SMAI Assignment 2

Shikhar Shasya, Roll No.: 20171029

2 November 2019

1 Basic Questions

1.1 Eigenfaces

Eigenfaces is the name given to a set of top k eigenvectors when they are used in the computer vision problem of human face recognition.

1.2 Number of Eigenvectors required:

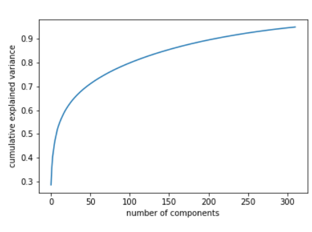
The number of eigenfaces required to satisfactorily represent the images in each of the datasets is as follows:-

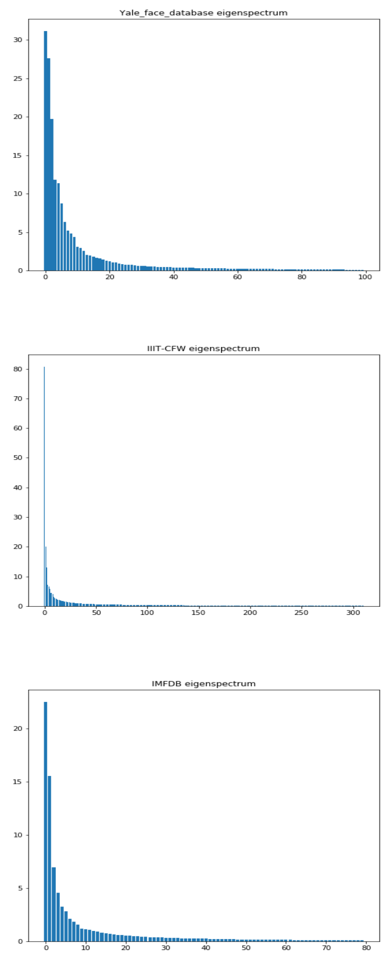
IMFDB :- 80

Yale Face Database :- 100

IIIT-CFW :- 311

These numbers were determined by analysing the eigenspectrum of the each dataset, i.e all eigenvectors corresponding to eigenvalues more than 1e-1 are plotted in eigenspectrum. It was observed that the threshold captured greater than 90 percent of the original data in all the three datasets More specifically we construct the **explained variance ratio graph**. This graph plots how much percentage of data is captured vs no of components. Plot of cumulative explained variance vs ncomponents for IIIT-CFW dataset.



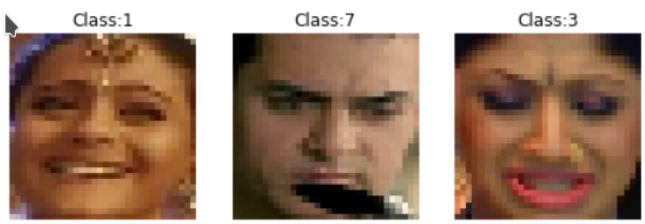


Eigenspectrum for the datasets

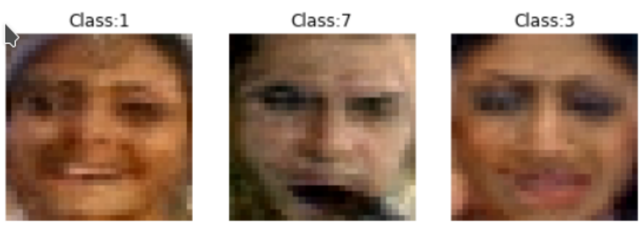
1.3 Reconstructed Images and related problems

To see which person/identity is difficult to represent with fewer eigen vectors we calculating the reconstruction error for each class separately and taking the classes that have reconstruction error greater than 97% of max reconstruction error of all the classes.

In IMFDB, the most difficult person to represent is Shilpa Shetty, AkshayKumar, AmitabhBachan. In IIIT-CFW the most difficult person to represent is Aamir Khan, Dwayne Johnson and Aishwariya Rai.
Original Images example:



Reconstructed Images example:



IIIT-CFW dataset is not easy to represent with fewer eigenvectors because lesser number of eigenvectors is unable to fully capture our original data. Suppose we take 40 components for IIIT-CFW dataset instead of threshold value 311, we get high reconstruction error since these many components are not able to capture data fully.

2 Problem 2

2.1 Classification Results

The classifer used to get the performance for various features is MLP. 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 **KPCA and KLDA**.

Analysing the dataframe and taking the original features (no concatenation), for every dataset ResNet feature performs best and KPCA worst. The following tables do a comparitive study of performance for the three datasets using various feature spaces. The classifier used was MLP.

