Comparative Analysis of Dimensionality Reduction Techniques for Image Recognition (in R)*

Submitted by:

Arkajyoti Bhattacharjee †

Supervised by:

Dr. Sebastian Kurtek †

THE OHIO STATE UNIVERSITY

COLLEGE OF ARTS AND SCIENCES

Submitted on:

18^{th} March, 2024

*This report has been prepared towards the partial fulfillment of the requirements of the course *STAT7730: Advanced Computation Statistics.*

†Department of Statistics, The Ohio State University.

Contents

1 Introduction

Identifying known objects or shapes in images is a vital and intricate field with multi-domain applications including medical diagnostics (Huang (2019)), bio-metrics (Ross et al. (2008)), surveillance (Jain et al. (2004)), and navigation. One such application is facial recognition - identifying individuals from their facial images.

Facial recognition technology has gained prominence due to its potential for enhancing security measures, streamlining authentication processes, and enabling personalized services. It involves the automated detection, analysis, and interpretation of facial features from images or video frames to identify individuals uniquely (Zhao et al. (2003), Turk and Pentland (1991)).

There is a vast literature on facial recognition, that has been backed by research in the fields of computer vision, pattern recognition, and machine learning (Gonzalez (2009), Duda et al. (2012), Forsyth and Ponce (2002)). Numerous studies have explored various aspects of facial recognition, including feature extraction, classification algorithms, and performance evaluation metrics.

The focus of this project is recognizing individuals from facial images, comparing three distinct approaches: Principal Component Analysis (PCA), Fisher's Discriminant Analysis (FDA), and Simple Projection (SP). Our objective is to investigate the effectiveness of these methods in accurately identifying known individuals within images.

PCA, FDA, and SP represent fundamental techniques in dimensionality reduction and feature extraction, that are essential steps in facial recognition systems. PCA (Jolliffe (2003), Shlens (2014)) aims to capture the most significant variations in facial images by projecting them onto a lower-dimensional subspace defined by the principal components; FDA (Fisher (1936), Duda and Hart (1973), Martinez and Kak (2001)) maximizes the discriminative power of features by considering class separability when projecting the data; SP involves projecting the images onto a lower-dimensional space without considering class information.

Through our investigation, we aim to evaluate the performance of these methods using recognition performance, denoted by $F(k)$, for different values of k, the dimension of the projected lower-dimensional subspace, and make suitable comparisons among the three methods.

In Section 2, we discuss the three dimensionality-reduction approaches for our facial recognition problem. We provide theoretical justifications for the methods. The problem setup is also discussed. In Section 3, we see some example faces and summarize our findings. In Section 4, we discuss the effectiveness of the three projections.

2 Methodology

Due to the high dimensionality of images, direct analysis poses challenges. Common strategies involve reducing their dimensionality using methods like Principal Component Analysis (PCA), Fisher's Discriminant Analysis (FDA), and other similar methods, like Simple Projection. All the aforementioned methods are linear transformations. PCA identifies principal components that capture maximum data variance, while LDA focuses on finding directions that enhance class separation, aiding in tasks like image recognition and pattern classification. Simple projection involves projecting the data onto a lower-dimensional subspace without considering class separation or variance structure, offering a straightforward approach to dimensionality reduction. We now discuss each method in some detail.

2.1 PCA

PCA is essentially a change-of-basis technique that is majorly used when we have unlabelled data. Consider we have a high-dimensional n-vector $X \in \mathbb{R}^n$, that is, n is very large. We would like to reduce the dimensionality be projecting X into a lower-dimensional subspace, of dimension, say $d << n$. In essence, we would like to find an orthonormal matrix $U' \in \mathbb{R}^{d \times n}$ such that $Z = U'X \in \mathbb{R}^d$. The existence of such a U is guaranteed by Singular Value Decompostion (SVD). Such $Z = U'X$ are called the principal components (PCs) of X. The PCA algorithm can be summarized as follows:

Algorithm 1: PCA algorithm

---

1. Compute the sample covariance matrix $C(n \times n)$ of X.
2. Compute the SVD of $C = USU'$ to obtain the orthogonal matrix U.
3. Set U_1 to be the first d columns of U $(d << n)$.
4. Define $Z = XU_1$.

