Principal Component Analysis and K-Means Clustering-Biometrics

Harshit Vijayvargia
Department of Computer & Information Science, University of Florida
Gainesville, USA
hvijayvargia@ufl.edu

Abstract— **The purpose of this report is to discuss about the implementation of a facial recognition system by employing a widely-used technique known as PCA (Principal component Analysis). By carrying out experimental study, I have demonstrated how PCA can make a facial recognition system efficient. Along with PCA, I have also discussed about Soft Biometric classification which is a technique used to improve systems recognition performance. This is implemented using K-Means clustering algorithm. In the end, through empirical study I have shown that clustering with the Principal components instead of the original image features does not improves cluster quality; however, it makes the clustering algorithm more efficient.**

I. Introduction

With the development of information technology, biometric identification technology has attracted more attention especially the facial recognition systems. Facial recognition (or face recognition) is a type of biometric software application that can identify a specific individual in a digital image by analyzing and comparing patterns. These systems are commonly used for security purposes but are increasingly being used in a variety of other applications such as HCI, Gaming etc.

When we are working in real time environment accuracy is not just the major factor in designing an efficient algorithm for facial recognition, there is one more thing equally important which is computation time. A lot of research work is done in this field to develop algorithms which are efficient in terms of computation time and recognition performance. Our objective behind using PCA is reducing time complexity of algorithm. It is useful when we have large number of redundant features in our data set as it eliminates them and only keep those features which contribute towards the maximum variance in data. There are other algorithms too like LDA (Linear discriminant analysis) which is also a well-known scheme for feature extraction and dimension reduction. In this we project the data onto a lower-dimensional vector space such that the ratio of the between-class distances to the within class distance is maximized, thus achieving maximum discrimination. Neural networks backpropagation algorithm is another way which is robust and works well when we have complex sensory input.

To further improve the performance of recognition system there is a technique known as soft biometric classification employed in face recognition algorithms. It classifies the data based on a soft biometric trait which in our case is Gender. By providing some information about the user, It reduces the search space for recognition and helps in making algorithm more efficient. This classification can be done through any clustering algorithm. The important part is how this is integrated with the recognition algorithm.

II. Principle Component Analysis

PCA is designed to reduce the dimension of large data matrix to a lower dimension by retaining most of the original variability in the data [5]. It retains the maximum variance while minimizing the least square reconstruction error. The steps I followed for implementing PCA are shown below:

Suppose V is an N^2x1 vector, corresponding to an

NxN face image I.

The idea is to represent an image into a low-dimensional space i.e as a linear combination of product of eigenfaces ($u_1,u_{2\ldots.}u_k$) and weight parameters ($w_1,w_{2\ldots.}w_k$).

Image(I) = $w_1u_1 + w_2u_2 + \cdots w_ku_k$ ($k<<N^2$)

[2] *eigenfaces computation*:

Step 1: obtain face images $I_1, I_2, ..., I_M$ (training faces)

Step 2: represent every image I_i as a vector V_i

Step 3: compute the average face vector μ:

$$\mu = 1/M \sum_{n=1}^{M} \mathrm{V}n$$

Step 4: subtract the mean face:

$$x_i = V_i - \mu$$

Step 5: compute the covariance matrix C:

$$C = \sum_{n=1}^{M} x^k x = A^t A$$

Step 6: compute the eigenvectors u_i of AA^T

The matrix AA^T is very large if size of images is too large then computational complexity will be too high, so to improve efficiency we will compute eigenvectors v_i of A^TA which is smaller dimension matrix and then will evaluate u_i through this relationship:

$u_i = Av_i$.

III. METHODOLOGY

Once we have performed PCA we can proceed forward to perform facial recognitions using features in new coordinate system. Each image can be represented as the combination of eigenfaces multiplied with eigen coefficients. The whole recognition algorithm is shown in the following block diagram:

