ENSEEIHT

Applied mathematics and computer engineering department

Data analysis and scientific computing

2021-2022

Project report

# Application of PCA : "Eigenfaces" - Power iteration methods

Issam El Kadiri | Noussayma Ouahi | Yasmine Charifi

# Contents



# List of Figures

# Introduction :

In this first part of our data analysis  scientific computing project, we will use PCA, power methods, in or to reduce the dimensions of a given space, composed of images of individuals wearing or not wearing a mask. The work is done using Matlab.

There will be a second and a third part, which will contain the conclusion of this project.

# 1 The "Eigenfaces" :

## 1.1 Eigenspaces :

In this section, we aim to compute the $n-1$ eigenpairs of $\Sigma = X_c^T X_c/n$ , the covariance matrix of our dataset, composed of 4 faces (2 males, 2 females), under 4 positions, where $X_c$ is the centred dataset matrix. Since the dimension of $\Sigma$ is huge ($p \times p = 120000 \times 120000$), we instead computed the same eigenpairs with $\Sigma_2 = X_c X_c^T/n$, whose dimension is $n \times n = 16 \times 16$.

The figure below shows the $n-1$ axis of the eigenspace of our dataset and the mean person.

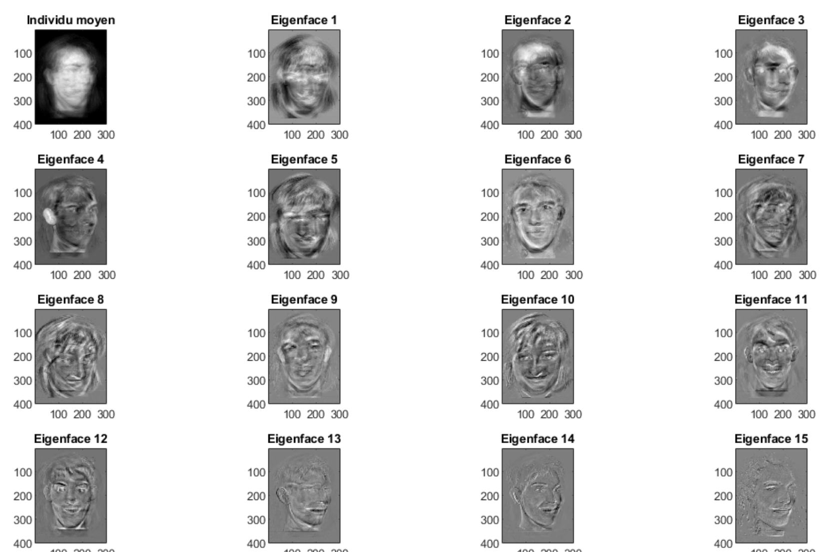


Figure 1: $n-1$ axis of the eigenspace of our dataset and the mean person

## 1.2 Projection of the images on the resulting eigenspace :

In this section, we use the $n-1$ computed eigenfaces, as a projection basis, in order to reconstruct the faces using the first $q$ eingenfaces, where $q \in [0, n-1]$ ($q = 0$ corresponds to the mean person).

The following figures show the projections (without masks) using the first eigenface, all the 15 eigenfaces, and the evolution of the root mean square error (RMSE) : as we use more eigenfaces, RMSE decreases.

Figure 2: Projection using the first eigenface (without mask)

Figure 3: Projection using all the eigenfaces (without mask)