Note that X is assumed to have mean 0; otherwise, it can be centered.

2.2 FDA

We tend to use FDA when we have labeled data at hand. The data labels are taken into account when determining the optimal lower dimensional projection.

Suppose there are m labels and hence, m classes, $C_j, j = 1, \ldots, m$. Further, let $|C_j| = n_j, j = 1, \ldots, m.$. Let $X_i^j, i = 1, \ldots, n_j, j = 1 \ldots, m; \sum_{j=1}^m n_j = n$. Then, we have

$$C_j : X_1^j, \ldots, X_{n_j}^j \to \bar{x}_j \in \mathbb{R}^n, j = 1, \ldots, m.$$

We define

$$\bar{x} = \frac{1}{m} \sum_{j=1}^m \bar{x}_j \in \mathbb{R}^n.$$

Now we define two scatter matrices, S_B and S_W, that capture the between-class and within-class separation. Mathematically,

$$S_B = \sum_{j=1}^m (\bar{x}_j - \bar{x})(\bar{x}_j - \bar{x})' \in \mathbb{R}^n,$$

$$S_W = \sum_{j=1}^m \sum_{i=1}^{n_j} (X_i^j - \bar{x}_j)(X_i^j - \bar{x}_j)' \in \mathbb{R}^n.$$

Now, our goal is to find a projection, denoted by the basis U, such that the observations from the same class cluster closer together and the classes separate away maximally, i.e.,

$$Z_i^j = U' X_i^j \in \mathbb{R}^d, U \in \mathbb{R}^{n \times d}.$$

Under this transformation, it can be shown that

$$S_B^Z = U' S_B U, S_W^Z = U' S_W U.$$

The desired U is that $\hat{U}$ such that

$$\hat{U} = \max_{U \in \mathbb{R}^{n \times d}, U'U = I_d} \frac{det(S_B^Z)}{det(S_W^Z)},$$

which is the same as solving the generalized eigenvalue problem,

$$S_B\hat{U} = \lambda S_W\hat{U}.$$

Finally, we select eigenvectors corresponding to the d largest eigenvalues to get $U \in \mathbb{R}^{n\times d}$.

2.3 SP

Kurtek (2024) We use Simple Projection for the sake of comparison among the other dimensionality reduction methods. Here, the projection matrix U is simply the first d columns of I_n, the Identity matrix.

2.4 Problem Setup[1]

There are two sets of images available for such experiments: (1) training set: these images are already recognized and labeled by some expert, and (2) test images: which are new images that are to be recognized and labeled. The goal is to use the similarities between the test and the training images to label the test images. We will assume n_2 training images each for n_1 people in our database. The size of each image is $s_1 \times s_2$. These images are taken at different orientations, different facial expressions, etc. Figure 1 shows

Figure 1: An example of the original picture and some downsampled pictures provided for this project. These images are taken from the ORL face database.

some examples of the images. The left panel shows an image of the original size, while the remaining panels show images of the size provided with the dataset. There are two parts to the procedure. One is to analyze the training images and compute a projection to their principal k-dimensional subspace. The second part is to recognize test images by projecting them into this k-dimensional subspace and comparing them with the training data. We start by outlining the first part. Let the training images be arranged as vectors in a matrix

[1]This section has been modified from Dr. Kurtek's midterm project guide,Kurtek (2024).

of size $(s_1 s_2)(n_1 \times n_2)$. Let's call it Y_{train}. The first n_2 columns are images of person 1, next n_2 columns are images of person 2, and so on, with a total of $n_1 \times n_2$ columns.

(a) Feature Extraction

PCA: We perform PCA on Y_{train}, by computing SVD of Y_{train} directly according to $Y_{train} = U\Sigma V'$. Let U_1 be the first k columns of the orthogonal matrix U. The size of U_1 is $(s_1 \times s_2) \times k$.