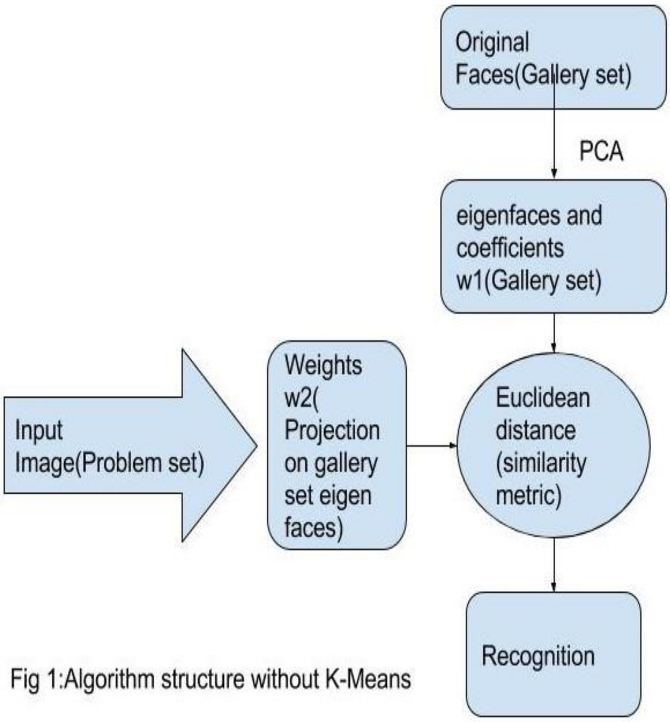

Original Faces(Gallery set)
PCA
eigenfaces and coefficients w1(Gallery set)
Input Image(Problem set)
Weights w2(Projection on gallery set eigen faces)
Euclidean distance (similarity metric)
Recognition

Fig 1:Algorithm structure without K-Means

It starts with computing principal components of gallery set. The images in gallery set are of dimension 50*50 so after doing PCA we obtain an eigen vector matrix of dimensions 2500*2500. Next step is determining the number of principal components(k) which contributes towards the maximum variance.

For determining K, I plotted the graph below which at any point shows the percentage of total variance contributed by top k principal components with highest eigen values (Variance).

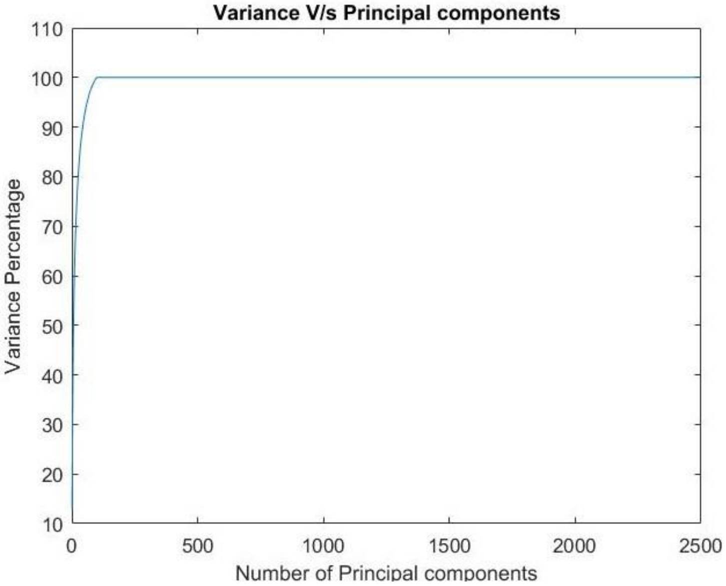


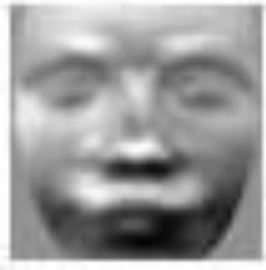
Figure 2

From the above graph, it can be clearly seen that top 100 eigen vectors/eigenfaces with maximum variance sum to 100% of total variance. So, we don’t need to consider values ok K>100.

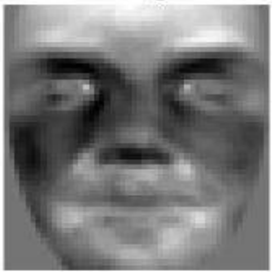
Inference from principal components:

The first three principal components(eigenfaces) are shown below*:*

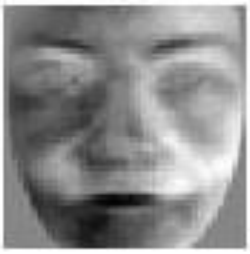
First Eigenface



Second Eigenface



Third Eigenface



No. of Principal Components	Percentage of total Variance Contributed
1	12.9
2	22.6
3	31.3

When eigen vectors are sorted in decreasing order based on their eigen values then the **First Eigen Vector** represents the direction of maximum variance of data with minimum re-construction error (Sum of squared distance between data points and their projections along principal component).

The variance is given by the corresponding eigen value. This component also captures the variation in illumination in the horizontal direction. This was observed from the variation of images along first Eigenface.

The second eigenface which is orthogonal to first eigenface explains residual variance in the data. It also captures variations in brightness (From dark to bright). This could be also observed from the variations of images along second Eigenface.

