Assignment 3

Statistical Machine Learning

Theory Questions

1. **(10 points)** Question 39 from chapter 3
2. **(10 points)** Question 49 from chapter 3
3. **(10 points)** Question 40 from chapter 3 (**only for CSE 542**)

Practical Questions

4. **(20 points)** Download the databases from the links supplied. The database 1 contains 713 face images of 11 subjects (people). Each folder contains images of one particular subject, and hence considered as a class. Database 2 contains images 60000 colour images in 10 classes, with 6000 images per class.

   Database 1 (Face Dataset): https://drive.google.com/file/d/1jxJ9nRrzjtRD-1x5AH
   6pqZYcWjhJ3C8R/view?usp=sharing
   Database 2 (CIFAR 10): https://www.cs.toronto.edu/ kriz/cifar.html

   Protocol: For database 1 Randomly split your data set into 70% training and 30% testing instances. For database 2 there are 50000 training images and 10000 test images. Perform 5 fold cross validation,
   for database 1: in each fold, randomly put 70% of the data in training, and rest for testing, and
   for database 2: in each fold randomly put 50000 of the data in training and rest for testing.

   Now perform the following tasks:

   1. Use Linear discriminant analysis (LDA) to find out the best projection directions.
   2. Project your data on the new projection matrix given by LDA.
   3. Classify the projected data using 5 fold cross-validation, report the mean and standard deviation of classification accuracy. Draw the ROC curve. How does the results on the 2 databases differ and why ?
   4. Perform PCA on the same data and report accuracy and standard deviation over the 5 folds. Compare and analyze the results obtained by PCA and LDA.
   5. If you perform LDA on the PCA projected data, find out the classification performance for both the databases. Analyze the result. If we perform the process the other way round, what performance do you get and why?

   Submission files: Please submit a report for all your analysis and observations only and only in the PDF format. Other format will not be evaluated. Along with the report, submit following files:

   1. main.py/.m: To read, to make partitions of the data and to call the training and testing functions for both PCA and LDA.

   2. train_PCA/LDA.py/.m: To train LDA/PCA on the input data set. 3. test_PCA/LDA.py/.m: To perform classification using LDA/PCA on the input data set for all the questions.

   4. Please submit your trained model LDA/PCA projection matrix file for 70-30 train-test split.

   NOTE: You are allowed to use only PYTHON and MATLAB for the programming assignment.

5. **(10 points)** Consider you are having a very old mobile phone, with a worn-out speaker and you will have to attend an important interview using that phone. However, instead of hearing the actual words of the interviewer, you hear some noisy inputs (actual words + noise). In this assignment, we will try to infer the actual words spoken by the interviewer from this noisy data. Our task is to decode an English phrase from a non-text noisy observation using Hidden Markov Model (HMM). In a practical scenario, these observations can take any real valued number, measuring the noisy signal we receive through the mobile phone. However for simplicity sake we will assume that the observations that we receive take only binary values. But the high level concepts of the problem remain intact.

   Consider a HMM with 26 number of hidden states (i.e., $S_t = \{1, 2, 3, \dots, 26\}$) and binary observations in each state $O_t = \{0, 1\}$. The three parameters of the HMM are attached with this assignment. The assignment file also contains an observation sequence of length 60000 in files *observations_art.txt* and *observations_test.txt*. Use the Viterbi algorithm (as discussed in class) to compute the most probable sequence of hidden states conditioned on this particular sequence of observations.

   Submission files: Please submit following files for this question:

   1. main.py: To read HMM parameters, to call Viterbi algorithm etc.
   2. viterbi.py: Implements Viterbi algorithm. It takes observation sequence and HMM parameters as input and outputs the respective state sequence.
   3. Output sequence: write your output sequence in out_seq.txt file for observation sequence given in file *observations_test.txt* and submit it with other files.

   Note: you are not allowed to use any library for HMM. You can use PYTHON or MATLAB for implementation. To cross-check your answer, assume the hidden states $\{1, 2, ..., 26\}$ represent the alphabets $\{a, b, ..., z\}$ respectively. If you decode your most probable state sequence using Viterbi algorithm (ignoring repeated states), you will find out its meaning. The state sequence from *observation_art.txt* should come out to be *art*. The state sequence corresponding to observations in *observations_test.txt* is not given and you have to report its state sequence in file *out_seq.txt*.