FDA: We use PCA to reduce the size of Y_{train} from $(s_1 \times s_2) \times (n_1 \times n_2)$ to $d \times (n_1 \times n_2)$, where $d = \frac{n_1 \times n_2}{2}$. We call this new matrix Y_{train} and call the $(s_1 \times s_2) \times d$ projection matrix U_0. We use all the vectors from the same person as observations from the same cluster. In Y_{train}, the first n_2 columns belong to the first person, the second n_2 to the second person, and so on. We then form the between-class and within-class scatter matrices, and use the generalized eigendecomposition ('geigen' function in the geigen package in R, Hasselman and Lapack authors (2019)) to find k eigenvectors that correspond to the largest eigenvalues; we call that submatrix V . We then use `V = orthonormalization(V, basis = TRUE, norm = TRUE)` (in the far package in R, Serge (2022)) to make the columns orthogonal; it is then a $d \times k$ matrix. We define an $(s_1 \times s_2) \times k$ orthogonal matrix $U_1 = U_0 V$.

Simple Projection: Here, U_1 is simply the first k columns of the $(s_1 \times s_2)$ identity matrix. Each image in Y_{train} can now be reduced to a k-dimensional vector using the projection $Y_1 = U_1' Y_{train}$. Y_1 is of the size $k \times (n_1 \times n_2)$.

(b) Classification: In the second part, we are ready to perform classification (recognition). We are given n_2 test images per person, and the test set Y_{test} is in the same form as the training set. We perform the classification of each image in the test set as follows (the procedure is given for a single test image I).

1. Form feature vector: We compute the projection of I using $I1 = U_1' I$. I_1 is a k-vector.
2. Compute metric: We compute the distance between I_1 and each column of Y_1 using the $2-norm$.
3. Find Nearest Neighbors: We find the label of the column that has the smallest distance to I_1. If this label matches the true label, then the recognition is successful; otherwise, it is a failure.

Finally, we compute the percentage of successful recognition for the three projections: PCA, FDA and simple projection. We call this number $F(k)$ for a given dimension k and compare across the three methods for different values of $k = 1, \ldots, 40$.

Here, the image size is $s_1 = 28$, $s_2 = 23$, the number of people is $n_1 = 40$, the number of training images per person is $n_2 = 5$. So $Y_{train}(Y_{test})$ is a matrix of size 644×200.

3 Results

Here, we summarize our findings. For demonstration purposes, we randomly selected Person 7 from the test data set. The closest training images corresponding to the chosen test image based on PCA, FDA, and SP, for $k = 10, 25, 40$ is shown in Figure 2. Clearly, we see that PCA and FDA succeed in correctly identifying the person for all three ks. However, SP fails to identify the correct person for $k = 10$ and $k = 25$. In fact, for $k = 10$, SP identifies Person 14 from the training images and for $k = 25$, SP identifies Person 18 from the training images. However, it correctly identifies for $k = 40$. This indicates an improvement in performance as k increases.

