Assignment-2 (CS-419)

Due Date: 26-Mar-2014 (Wednesday) 23:55hrs

Note: Please do not copy answers from your friends. This assignment carries NO explicit marks. However it is mandatory to answer ALL questions. You are free to use any programming language or software or scripting language. Every question has a clearly mentioned deliverable(s). You MUST create a report consisting of ONLY these deliverables. Do not write anything other than the deliverables in the report. Upload the report in moodle. Each student is given a separate dataset with which he/she has to answer the questions. The student-dataset map is here: `https://docs.google.com/spreadsheet/ccc?key=0An5k7xgBZpnSdHU2enJ1SHgxT3BHMFJZaVlzaTZhaFE&usp=sharing` and the datasets are available at: `http://archive.ics.uci.edu/ml/datasets.html?format=mat&task=cla&att=num&area=&numAtt=&numIns=&type=mvar&sort=typeUp&view=list`.

1. The goal is to compare the following classifiers on your dataset:
   (a) k-nearest neighbors classifier[1]. Employ Euclidean metric in the given input feature space for measuring distances. You should code this up by yourself[2]. Here k is the hyperparameter. Try different values of $k = 1, 5, 10, 50, 100$.
   (b) Logistic Regression. The code is here: `http://www.csie.ntu.edu.tw/~cjlin/liblinear/`. Please go through the README file to know how to use the code. You need to only tweak the following options for this code in the command line and leave others to default:
      i. -s should be set to 0 or 7 (i.e., regularized logistic regression[3]).
      ii. you should try different values[4] of C i.e., the -c should to set to 0.001, 0.01, 0.1, 1, 10, 100, 1000. Hence you need to run the code 7 times with the different values of C.

[1] The label for x is predicted as the majority label among the k nearest neighbors of x in the training set

[2] You may use any data structure for storing the datapoints in the training set

[3] You may try both; but they should ideally give you the same result.

[4] This code implements the case where Gaussian prior over w with zero mean is used. C is the hyperparameter controlling the weightage of the regularizer and the logistic regression log-likelihood. In the lectures C was an expression involving the variance of the prior.

iii. Set -B as 1. This will allow for affine functions rather than linear functions as the classifiers.

(c) k-Gaussian Mixture Bayes classifier, where each class conditional is a Gaussian mixture model with k components. Only if the number of attributes/features in your dataset is more than 100 then use the Naive assumption of each Gaussian's covariance matrix being a diagonal one (else estimate full matrices). Implement the EM algorithm by yourself for estimating the GMM's parameters. Here k is the hyperparameter and try different values of it: $k = 1, 2, 3, 4, 5$.

Split your dataset into "equal" parts such that both the parts contain nearly the same number of examples belonging to a class. For e.g., if the total number of examples belonging to class a,b,c are 50,100,150 repectively in your dataset , then each "equal" part should contain 25,50,75 examples of class a,b,c repectively. Use one of them as the training set. Call the other the Validation set. The deliverables are as follows:

- Provide a plot of log-likelihood vs. iteration number (an iteration ends after an E and an M step) with your EM algorithm when run on any of the class-conditional data. Is the behaviour of the plot intuitive?
- for each of the three algorithms above provide two plots: i) training set accuracy vs. the hyperparameter value. ii) validation set accuracy vs. the hyperparameter value. Note that in both the plots the parameter tuning for each model is performed using the training set alone. Accuracy is simply the average over classes of the fraction of correct predictions in each class.
- Over all the hyper-parameter and the three model/classifier combinations which one achieved the highest training set accuracy and which one obtained the highest validation set accuracy? Is your finding intuitive?