# SMAI Assignment 2

Faizan Farooq Khan, Roll No 20171209

3 November 2019

## 1 Problem-1

### 1.1 Eigenfaces

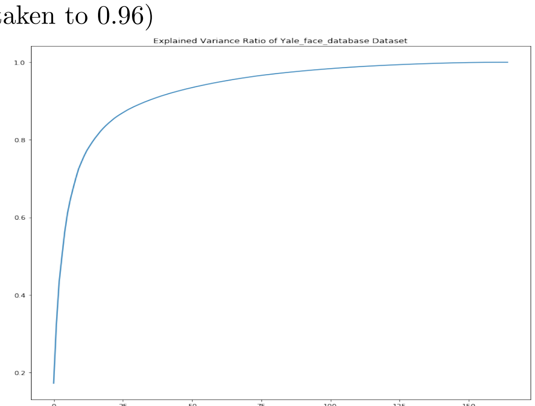
Eigenfaces are the eigenvectors of the covariance matrix of face images, and are used in the field of computer vision in facial recognition, and in general pattern recognition.

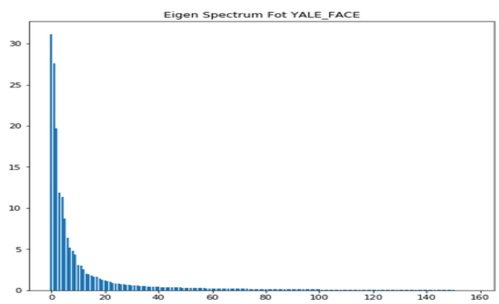
### 1.2 Number of Eigenfaces required

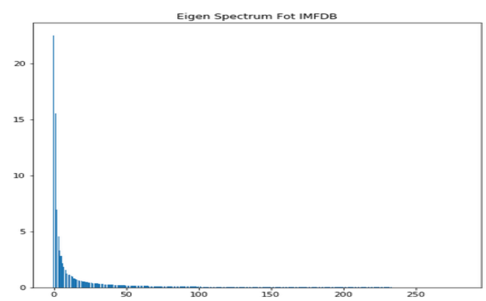
The number of eigenfaces required to satisfactorily represent the images in each of the datasets is as follows:-

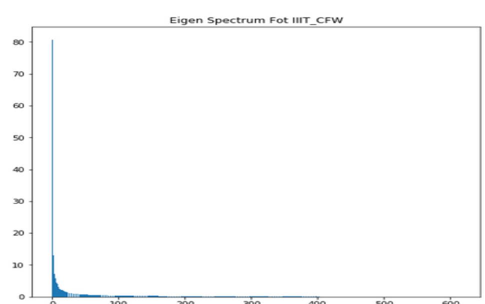
IMFDB :- 100
Yale Face Database :- 50
IIIT-CFW :- 110

The numbers were determined by analysing the plotted eigen value spectrum of the the dataset, all the values above the and taking the number of eigen values that are above a threshold. More specifically we construct the **explained variance ratio graph**. And take the eigenvalues that sums up to a threshold (taken to 0.96)



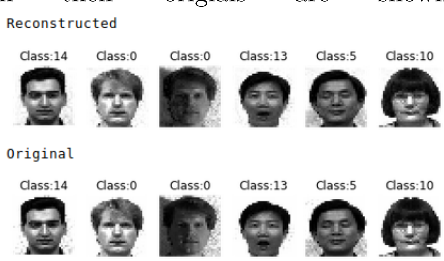
### 1.3 Reconstructed Images and related problems

To see which person/identity is difficult to represent with fewer eigen vectors we calculate the reconstruction error for each of the class separately and then we take the classes that have reconstruction error greater than 96% of max reconstruction error from all the classes .

In IMFDB, the most difficult person to represent is Shilpa Shetty, AkshayKumar, AmitabhBachan,KatrinaKaif and Shahrukh Khan.

In IIIT-CFW the most difficult person to represent is NarendraModi and ManmohanSingh.

Some reconstructed images along with their origials are shown below.




## 2 Problem 2

### 2.1 Classification Results

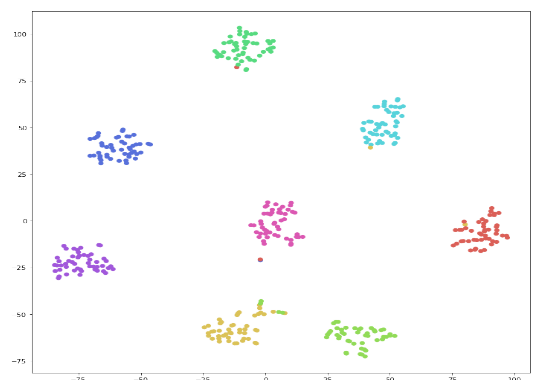
MLP classifer is used to get the performance for various features. Since in MLP weights are initialized randomly features some set of features perform almost same. Concatenation of features performs good like concatenation of all original features or concatenation of LDA and KLDA.