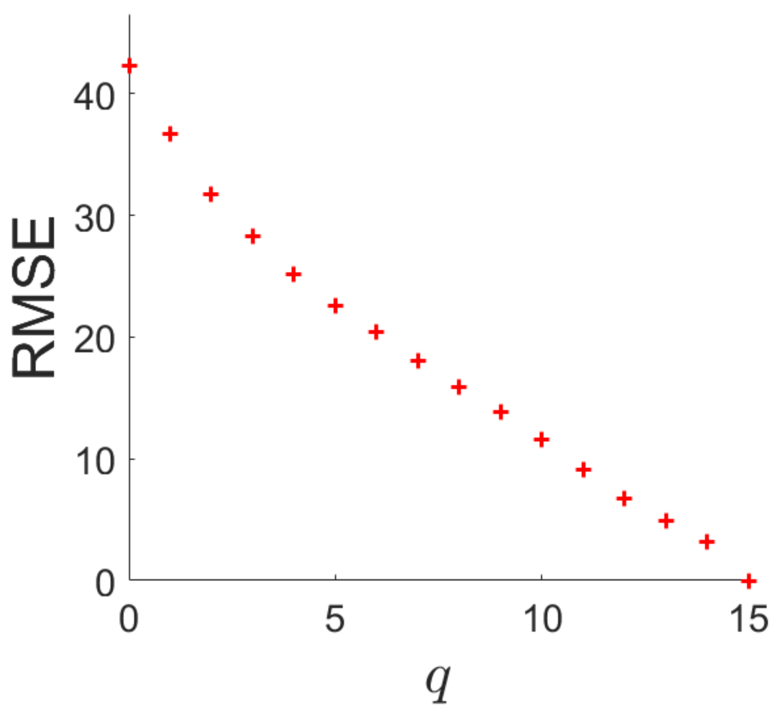


Figure 4: Evolution of RMSE (without mask)

The following figures show the projections (with masks) using the first eigenface, all the 15 eigenfaces, and the evolution of the root mean square error (RMSE) : as we use more eigenfaces, RMSE decreases.

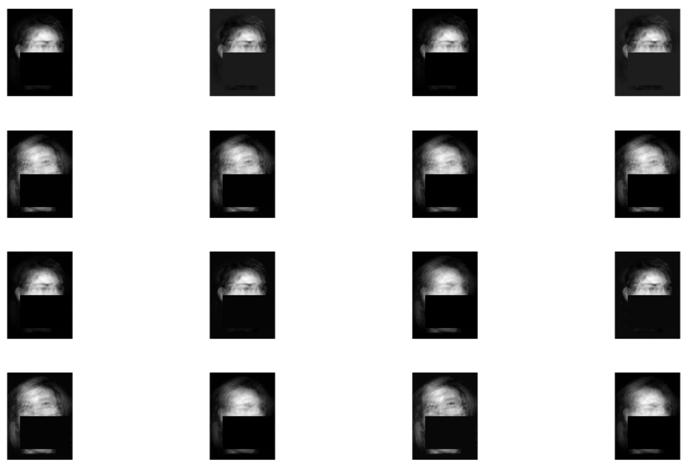

Figure 5: Projection using the first eigenface (with masks)

Figure 6: Projection using all the eigenfaces (with masks)

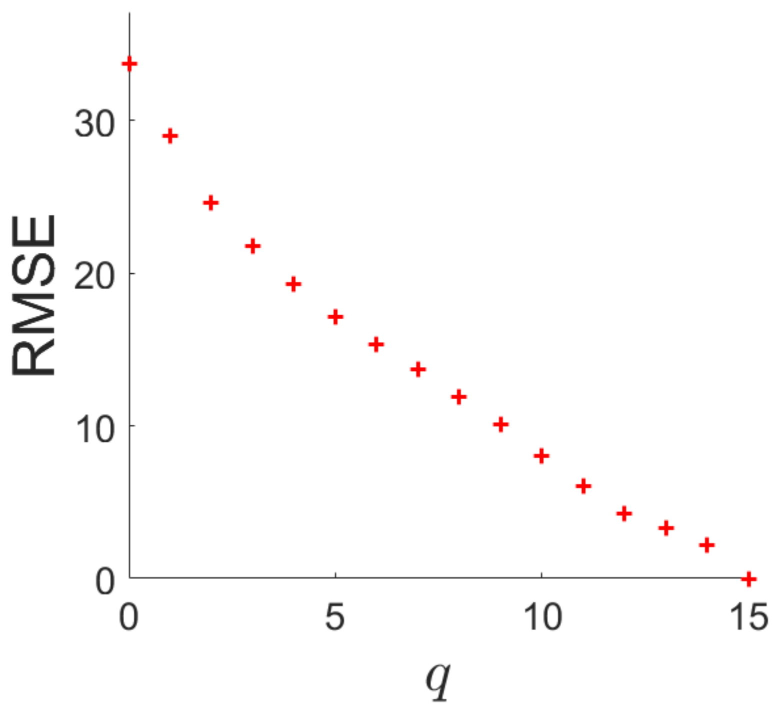


Figure 7: Evolution of RMSE (with masks)