IIIT-CFW Dataset:

	Method	Reduced Space	Error	Accuracy	F1-score
0	PCA	100	42.2222	57.7778	0.558545
1	KPCA	100	51.8519	48.1481	0.460546
2	LDA	7	4.44444	95.5556	0.945718
3	KLDA	7	4.44444	95.5556	0.945718
4	VGG	4096	31.1111	68.8889	0.638534
5	RESNET	2048	2.96296	97.037	0.970851
6	RESNET+VGG	6144	1.48148	98.5185	0.983046
7	KPCA+KLDA	107	3.7037	96.2963	0.95918
8	ALL FEATURES	6358	1.48148	98.5185	0.983046

Yale_face_database Dataset:

	Method	Reduced Space	Error	Accuracy	F1-score
0	PCA	100	6.06061	93.9394	0.933333
1	KPCA	100	9.09091	90.9091	0.887619
2	LDA	10	0	100	1
3	KLDA	10	0	100	1
4	VGG	4096	48.4848	51.5152	0.49
5	RESNET	2048	0	100	1
6	RESNET+VGG	6144	0	100	1
7	KPCA+KLDA	110	0	100	1
8	ALL FEATURES	6364	0	100	1

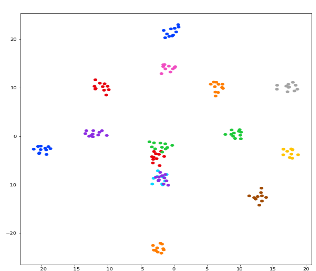
IMFDB Dataset:

	Method	Reduced Space	Error	Accuracy	F1-score
0	PCA	100	15	85	0.84558
1	KPCA	100	18.75	81.25	0.810381
2	LDA	7	5	95	0.935107
3	KLDA	7	5	95	0.935107
4	VGG	4096	12.5	87.5	0.864513
5	RESNET	2048	3.75	96.25	0.961602
6	RESNET+VGG	6144	0	100	1
7	KPCA+KLDA	107	5	95	0.935107
8	ALL FEATURES	6358	0	100	1

3 Problem 3

3.1 t-SNE Plots

Plot is included in the end. Yes similar classes are clustered together but some outliers are still there. Using **KLDA** as input feature for t-SNE, clearly different class clusters are formed which are not seen in **PCA**. (in the plot) **LDA and KLDA** form good clusters in the 2D-plot because the goal of LDA is to maximise interclass variance and minimise intraclass variance. Because of this property of LDA we observe that it forms good clusters in the t-SNE plots. On the other hand methods like **PCA and KPCA** show heavily merged clusters in the 2D-plot because PCA aims to only reduce the dimensionality of the problem, it is not supervised. The other parameter on which the t-SNE plots depends on the **perplexity**. Perplexity balance attention between local and global aspects of your data.Larger / denser dataset requires a larger perplexity ranging from 10-50. Perplexity value choosen for given dataset is 8.



t-SNE plot of Yale Face Database using KLDA 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 false or correct. Also some emperical results are also shown when the code is run for the knn classifier. The emperical results predicted Yes for most of the time in LDA/KLDA and No in the case of PCA/KPCA. RESNET also predicted yes most of the time and had highest accuracy across the datasets among non-concatenated feature spaces.

4.2 Analysis

Metric used to compare performance is accuracy

$$Accuracy = \frac{No.ofCorrectPredicitions}{TotalSamples} \quad (1)$$

Another metric that used is precision

$$Precision = \sum_{C \epsilon Classes} \frac{TP(C)}{TP(C) + FN(C)} \quad (2)$$

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

Comparitive study of performance of various feature spaces using the knn classifier.

IIIT-CFW Dataset:

	Method	Reduced Space	Verification Error	Accuracy	Precision
0	PCA	100	59.2593	40.7407	0.529153
1	KPCA	100	57.037	42.963	0.529365
2	LDA	7	2.96296	97.037	0.97197
3	KLDA	7	2.96296	97.037	0.97197
4	VGG	4096	31.8519	68.1481	0.625551
5	RESNET	2048	1.48148	98.5185	0.985714
6	RESNET+VGG	6144	0.740741	99.2593	0.991667
7	KPCA+KLDA	107	2.96296	97.037	0.97197
8	ALL FEATURES	6358	0.740741	99.2593	0.992188

IMFDB Dataset:

