Pattern Recognition –HW#3

About the Assignment

The main aim of the assignment is to learn feature extraction and classification and feature extraction. Contributions of this lab are;

- Ability to analyze the separable condition of features.
- Ability to analyze the non-separable condition of features.
- Understanding idea of feature extraction in the machine learning.

Part1:

In this HW, you are expected to make an experiment with L1-distance classifier on Principal Component Features (PCA) features, explained in class. Recall the previous lab, in like manner, the experiment will be conducted on the real life problem, called a classification problem among multi categories. The experiment is about Caltech-101 datasets, there are 15 classes, and each one contains different number of samples. The aim is to analyze the discriminative capability of PCA features for a classification problem. After extraction PCA features, we will train with L1-distance classifier by one-against-all methodology,.

According to one-against-all methodology, for L1-classifer, it means that the lowest distance score refers to predicted target class of processed sample.

Part2: Feature Extraction and Classification Stages

Download data from the link:

Stage1: *Read all images from related directories*.

Each image is in the 128x128x3 format. You should read and resize them to 64x64 format. Once you read the images, you have to convert them into vector format. Then it will become a vector with the format of 4096x1 size for each image. If there are n classes, for each class, you will have a matrix with 4096xn format.

Stage2: Select Eigenvalues and Eigenvectors of (PCA).

From stage 1, you have obtained 4096xn matrices for each class. Now, we will apply PCA on extracted matrices. For this purpose, you have to implement process explained in the course note (08-PCA-Example.pdf).

PCA Step1:

Extract covariance matrix for each class data (4096xn). Each covariance matrix will be nxn format.

PCA Step2:

Extract eigenvalues and eigenvectors from covariance matrices. There will be n eigenvalues and n eigenvectors. Each eigenvector is in the nx1 format.

PCA Step3:

Sort eigenvalues in descending order. Then select the eigenvectors corresponding to three (3) maximum eigenvalues. You will have a matrix containing nx3 eigenvectors

Stage3: *Extract PCA features by using Eigenvectors*.

In previous stage, you have extracted eigenvectors. Now, you will project the each class matrix (4096xn) into the eigenvector matrix of nx3 format. For this purpose, you have to multiply with eigenvector matrix.

After projections, each class is represented with 4096x3 feature matrix. These features are called as PCA features, or projections.

For example, each train classes will be represented as:

The class-1 (c1) is represented with 4096x3 feature matrix

The class-2 (c2) is represented with 4096x3 feature matrix

…...

……

The class-14 (c14) is represented with 4096x3 feature matrix

The class-15 (c15) is represented with 4096x3 feature matrix

C1: 4096x3

C2: 4096x3

….

C14: 4096x3

C15: 4096x3

Stage4: Test an Image

In order to predict the class label of a test sample, you should follow below steps.

Step1: Read image from test directory

Step2: Convert into the test vector format 4096x1. Then, combine n times the test vector. The size will be 4096xn. Then, you have to multiply with eigenvector matrix of nx3. Each test sample is again represented with 4096x3 feature matrix.

Step3: Compute sum of absolute distance between train feature matrix and test feature matrix.

Example: For a test sample (S), compute following distances and the class label will be determined with minimum distance.

distance1 = np.sum(np.abs(S-C1))

distance2 = np.sum(np.abs(S-C2))

….

distance14 = np.sum(np.abs(S-C14))

distance15= np.sum(np.abs(S-C15))

The label of test sample will be class name of corresponding minimum distance.

Submit the Assignment

Ex: No_Name_Surname_HW#.zip

Hint

You can look at the implementations available in internet.