Exercise 4 (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 5 (Python code + text)[1]:

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.