	Method	Reduced Space	Verification Error	Accuracy	Precision
0	PCA	100	36.25	63.75	0.74023
1	KPCA	100	36.25	63.75	0.74023
2	LDA	7	5	95	0.946825
3	KLDA	7	5	95	0.946825
4	VGG	4096	10	90	0.9
5	RESNET	2048	3.75	96.25	0.96875
6	RESNET+VGG	6144	3.75	96.25	0.959722
7	KPCA+KLDA	107	5	95	0.946825
8	ALL FEATURES	6358	1.25	98.75	0.9875

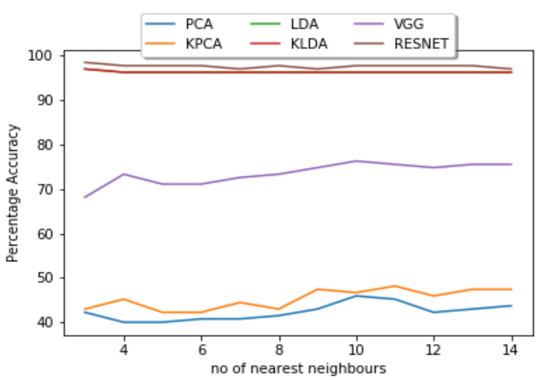
Yale_face_database Dataset:

	Method	Reduced Space	Verification Error	Accuracy	Precision
0	PCA	100	12.1212	87.8788	0.923333
1	KPCA	100	12.1212	87.8788	0.923333
2	LDA	10	0	100	1
3	KLDA	10	0	100	1
4	VGG	4096	57.5758	42.4242	0.427381
5	RESNET	2048	0	100	1
6	RESNET+VGG	6144	6.06061	93.9394	0.961111
7	KPCA+KLDA	110	0	100	1
8	ALL FEATURES	6364	0	100	1

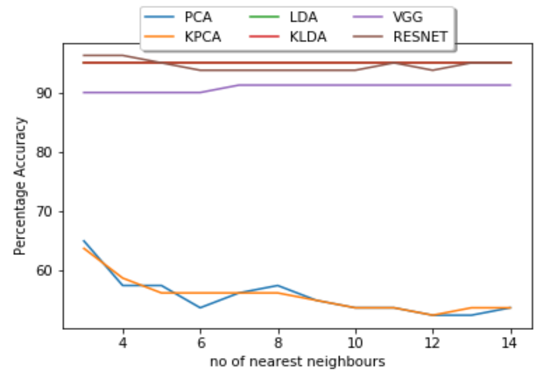
From these tables, one can see that RESNET performs the best among all the single feature spaces used. Also concatenation of feature spaces gives excellent results. KPCA performs marginally better from simple pca for the IIIT-CFW dataset. LDA and KLDA give the same performance for each dataset. PCA and KPCA are unable to form good clusters since they do unsupervised learning and hence their performance for knn is mediocre. On the other hand, LDA and KLDA form good clusters and hence show excellent performance.

4.3 Analysis for K

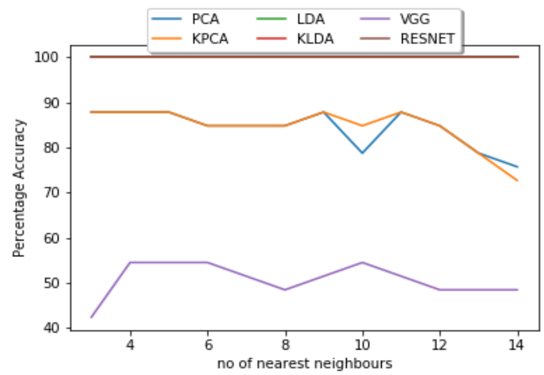
Plot for the accuracy with varying **K** for different input features for the dataset is plotted.



IIIT-CFW dataset



IMFDB dataset



Yale dataset

5 Problem 5

5.1 Problem Statement

The problem I have chosen is that of Emotion Classification - given an image of a person we need to predict the emotion and classify it to the set of emotions. 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 emotion 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 :-

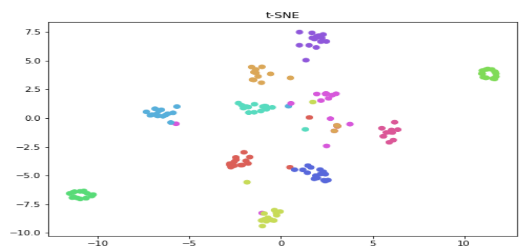
- Security :- Security systems in areas that are gender restricted can be automated using the gender identification system.
- Face Matching :- If we, for example, wish to check whether two given photos are of the same person, then, instead of directly using more advanced methods, we may use gender identification as a preliminary to eliminate many options.
- Motion Tracking :- As a follow up to face matching, we may extend our system to track motion of a single person in a large group of people without resorting to costlier methods.

