COGS 118A Natural Computation I - Assignment 5

Instructor: Prof. Zhuowen Tu
TA: Chen-Yu Lee
Due: 3/4/2014 11:59 PM

Note: Please send a **PDF** version to tbrsun@gmail.com with the subject [COGS 118A Assignment 5] by due date. Do not send a link to Google Doc or other formats. No hard copy. You are encouraged to start the assignment early as it might be time-consuming for people who are not familar with MATLAB.

In this assignment you will play with support vector machines (SVMs) using different kernels (linear and RBF) on different datasets. You can implement your own SVMs solver or you can use the off-the-shelf libraries available online such as LIBSVM (`http://www.csie.ntu.edu.tw/~cjlin/libsvm/`). In this assignment we only consider binary classification problem (which means you might need to convert a dataset with multiple class labels into a binary case).

1. Load the data "ionosphere.mat" from the course website. This Ionosphere dataset is from the UCI machine learning repository. X is a 351x34 real-valued matrix of predictors. Y is a categorical response: "b" for bad radar returns and "g" for good radar returns. This is a binary classification problem as we have two types of labels {b,g} with total 351 feature vectors in 34 dimensions.

  a. First randomly select 80% of the dataset as your training set and the rest 20% as your testing set. You need to convert the labels {b,g} to {1,-1} or {1,0} in order to mathematically train a classifier.

  b. Train a SVMs classifier using a linear kernel. You would need to use cross validation to select the parameter C for the cost of outliers. Usually 5-fold or 10-fold cross validation is sufficient for choosing the parameter C.

  c. Train a SVMs classifier using the radial basis function kernel (RBF) kernel. Again you need to use cross validation to select the parameters C and `gamma` for the RBF kernel.

  d. Now you can play with the different sizes of training and testing data and see how they effect the classification results. Please try 60% for training and 40% for testing and 40% for training and 60% for testing.

  e. In your report we would like to see your implementation (code) and a table listing all experimental results such as:

[Traing,Test]	[80,20]	[60,40]	[40,60]
Linear SVMs	-	-	-
RBF SVMs	-	-	-

2. Load "fisheriris.mat" dataset that contains 150 feature vectors in 4 dimensions. However the label contains three different values {setosa,versicolor,virginica}. In this assignment we treat setosa as one class and the versicolor and virginica as another class so that we can see the problem as a binary classification task. Now repeat the same procedure as in Q1 and list your final accuracy table and code in your report.

3. Load "arrhythmia.mat" dataset that contains 452 feature vectors in 279 dimensions. Again the label contains 13 different classes. In this assignment we treat label 1 as one class and the rest 12 labels as another class, so the task becomes a binary classification. Repeat the same procedure as in Q1 and list your final accuracy table and code in your report.

4. **(Bonus)** Go to UCI machine learning repository (`https://archive.ics.uci.edu/ml/datasets.html`) and choose **two** datasets that have multi-class labels (more than two classes for labels). Design a multi-class classification experiment using one-vs-all scheme and repeat the same procedure as in Q1 (but now it's a multi-class task so you don't need to force the labels to binary case). List your final accuracy table and code in your report.