MXB201
Linear Algebra

Semester 1, 2023

Group Project

Group 8

Rohan Boas (n11031743)
Sophia Sekulic (n000000)
Horace Fung (n000000)

Document last compiled: 2023-05-23 16:40

Contents

1 Introduction

2 Part I

2.1 Conclusion

3 Part II

In order to create a moustache detector, we need to identify aspects of a face programmatically. We consider the database of N images as a single matrix A, where each column of A is an r-by-c image reshaped into a very tall column vector in $\mathbb{R}^{rc}$. As our goal is to detect features (in our case, moustaches), we want to find aspects of the images different from "average", which, in this case, would be the mean face image.

Figure 1: The mean face generated from the data

With the mean face represented by the column vector $\bar{a}$, we take A_{centered} to denote the mean-centred representation of our data.

$$A_{\text{centered}} = A - \begin{bmatrix} | & & | \\ \bar{a} & \cdots & \bar{a} \\ | & & | \end{bmatrix} \tag{1}$$

3.1 Eigenfaces

"Eigenfaces" are left singular values of our centered face database A_{centered}. As the first left singular values $u_i \in \mathbb{R}^{rc}$ from a reduced singular value decomposition contain the most information, it largely suffices to find the first few significant eigenfaces.

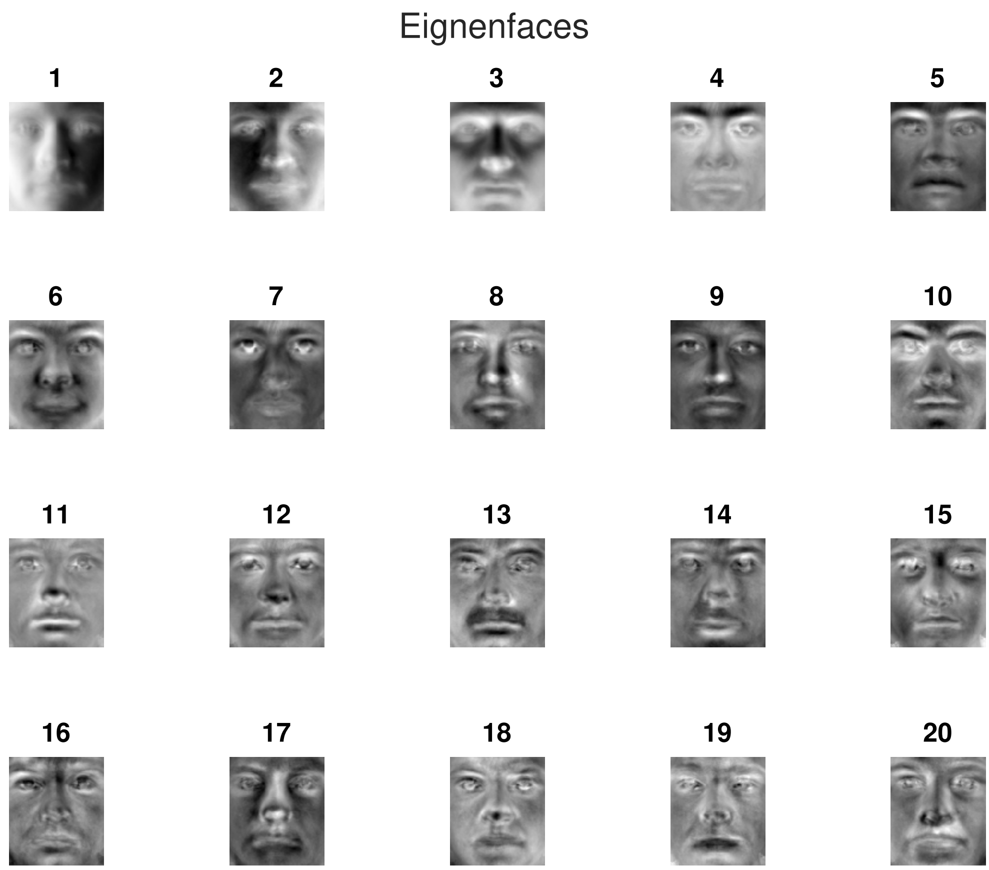

Figure 2: The first 20 eigenfaces of A

We can therefore construct an eigenface-space, whose basis is $W = \{u_i \mid i \leq \nu\}$. By projecting faces onto W, we can easily see the contribution of each eigenface to each face.
To pick a reasonable value of ν, we plot the ratio σ_ν/σ_1 to find the relative contribution of left singular values.