5.3 Model Pipeline

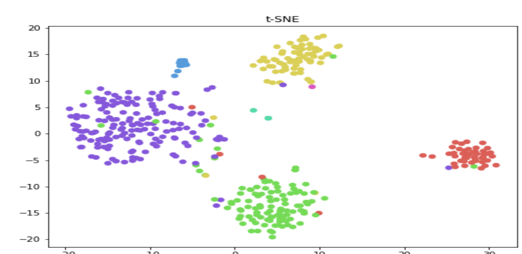
- We first take the IIIT-CFW and IMFDB datasets and find the labels for different types of emotion given in input.
- We now define our feature space. LDA featurs space used for training our model.
- After loading the input data and input labels for the input data, we perform a test-train split.
- We now train our model on train data using MLP classifier.
- Validate our model using metrics like accuracy and K-fold score.
- We now make t-SNE, PCA and Isomap plots of LDA feature space.
- Examples of correct and wrong prediction printed.

5.4 Results

The t-SNE visualization of the data, in 2D shown below.

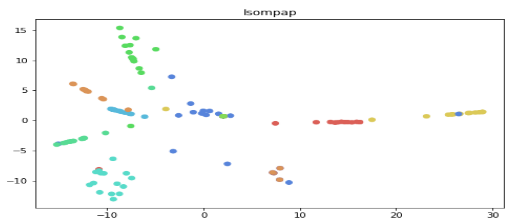


Yale Dataset

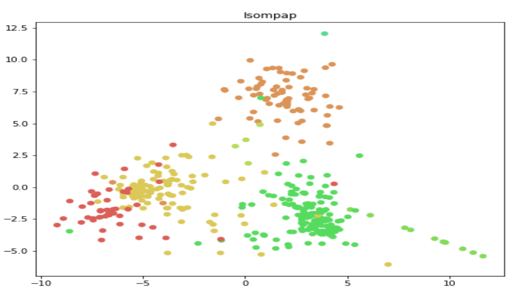


IMFDB Dataset

The Isomap visualization of the data, in 2D shown below.



Yale Dataset



IMFDB Dataset

5.5 Evaluation Metrics

Accuracy was calculated using the formula given below-:

$$Accuracy = \frac{No.ofCorrectPredicitions}{TotalSamples} \tag{4}$$

Also to ensure our model doesn't overfit, we do a K-cross fold validation. Formula given below-:

$$K{-}Foldscore = \sum_{K\epsilon Kthtestingset} \frac{Correctpredictions(K)}{TotalSamples(K)} \tag{5}$$

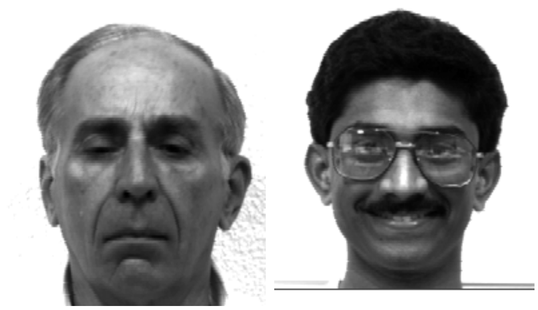
$$K - FoldScore = \frac{K - FoldScore}{No.oftestingsets} \tag{6}$$

5.6 Metric Results

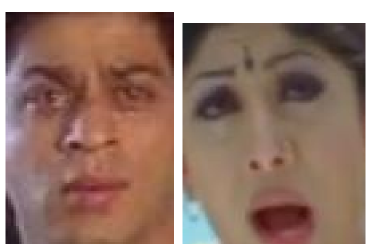
The classifier used for predicting the emotions was MLP. Testing accuracy for the Yale dataset was found to be 84.84 and Training accuracy was found to be 93.93. Similarly for IMFDB it was found to be 93.75 and 95.31 percentage.Since there is not a huge gap in values for training and testing accuracy we have avoided the problem of overfitting our data. Also since the model predicts the unseen data and trained data with good accuracy values we have also avoided underfitting.

Cross K-Fold Accuracy with k =5: for Yale dataset: 89.09 Cross K-Fold Accuracy with k =5: for IMFDB dataset: 94.20

5.7 Quantitative Results



In the first image the emotion predicted was supposed to be sad but our model predicted it as sleepy. In the second image emotion predicted was supposed to be happy but our model predicted it as sad.



In the first image the emotion predicted was supposed to be sad and our model predicted it correctly but becuse the truth label given in input was neutral, our model 'misclassified'. In the second image emotion predicted was supposed to be angry but our model predicted it as sad.