# 2 PCA and the power method :

## 2.1 Could we know the eigenelements of $HH^\top$ by knowing the eigenelemnts of $H^\top H$? (Question 4)

Consider a rectangular matrix $H \in \mathbb{R}^{n\times p}$ , knowing the proper elements; the eigenvalues and eigenvectors of $H^\top H$ allows to know the eigenelements of $HH^\top$because :
let $\lambda$ eigenvalue of $H^\top H$ let u be its associated eigenvector therefore :
$H^\top Hu = \lambda u \implies HH^\top H = \lambda Hu$.
Let v $= Hu$ then $HH^\top v = \lambda v$ so $\lambda$ is an eigenvalue of $HH^\top$ and its associated eigenvector is $v = Hu$ so we can have all the proper elements of the matrix $HH^\top$.

## 2.2 The iterated power

The *puissance_iteree.m* script contains two instances of the iterated power method: one applied to a matrix $A^\top A$, the other applied to the matrix $AA^\top$.

After completing this script,by the following Algorithm 1
Initialisation : cv $\leftarrow$ FALSE, (cv stands for convergence), i $\leftarrow$ 0, $\lambda \leftarrow x^\top Mx$
1. while NOT cv :
   $\mu \leftarrow \lambda$
   $x \leftarrow Mx$
   $x \leftarrow x/||x||$
   $\lambda \leftarrow x^\top Mx$
   $i \leftarrow i+1$
   $cv \leftarrow (\frac{|\lambda-\mu|}{|\mu|} \leq \epsilon).OR.(it_{max})$
2. Return $(\lambda, x)$

## 2.3 Is it more useful in theory to use a function such as eig or the method of the power to compute the eigenelements of if the goal is to perform a PCA to reduce the dimensions of a space? (Question 6)

If the goal is to perform a PCA to reduce the dimension of a space, it is more efficient to use the power method instead of a function like eig for the following reasons :

- The power method return the eigenpairs with the largest eigenvalues first, so we can stop the algorithm if we have enough eigenvalue (and thus enough information, knowing that we can now the sum of all the eigenvalues in our example where we deal with non-negative matrices, by evaluating the trace a the matrix).

- Even if we wanted all the eigenpair, it's shown in our matlab code, in $puissance_iteree.m$, that the eig function takes more time than the power method. Moreover, the return eigenpairs of the power method are already sorted descendingly, unlike the eig function, whose return needs to be sorted in order to perform PCA.

```
Erreur pour la methode avec la grande matrice = 9.871e-09
Erreur pour la methode avec la petite matrice = 9.859e-09
Ecart relatif entre les deux valeurs propres trouvees = 0.00e+00
Temps pour une ite avec la grande matrice = 7.521e-03
Temps pour une ite avec la petite matrice = 2.408e-04
Temps pour calculer la valeur propre maximale en utilisant la fonction eig et sort = 7.812e-01

Valeur propre dominante (methode avec la grande matrice) = 9.228e+04
Valeur propre dominante (methode avec la petite matrice) = 9.228e+04
Valeur propre dominante (fonction eig) = 9.228e+04
```

Figure 8: Results and time consumption of the power methods, and the eig function

## 2.4 Which matrix should we apply the algorithm's method to minimize the calculation time and the memory used? (Question 7)

We can conclude from the results that we had through the script of the power method that to minimize the time, using $\Sigma_2$ is a considerable solution ,and as we have said previously, both $AA^\top$ and $A^\top A$ have the same eigenvalues so we should chose between the two the smaller to minimize the memory used which lead to chose $\Sigma_2$ over $\Sigma$.