Analysing the dataframe in the notebook and taking the original features (no concatenation) For every dataset ResNet feature performs best and KPCA worst.

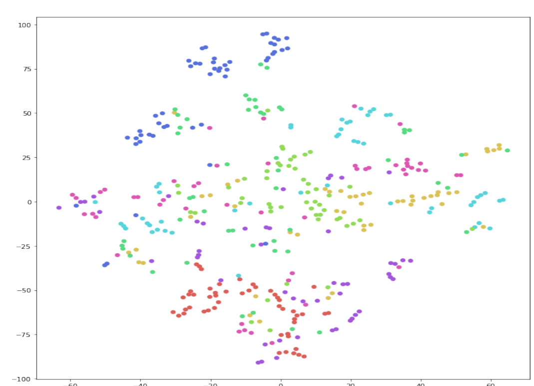
## 3 Problem 3

### 3.1 t-SNE Plots

Yes similar classes are clustered together but some outliers are still there. t-SNE depends on the feature t-SNE transformation applied. Different features produced different t-SNE plots. Using **KLDA** as input feature for t-SNE, clearly different class clusters are formed which are not seen in **PCA**. (in the plot) One other parameter on which the t-SNE plots depends is the **perplexity**. Perplexity balances attention between local and global aspects of the data.Larger / denser datasets requires a larger perplexity ranging from 10-50a and since our dataset in not a large one , I selected the perplexity value to be 6 .



t-SNE plot of IMFDB Database using KLDA features



t-SNE plot of IMFDB Database using PCA features

# 4 Problem 4

## 4.1 Formulation

For the problem, we first compute different features(using PCA, LDA, or other variants), and train a KNN classifier. Then, given a data point X and the corresponding class ID, we find the predicted ID(using our KNN classifier), and return yes or no according to whether it matches the given class ID or not.

The metrics could be the accuracy i.e. the number of times our output is correct w.r.t to the total number of tests.

## 4.2 Analysis

Metric used to compare performance is accuracy

$$Accuracy = \frac{No.of Correct Predicitions}{Total Samples} \quad (1)$$

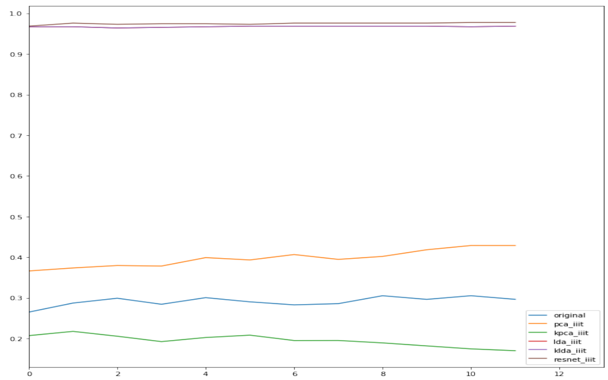
Another metric that used is precision

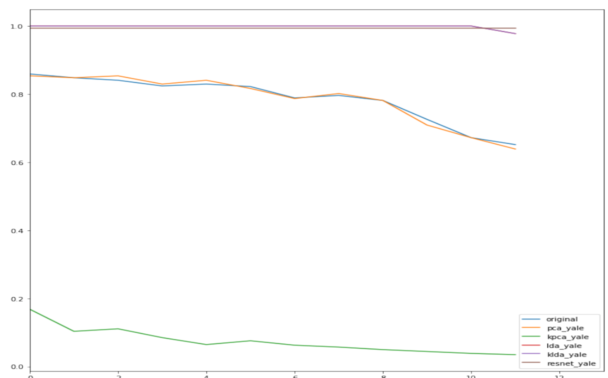
$$Precision = \sum_{C \epsilon Classes} \frac{TruePos.(C)}{TruePos.(C) + FalseNeg.(C)} \quad (2)$$

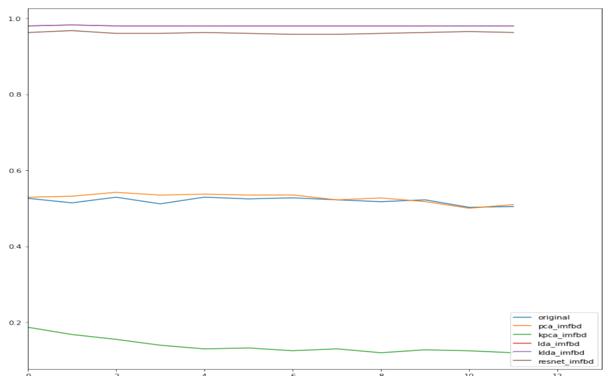
$$Precision = \frac{Precision}{No.of Classes} \quad (3)$$

## 4.3 Analysis

Plotting for the accuracy with varying **K** for different input features for all the datasets provided.