Once we know the range of K we can perform recognition by taking different values of K within that range. The recognition procedure is shown in Fig 1. I will perform recognition on problem set which has 200 images. For each image of problem set I will compute it’s eigen coefficients on PCA Subspace of gallery set and will compare them with eigen coefficients of gallery set using Euclidean distance as the similarity metric. For values ok K [10,200] I obtained the following plot:

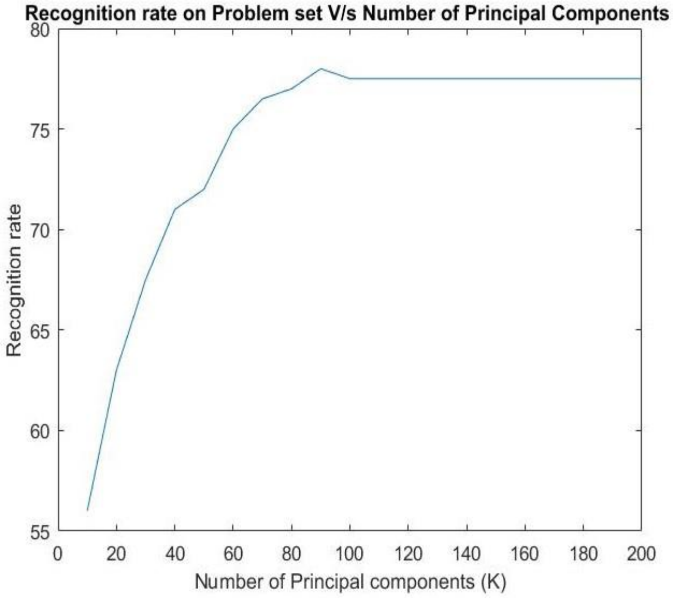


Figure 3

Observations:

The Recognition rate is increasing as I am increasing the number of principle components. The maximum recognition rate achieved is 78% at K=90. The reason behind this is the decrease in reconstruction error as we increase the principal components. We can also observe that, initially the rate at which recognition rate increases is higher due to higher variance of first principal components

Although at some instances it decreases due to addition of principle components, like at K=90 it is maximum (78%) and then on increasing K it decreases to 77.5 and becomes static. This is due to addition of principal components which are not reflecting the variations similar to problem set.

Also, when K reaches 100 the recognition rate reaches to 77.5% as could be observed from the above graph. As discussed earlier top 100 eigen vectors contributes to 100% of the total variance in image so increasing principal components will not improve the recognition performance beyond this point. This can be empirically justified from the recognition rate which I evaluated using the original image with all features which also comes out to be 77.5%.

Reason behind 78% accuracy with & without PCA:

The reasons behind this recognition performance are:

a) Eigenface representation are dependent on original images so under varying pose and elimination its recognition rate can decrease. This conclusion can be drawn from observing images in gallery set(*Figure 4*) and problem set(*Figure 5*) which differ in expressions:

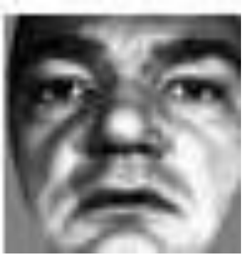


Figure 4

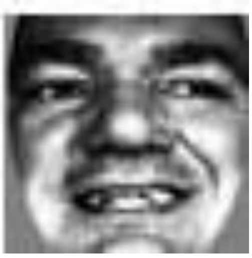


Figure 5

b) PCA could not capture the invariance in images unless the training data explicitly provides this information. [3].

c) When I measured recognition performance without using PCA considering all features, recognition rate comes out to be almost same (77.5%). One more reason behind this is the similarity metric we are using which is Euclidean distance in both cases. It can only determine a linear relationship between variables. if there is change in pose or expressions of an individual then there's a possibility that it may give wrong results.

IV. K-MEANS CLUSTERING ALGORITHM

[3] K- means is a method in cluster analysis to partition observations into k pre-determined number of disjoint clusters. The algorithm works as follows:

Step 1: Choose random K (Number of clusters) according to internal and external indices.

Step 2: Initialize K centroid points and calculate distance of each point from centroids. Assign points to clusters based on their distance from their centroids. The distance considered here is Euclidean distance. Update the centroids in next step.

Relevance in our problem:

Clustering is used here to perform soft biometric classification. Grouping data based on some trait can speed up the recognition process as it will reduce the search space thus decreasing number of comparisons during recognition.

