Project One (COSC6328 W08)

Due in the class on March 6th, 2008

You have to work individually. Submit before the deadline. No late submission will be accepted. No Handwriting. Direct your queries to Hui Jiang (hj@cse.yorku.ca).

Build A 2-Class Classifier With Gaussian Models

You are asked to solve a 2-class (class A & B) classification problem based on multivariate Gaussian models. Assume two classes have equal prior probabilities. Each observation feature is a 3-dimension vector. Assume you can collect a set of training data for each class. Based on the training data (provided in the course Web), you consider the following different ways to build such a classifier. Then the estimated models are used to classify some new test data (also provided in the course Web).

1. First of all, let's consider to build a very simple classifier based on single multivariate Gaussian model. Each class is modeled by a single 3-D multivariate Gaussian distribution. For simplicity, we assume each multivariate Gaussian has a **diagonal** covariance matrix. Show how to estimate Gaussian mean vector and covariance matrix for each class based on the Maximum likelihood (ML) estimation. Report the classification accuracy of the ML-trained models as measured in the test data set.

2. Consider to improve the above simple Gaussian classifier by using a more complicated models, namely Gaussian mixture models (GMM), to model each class. Again, you can assume each Gaussian has a **diagonal** covariance matrix. Use the k-means method to initialize the GMM's. Then improve the GMM models iteratively based on the EM algorithm. Investigate and report results for GMM's which have 2, 4, 8 mixture components respectively.

3. Consider to improve the ML-trained single Gaussian classifier in step 1 by using some more advanced training techniques. In this part, you are asked to improve the single Gaussian classifier you got in the first step based on one of the discriminative training methods, namely minimum classification error (MCE) estimation with Generalized probabilistic descent (GPD) algorithm. Study how the following quantities change as you proceed with more and more iterations in GPD training: 1) the actual error rate in training data; 2) the objective function in MCE; 3) the actual error rate in test set. Please note that in GPD training, the key issue is how to choose a proper step size experimentally.

For all above methods, you can only use training data to learn the models. The estimated models are evaluated in the test set. Compare all of these different methods and discuss your results. Report the best classification accuracy you can achieve in the test set.

The Data File Format: all train data in the file *Train.dat*, all test data in the file *Test.dat*; Each line represents one observation feature vector in the format as:

$$Z: \quad (x1 \;\; x2 \;\; x3)$$

where Z is class label, and $(x1 \;\; x2 \;\; x3)$ is a feature vector.

You are required to use a language, such as C, C++, Java, Perl, etc, to implement single Gaussian estimation, EM and MCE/GDP. Email me your programs (three programs for three questions, i.e., Q1.X, Q2.X and Q3.X) before the deadline. Finally, write a report (maximum 6 pages) to summarize what you have done and submit a hardcopy of your report.