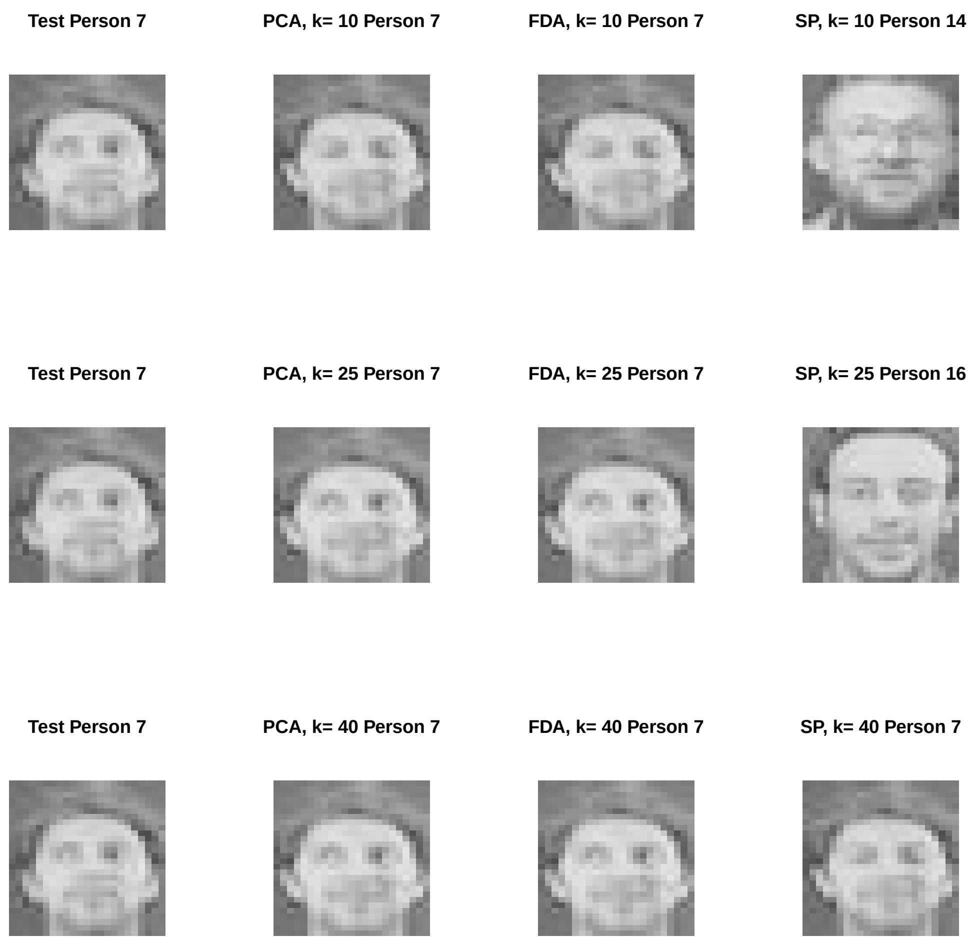

Figure 2: Few examples of test images and the corresponding closest image in the training set.

Based on our R computation, the variation of $F(k)$ versus k is summarized in Table 1. Note that to obtain $F(k)$, we randomly selected a test image and tested if it correctly identified the person from the train images and performed this repeatedly for 1000 times. This should match with the results if instead, we found $F(k)$ for all test images for a fixed k. This follows from the law of large numbers. Clearly, we see an approximately monotonic increase in performance as k increases, which is intuitive and expected. This is further evident in Figure 3. We see that FDA and PCA perform much better than SP for almost all k. F_{PCA} and F_{FDA} follow each other very closely. However, for large k, we see PCA to perform better than FDA.

k	$F_{PCA}(k)$	$F_{FDA}(k)$	$F_{SP}(k)$	k	$F_{PCA}(k)$	$F_{FDA}(k)$	$F_{SP}(k)$
1	7.6	13.1	11.8	2	27.3	30.5	19.2
3	39	48	13.8	4	47.3	56.7	22.3
5	57.8	67.5	18.5	6	71.6	70.8	25.6
7	72.3	72.8	29.8	8	73.9	77.2	28.8
9	77.8	81.2	32.4	10	80.4	79.2	32.1
11	84.8	84.6	34.4	12	83.9	82.6	36.5
13	87.3	82.5	36.3	14	87.1	85.7	40.2
15	87.5	87.1	44.4	16	89.3	88.8	37.5
17	90.4	88.8	34.9	18	90.8	89.5	34.4
19	91	91.2	37.8	20	88.7	88.8	33.1
21	91.8	91.2	39.8	22	93.5	90	41.8
23	92.8	89.6	39.8	24	92.5	89.4	40.1
25	91.6	88.9	38.6	26	91.8	88	38.8
27	93.1	88	38.8	28	93.3	87.9	44.3
29	93.6	85.1	44.9	30	94.6	88.1	45.9
31	95.1	86	49.3	32	93.4	89.4	48.8
33	93.9	89.7	48.1	34	94.4	89.7	50
35	95.5	88.8	49.9	36	95.3	88.4	50.5
37	94.7	89.2	50.9	38	94.7	89.2	53.7
39	95.9	92.7	53.5	40	96.6	91.4	52.2

Table 1: Recognition performance, F(k), for the three methods for different values of k.