Clustering with/without PCA:

There are two ways to perform K-Means clustering on testing data:

a) Represent testing data with the new set of features obtained after doing PCA.

b) Represent testing data using original image features.

There are many scenarios when it is appropriate to do clustering after PCA like:

When we have so many attributes in our input samples, the problem of scaling may occur and K-Means clustering is sensitive to scaling. In such scenarios, it is common practice to apply PCA as it eliminates redundant features.

Also, clustering performance can get affected if there is noise in input. PCA also helps in overcoming this input noise.

However, PCA can be sensitive on non-Gaussian distributed data, which involve skewed observations such as outlying values. In such cases, we have to consider other techniques like Tukey's biweight correlation based on M-estimate approach [5].

In current scenario, I will implement both ways of clustering to determine which is better in terms of accuracy and efficiency. This will be carried out by analyzing classification results at different values of K (number of principal components) and determining if any relationship exists between principal components and clustering performance.

K-Means Clustering performance also depends on the seed chosen during initialization of centroids. So, for evaluating confusion matrix for each K value, we will run K- Means clustering multiple times for each K value and take the most occurring classification values.

For comparing my cluster validation, I will use both internal criteria as well as external criteria indices.

Internal Criteria:

Internal criteria indices are used when we have to assess quality of an unsupervised learning problem. These indices are highly data dependent. I will use Silhouette index to evaluate cluster validity and to determine the optimal number of clusters for different values of K.

I experimented for different values of K from 1 to 100. The results obtained are shown in following plot:

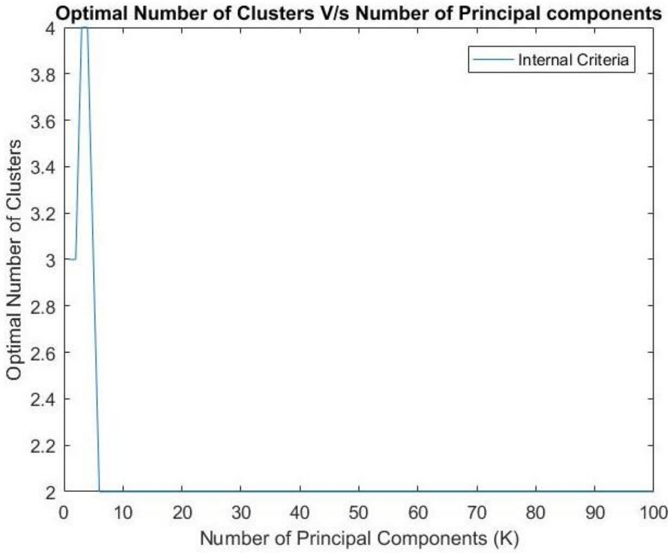


figure 6

Observations: It can be observed from *figure 5* that for few eigen vectors results are not accurate. Although the optimal clusters are 2 in the data set, I am getting values as 3 ,4 for K <=5. The reason behind this is that with fewer components we are not utilizing all parameters for classification and it's affecting clustering performance.

For values of K>5 optimal number of clusters returned by Silhouette criteria is 2 which is correct. This shows that much of the information or variance retained in first 5 principal components is sufficient for classifying data into two groups.

External criteria:

External criteria indices evaluate results based on predefined structure. In our case since we know which image belong to which class. We can use external criteria indices for evaluating the accuracy of clustering. I will use F-Measure as my accuracy measure. Its value ranges from [0 1] and higher the F-Measure value better is the clustering quality.

I determined F-Measure value for different values of K ranging from 1 to 100 and obtained the following plot:

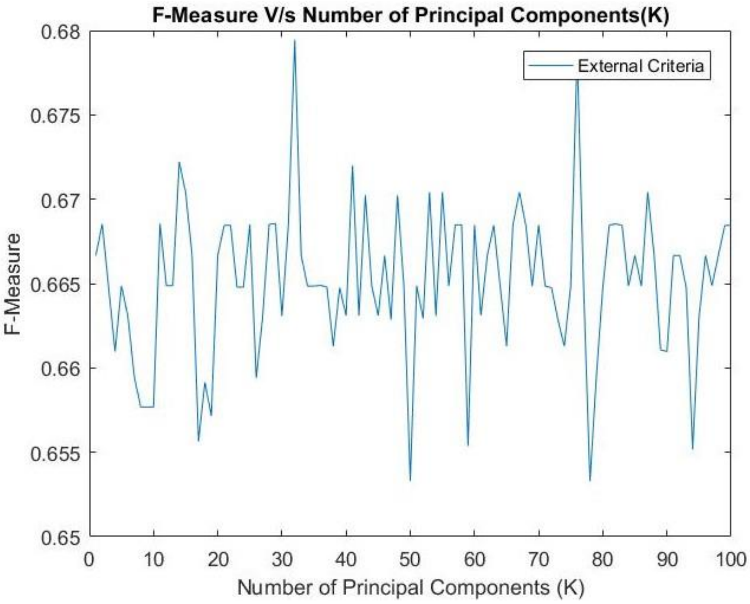
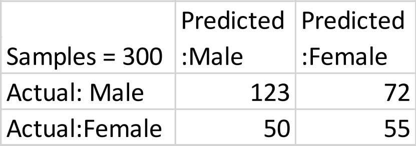


