class 1 contains all the points that lie on the positive side of all lines, as well as all the points that lie on the negative side of all lines. Class 0 contains all points of the remaining (polygonal) regions

(i) Design the regions on the plane that correspond to each class.

(ii) Design a multilayer perceptron that solves the above classification problem, where each node is modeled by the relation $y = f(\boldsymbol{w}^T\boldsymbol{x} + w_0)$, where $f(z) = 1$, for $z > 0$ and $f(z) = 0$, otherwise. Give the full architecture along with the weights and thresholds of each node (describe in some detail the steps you followed for designing the network).

Hint: (i) Use the point (4,4) to identify the positive and the negative sides of each line

(ii) Use the theory given in the lecture.

(iii) The equation of a plane that passes through the points (x_{11}, x_{12},x_{13}), (x_{21}, x_{22},x_{23}), (x_{31}, x_{32},x_{33}) is $\begin{vmatrix} x_1 & x_2 & x_3 & 1 \\ x_{11} & x_{12} & x_{13} & 1 \\ x_{21} & x_{22} & x_{23} & 1 \\ x_{31} & x_{32} & x_{33} & 1 \end{vmatrix} = 0$

Exercise 5 (Python code + text):

Consider a two-class, two-dimensional classification problem for which you can find attached two sets: one for training and one for testing (file *HW9a.mat*). Each of these sets consists of pairs of the form (y_i,$\boldsymbol{x}_i$), where y_i is the class label for vector $\boldsymbol{x}_i$. Let N_{train} and N_{test} denote the number of training and test sets, respectively. The data are given via the following arrays/matrices:

- ***train_x*** (a N_{train} x2 **matrix** that contains in its rows the training vectors $\boldsymbol{x}_i$)
- ***train_y*** (a N_{train}–dim. column **vector** containing the **class labels** (0 or 1) of the corresponding training vectors $\boldsymbol{x}_i$ included in ***train_x***).
- ***test_x*** (a N_{test}x2 **matrix** that contains in its rows the test vectors $\boldsymbol{x}_i$)

- ***test_y*** (a N_{test}–dim. column **vector** containing the **class labels** (0 or 1) of the corresponding test vectors $\boldsymbol{x}_i$ included in ***test_x***).

Train the SVM classifier using the training set given above and **measure** its performance using the test set, **using**: (a) the linear kernel, (b) the polynomial kernel and (c) rbf kernel. Perform **several runs** using the attached code, for several choices of the **parameters** included in each kernel and for various values of C.

Exercise 6 (Python code + text):

Consider a two-class, two-dimensional classification problem for which you can find attached two sets: one for training and one for testing (file *HW9b.mat*). Each of these sets consists of pairs of the form $(y_i, \boldsymbol{x}_i)$, where y_i is the class label for vector $\boldsymbol{x}_i$. Let N_{train} and N_{test} denote the number of training and test sets, respectively. The data are given via the following arrays/matrices:

- ***train_x*** (a N_{train} x2 **matrix** that contains in its rows the training vectors $\boldsymbol{x}_i$)
- ***train_y*** (a N_{train}–dim. column **vector** containing the **class labels** (0 or 1) of the corresponding training vectors $\boldsymbol{x}_i$ included in ***train_x***).
- ***test_x*** (a N_{test}x2 **matrix** that contains in its rows the test vectors $\boldsymbol{x}_i$)
- ***test_y*** (a N_{test}–dim. column **vector** containing the **class labels** (0 or 1) of the corresponding test vectors $\boldsymbol{x}_i$ included in ***test_x***).

Train a neural network classifier with a single hidden layer where the nodes have the hyperbolic tangent output function, for (a) 3 nodes, (b) 4 nodes, (c) 10 nodes, (d) 50 nodes (use the MLPClassifier Python function inserting properly the required parameters, see also the attached code), using the training set given above and **measure** the performance using the test set. Comment on the results.