# 5 Problem 5

## 5.1 Problem Statement

The problem I have chosen is that of emotion prediction - given an image, we need to predict the emotion of the image. The problem is evidently not trivial - given a photo of a person, it is difficult to identify the features that may help us predict the emotions of the person. As such the problem may be intractable for non machine learning methods.

## 5.2 Applications

If we are able to build a system that can detect the emotion of a person with a single photo, it may be put to use in diverse ways. Some of the applications are :-

- Making Cars Safer and Personalized :- For example ,Using facial emotion detection smart cars can alert the driver when he is feeling drowsy.
- Facial Emotion Detection in Interviews :- All recruiters would appreciate technology that can tell them what a candidate is feeling.
- Testing for Video Games :-Facial expressions while playing is a great metric to understand if the game is successful in making your experience enjoyable.
- Blind People :- Facial Emotion Recognition allows us to make a narrator for blind poeple to help them observe a picture.

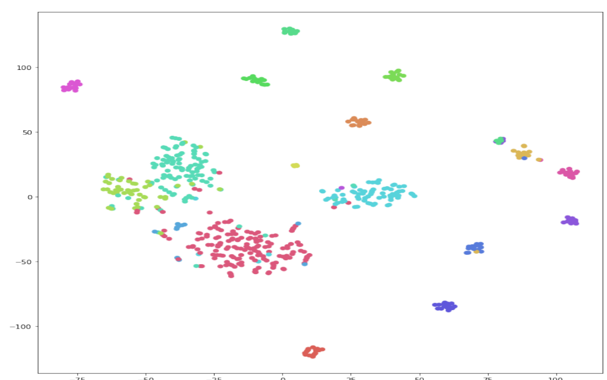
## 5.3 Model Pipeline

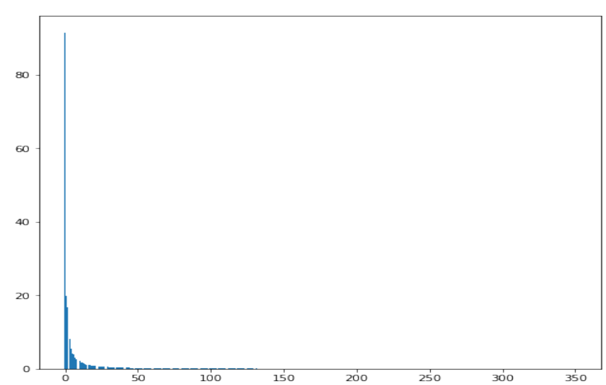
- We first take the YALE Face Dataset and IMFDB datasets along with the files 'emotion.txt'. We then concatenate the two datasets to form a single dataset.
- Use Klda features as our training set(additionally, resent should not be used as it is a classifier , so it'll give bad results compared to KLDA)
- Split the data into training and testing sets, then train our classifier. I use MLP on the features and I end up with accuracy 0.90.
- Validate our classifier on the testing set.

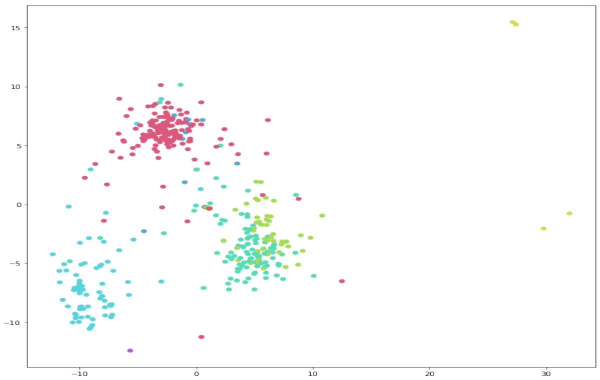
## 5.4 Results

The MPL classifier with KLDA features given us cross validation score and accuracy. The accuracy during training time was 90% and the testing accuracy came about 92%.

The t-sne , eignevalue spectrum and isomap plots are shown below :






from 2019-11-03 22-22-21.png from 2019-11-03 22-22-21.png