Figure 7

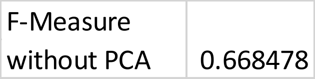
Observations: It can be observed that there is no trend in F-Measure values with respect to number of principal components. Whereas in recognition we saw a trend that recognition performance was increasing with increase in principal components. The value is Fluctuating in range of 0.65 to 0.68. If K is increased further, we can't predict how will the clustering accuracy change as it depends on certain set of principal components not just the first K principal components. There could be any set of principal components from 2500 eigen vectors which can contribute to best clustering performance. Determining these set of eigen vectors is computationally expensive as optimal clustering is a NP-Hard problem.

To check whether there is any improvement in clustering performance after doing PCA, I evaluated confusion matrix and F-Measure score without PCA and obtained the following results:

Confusion Matrix & F-Measure without PCA:



Samples = 300	Predicted :Male	Predicted :Female
Actual: Male	123	72
Actual:Female	50	55



F-Measure without PCA	0.668478

It can be observed that F- Measure value is in the same range without doing PCA**. So, the clustering accuracy with and without PCA is almost same but what has changed is the efficiency of algorithm**. After doing PCA, we are able to classify data using fewer dimensions which is good when we are concerned about computation time.

It is of point of interest to explore, how with fewer principal components, we can classify data with the same accuracy. This is not true in all cases and cannot be generalized, but when we delve deeper into K-Means and PCA , we will see that both methods are exactly minimizing the same objective function.

<u>Relationship between K-Means clustering and PCA</u>:

Although K-means clustering and PCA appear to have different goals and at first sight doesn’t seem to be related, yet there is a connection between them. Both methods try to minimize the same objective function. PCA tries to represent input vectors as a linear combinations of eigen vectors such that the mean-squared reconstruction error could be minimized. K- means on the other hand tries to represent input vectors as a linear combination of small number of cluster centroid vectors where all weights are zero except one. This is also done to minimize the Mean squared reconstruction error. **So, K-means can be seen as a thinly dispersed PCA.**

V. CONCLUSION

In this report we presents a face recognition approach using PCA. Based on experimental study, I can conclude that whenever we are dealing with facial recognition and our images are of same size and taken under same lighting conditions with same head orientation then PCA can turn out be an efficient recognition algorithm. I also empirically justified that PCA guided clustering approach does not necessarily improve the quality of clusters but can be used to improve the efficiency of K-Means clustering.

REFERENCES

[1] Ding, C. and Xiaofeng, H. 2004. K-means Clustering via Principal Component Analysis. In proceedings of the 21st International Conference on Machine Learning, Canada.

[2]F. Ã–zen, “A Face Recognition System Based on Eigen-faces Method,” *Procedia Technology*, Vol. 1, 2011, pp. 118-123.

[3] S. M. Shaharudin, N. Ahmad, Improved Cluster Partition in Principal Component Analysis Guided Clustering. International Journal of Computer Applications (0975 – 8887) Volume 75– No.11, August 2013

[4] Everitt, B. S. and Dunn, G. 2001. Applied Multivariate Data Analysis. London: Arnold Publisher
of Engineering Science and Technology Vol. 2(9), 2010, 4373-4378

[5]T. F. Karim, M. S. H. Lipu, M. L. Rahman and F. Sultana, “Face Recognition Using PCA-based Method,” *in Ad-vanced Management Science (ICAMS), 2010 IEEE Inter-national Conference on*, 2010, pp. 158-162.

[6]J. Meng and Y. Yang, “Symmetrical Two-Dimensional PCA with Image Measures in Face Recognition,” *Int J Adv Robotic Sy*, Vol. 9, 2012.

[7]Marghny, M.H., Abd El-Aziz, R.M., Taloba, A.I. 2011. An Effective Evolutionary Clustering Algorithm: Hepatitis C Case Study. International Journal of Computer Applications. Vol 34-No.6.