Statistical Machine Learning Assignment: 2 MM:50+85

Note: You are allowed to discuss but the final answer should be your own. Any instance of cheating will be considered as academic dishonesty and penalty will be applied. The question numbers given below are from the second chapter of the second edition of the printed book. Make sure that you are attempting the correct problem.

• Theory questions:

1) Section 2.3, question 3 **10 Marks**
2) Section 2.3, question 9 **10 Marks**
3) Section 2.4, question 14 (a-c) **15 Marks**
4) Section 2.8, question 37 **15 Marks**

• Programing Questions:

From the book: computer Exercise

1) Section 2.5, question 2 **10 Marks**
2) Section 2.5, question 4 **10 Marks**
3) Section 2.8, question 7 **10 Marks**
4) Section 2.8, question 8 **30 Marks**

5) Design a Naive Bayes classifier for classifying the numbers in the MNIST handwritten digit recognition databases (http://yann.lecun.com/exdb/mnist/). **3+2+2+3 Marks**

A. Design a classifier to distinguish between 0 and 1. Use the training database of available for both to design the classifier. Use complete training data of 0's and 1's for training and use complete testing data of 0's and 1's for testing.
B. Use the testing database to compute the classification accuracy ((TP+TN)/ (TP+TN+FP+FN)) of the Bayesian model.
C. Plot the ROC curves between FAR vs GAR.
D. Repeat the above three parts for classifying the digits 3 and 8. However, select only 10% of digit 3's data and 90% of digit 8's data in both training and testing randomly. Obtain classification accuracy for classification of 3 and 8 by the method mentioned in part (B). Do you think the method to obtained accuracy can be treated as a unbiased metric? Give reasoning to support your argument. If no, obtain accuracy to make the metric unbiased.

Submission files:

Along with the report, submit following files:

1. main.py: To read, to make partitions of the data and to call the training and testing functions.
2. train_model.py: To train the classifier on the input data set. This script has following arguments: a) Input: (i) trainset (training set of your split), (ii) labels (class information of

the training data) b) Output: model (a structure having required probabilities for both the classes, example. model.low represents the posterior probability for low scored TA class)

3. test_model.py: To perform classification on the input data set. The script has following arguments: a) Input: (i) testset, (ii) testLabels (iii) model (trained model), b) Output: TPR, FAR and percent accuracy over multiple thresholds so that you can plot ROC from the values.
4. Please submit your trained model file. (unless explicitly specified in the question).

6) The database supplied (link given) contains 713 cropped images of 11 subjects (people). Each folder contains images of one particular subject, and hence considered as a class.

6+2+5+2 Marks

a) Learn three different PCA representations (usage of inbuilt functions for PCA is not allowed) on these images by preserving 90%, 95% and 99% eigenenergy respectively.
b) Visualize (as images) the top eigenvectors in the PCA projection matrix. What can you analyze using these visualizations.
c) Create the reduced dimensional representation for each image by preserving 99% of eigenenergy. Use any classifier (you can use an inbuilt function) to classify the images using the reduced dimensional representation. Create a 50-50% train and test split, that is 50% of the images of one subject goes into the train set and the rest 50% goes into the test set.
d) Compare the classification accuracy of the PCA transformed images and the original images. What can you infer from this comparison.

Link for Database:
https://drive.google.com/file/d/1jxJ9nRrzjtRD-1x5AH6pqZYcWjhJ3C8R/view?usp=sharing

Submission files:
Along with the report, submit following files:

1) The codes for creating the PCA projection matrix and finding out the reduced dimensional representation.
2) The code for classification and testing.
3) Submit your PCA projection matrices for 90%, 95% and 99% eigenenergy respectively.

Submission format: Please submit a report for all your analysis and observations only and only in the PDF format. Other format will not be evaluated. You are allowed to use only PYTHON or MATLAB for the programming assignment. All the graphs should have labels (on the axis), legends, title. You should also try to combine the graphs and plots for comparison and better representation.