1. Programming Exercise: Kernalized SVM.

Write a piece of PYTHON code to learn an SVM classifier from a data set. Your program should take as input the training set (X, y), parameter C, kernel type (linear/polynomial/rbf) and kernel parameter (which you can take here to be a single real number 'r'; this is used as the degree parameter d in the polynomial kernel, and as the lengthscale parameter l (or σ) in the RBF kernel.

Code : You have been given a PYTHON code SVMcode.py to learn an SVM classifier. You need to edit in the PYTHON code "SVMcode.py" as directed.

Dataset : You are provided a 2-dimensional synthetic data set for this problem. There are two files train.txt and test.txt containing training instances and test instances respectively. The CrossValidation folder contains the train and test data for each fold of the 5-fold cross-validation procedure on the training set train.txt.

Instructions:
1. Your program should read the training data, test data, parameter C, kernel type, r and print the percentage error of your trained SVM model.

2. Run your implementation of SVM on the training set provided using a linear, degree-2 polynomial, degree-3 polynomial and RBF kernel, selecting C from the range $\{1, 10, 10^2, 10^3, 10^4\}$ via 5-fold cross-validation on the training set. Report the average cross-validation error (over 5 folds) for each value of C in this range. [Note: Use files from CrossValidation folder here.]

3. In the case of the RBF kernel, in addition to C, you will also have to choose the lengthscale parameter l from the range $\{1/32, 1/4, 1, 4, 32\}$ using cross-validation. Report the average cross-validation error for each value of C (and l) and the training and test errors achieved by the best parameter value(s). [Note: Use train.txt and test.txt files here.]