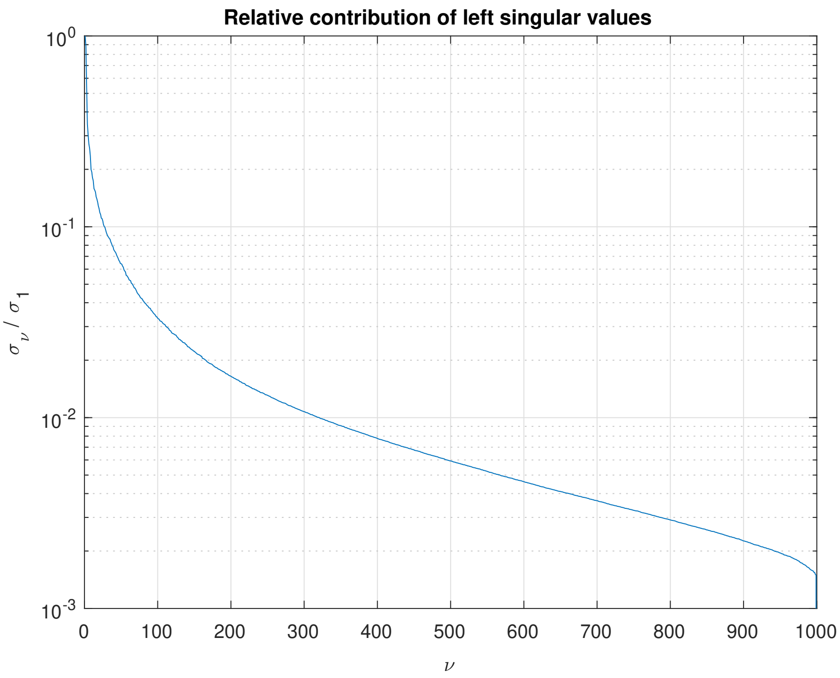

Figure 3: Relative contribution of left singular values

As seen in Figure 3, contribution drops off greatly after 100, so we pick $\nu = 100$. We can construct our approximate matrix for A_{centered} by $\tilde{A}_{\text{approx}} = U(:, 1 : \nu)\Sigma(1 : \nu, 1 : \nu)V(1 : \nu, 1 : \nu)^\top$.

3.2 Moustache Detector

From Figure 2, we can see eigenface 13 is our "moustache face", i.e. it contains the information necessary to identify a moustache. Therefore, by projecting our moustache face and target face onto W we can calculate the distance between them to find how close they are, and therefore the "amount" of moustache, which we will call our metric. We do this by multiplying faces with the pseudoinverse of $\tilde{A}_{\text{approx}}$ and finding our metric between them. Since $\|v\| \in [0, \infty)$ where 0 represents "full moustache", we can rescale it into a more readable score by setting our metric to be $f(\|v\|) \in (0, 1]$ where $f(x) = e^{-x}$, giving 1="full moustache" and 0="no moustache". TODO: fix everything

3.3 Conclusion

Copypasted from practical worksheet: In the string example, the columns of turned out to be sine waves, telling us something fundamental about the physics behind the data. In the eigenface example, the columns of will look just like human faces with particular features emphasised, e.g a moustache. In both cases, the SVD is uncovering the essence of how to represent the data in the most compact way. In doing so, it's highlighting the main features of the data. By projecting the actual data onto this low-dimensional feature space, you are getting a very compact but still highly accurate representation of the data, e.g. in the string example, each string profile is really just composed of a handful of sine waves. But in the damped string example, you should have found that the basis vectors weren't just pure sine waves any more. That's because the SVD

discovered that it could “exploit” the asymmetry of the data we provided, to get a more efficient representation in terms of lopsided basis functions. In some sense, it was too clever for its own good there, because the lopsidedness was really an artefact of the data we happened to provide, rather than anything fundamental to the physics. Had we chosen another initial condition (e.g. pulled the string up at instead), it would have come out lopsided the other way around. But there’s nothing truly lopsided in the physics: it’s really just sine waves no matter what.
Copypasted from task: Back to digital health now: the idea is that these ν-dimensional coordinate vectors could be used to train a machine learning system to identify warning signs of disease. The “faces” are really going to be MRI brain scans, remember, and the features will be whatever the SVD identifies as the most distinguishing characteristics of human MRI brain scans. With a very large dataset of pre-labelled healthy and diseased brain images, the system can try to learn which combinations of features are associated with greater incidence of (say) neurodegenerative disease. New patients whose coordinate vectors have a lot in common with these learned “biomarkers” could then benefit from much earlier intervention than would be possible using only the diagnostic expertise of a human operator, with only a human lifetime’s capacity for learning.