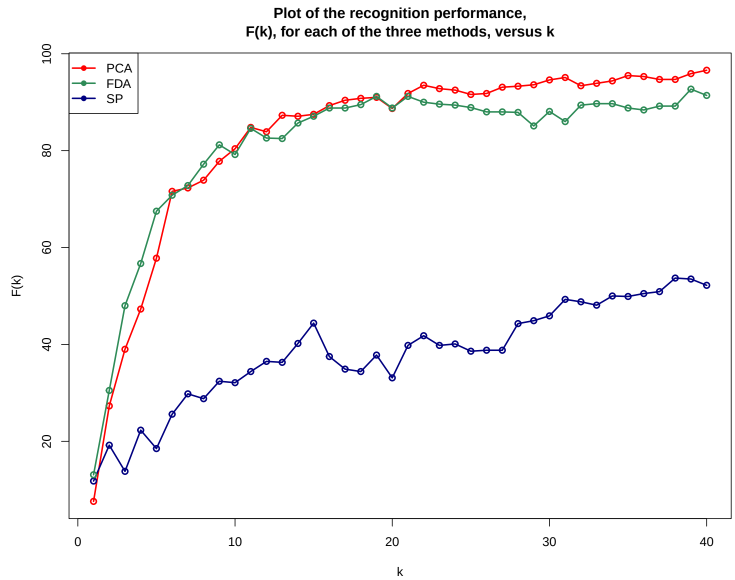

Figure 3: Recognition performance (F(k)) versus the number k for the three dimension reduction techniques.

4 Discussion

In this project, we did a comparative analysis of three dimensionality-reduction techniques for the problem of facial recognition. We found that PCA performed better. Intuitively, FDA should have performed better since it used the information of class labels. However, the performance of FDA probably is not as good as dimensionality becomes very large.It could also be that the image classes are not as well-separated as they should for FDA to perform better than PCA. In addition, simple projection, although relatively easier to compute, shows relatively less recognition performance than both PCA and FDA.

5 R Codes

The R code for the plots and analysis done here are available at github.com/ArkaB-DS/STAT7730eigenfaces.

6 Acknowledgements

I would like to thank Dr. Kurtek for the opportunity of this project, his lecture notes, slides, and project guide (Kurtek (2024)), and for providing guidance whenever needed. Finally, I greatly appreciate the authors of Mostafa and Hossain (2020) as their paper provided a great source of reference and benchmarking.

References

Duda, R. O. and Hart, P. E. (1973). *Pattern classification and scene analysis*. Wiley.

Duda, R. O., Hart, P. E., and Stork, D. G. (2012). Pattern classification. 605 third avenue.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188.

Forsyth, D. A. and Ponce, J. (2002). *Computer vision: a modern approach*. prentice hall professional technical reference.

Gonzalez, R. C. (2009). *Digital image processing*. Pearson education india.

Hasselman, B. and Lapack authors (2019). *geigen: Calculate Generalized Eigenvalues, the Generalized Schur Decomposition and the Generalized Singular Value Decomposition of a Matrix Pair with Lapack*. R package version 2.3.

Huang, H. (2019). *Pacs-based multimedia imaging informatics: Basic principles and applications*. John Wiley & Sons.

Jain, A. K., Ross, A., and Prabhakar, S. (2004). An introduction to biometric recognition. *IEEE Transactions on circuits and systems for video technology*, 14(1):4–20.

Jolliffe, I. T. (2003). Principal component analysis. *Technometrics*, 45(3):276.

Kurtek, S. (2024). Stat 7730: Advanced computation statistics: Notes, slides, project guide. Handwritten/-Typed/Carmen.

Martinez, A. M. and Kak, A. C. (2001). Pca versus lda. *IEEE transactions on pattern analysis and machine intelligence*, 23(2):228–233.

Mostafa, F. B. and Hossain, S. (2020). Revisiting the performance of pca versus fda versus simple projection for image recognition.

Ross, A., Nandakumar, K., and Jain, A. K. (2008). Introduction to multibiometrics. *Handbook of biometrics*, pages 271–292.

Serge, D. J. G. (2022). *far: Modelization for Functional AutoRegressive Processes*. R package version 0.6-6.

Shlens, J. (2014). A tutorial on principal component analysis. *arXiv preprint arXiv:1404.1100*.

Turk, M. and Pentland, A. (1991). Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1):71–86.

Zhao, W., Chellappa, R., Phillips, P. J., and Rosenfeld, A. (2003). Face recognition: A literature survey. *ACM computing surveys (CSUR)*, 35(4):399–458.