ISYE 6740 Summer 2024
Homework 2
(100 points + 10 bonus points)

Andrea Ruiz Marquez

6/8/2024

1. Conceptual questions [25 points].

1. (5 points) Please prove the first principle component direction v corresponds to the largest eigenvector of the sample covariance matrix:

$$v = \arg \max_{w:\|w\|\leq 1} \frac{1}{m} \sum_{i=1}^{m} (w^T x^i - w^T \mu)^2.$$

You may use the proof steps in the lecture, but please write them logically and cohesively.
First we expand the function, where C represents the covariance matrix:

$$\begin{aligned} &= \arg \max_{w:\|w\|\leq 1} \frac{1}{m} \sum_{i=1}^{m} \left(w^T (x_i - \mu)\right)^2 \\ &= \arg \max_{w:\|w\|\leq 1} w^T \left(\frac{1}{m} \sum_{i=1}^{m} (x_i - \mu)(x_i - \mu)^T \right) w \\ &= \arg \max_{w:\|w\|\leq 1} w^T C w \end{aligned}$$

And from lecture we assume that, using the Lagrange multiplier method:

$$L(w, \lambda) = w^T S w - \lambda (w^T w - 1)$$

where λ is the Lagrange multiplier. To find the maximum, we differentiate $L(w, \lambda)$ with respect to w and λ, and set the derivatives to zero:

$$\nabla_w L(w, \lambda) = 2Sw - 2\lambda w = 0$$

Then solve:

$$Sw = \lambda w$$

which is the eigenvalue problem. The solution to this problem is given by the eigenvector corresponding to the largest eigenvalue of S.

2. (5 points) Based on the outline given in the lecture, show that the maximum likelihood estimate (MLE) for Gaussian random variable using observations $x^1, \ldots, x^m$, that are *i.i.d.* (independent and

identically distributed) following the distribution $\mathcal{N}(\mu, \sigma^2)$, and the mean and variance parameters are given by

$$\hat{\mu} = \frac{1}{m}\sum_{i=1}^{m} x^i, \quad \hat{\sigma}^2 = \frac{1}{m}\sum_{i=1}^{m}(x^i - \hat{\mu})^2,$$

respectively. Please show the work for your derivations in full detail.

Derive $\hat{\mu}$, $\ell(\mu, \sigma^2)$ with respect to μ and set it to zero:

$$\frac{\partial \ell(\mu, \sigma^2)}{\partial \mu} = 0$$

$$\frac{\partial \ell(\mu, \sigma^2)}{\partial \mu} = \sum_{i=1}^{m} \frac{\partial}{\partial \mu} \log f(x^i; \mu, \sigma^2)$$

Substitute PDF of the Gaussian:

$$= \sum_{i=1}^{m} \frac{\partial}{\partial \mu}\left[-\frac{1}{2}\log(2\pi\sigma^2) - \frac{(x^i - \mu)^2}{2\sigma^2}\right]$$

$$= \sum_{i=1}^{m} \frac{2(x^i - \mu)}{2\sigma^2}$$

And setting to zero:

$$\sum_{i=1}^{m} \frac{x^i - \hat{\mu}}{\sigma^2} = 0$$

$$\Rightarrow \hat{\mu} = \frac{1}{m}\sum_{i=1}^{m} x^i$$

Derive $\hat{\sigma}^2$:

$$\frac{\partial \ell(\mu, \sigma^2)}{\partial \sigma^2} = \sum_{i=1}^{m} \frac{\partial}{\partial \sigma^2} \log f(x^i; \mu, \sigma^2)$$

Substitute PDF of the Gaussian:

$$= \sum_{i=1}^{m} \frac{\partial}{\partial \sigma^2}\left[-\frac{1}{2}\log(2\pi\sigma^2) - \frac{(x^i - \mu)^2}{2\sigma^2}\right]$$

$$= \sum_{i=1}^{m} \left[-\frac{1}{2\sigma^2} + \frac{(x^i - \mu)^2}{2\sigma^4}\right]$$

Now, setting this derivative to zero:

$$\sum_{i=1}^{m} \left[-\frac{1}{2\hat{\sigma}^2} + \frac{(x^i - \mu)^2}{2(\hat{\sigma}^2)^2}\right] = 0$$

$$\Rightarrow \hat{\sigma}^2 = \frac{1}{m}\sum_{i=1}^{m}(x^i - \mu)^2$$

Based on results substitute previous result into $\hat{\sigma}^2$:

$$\hat{\sigma}^2 = \frac{1}{m}\sum_{i=1}^{m}(x^i - \hat{\mu})^2$$

3. (5 points) Explain the three key ideas in ISOMAP (for manifold learning and non-linear dimensionality reduction).
   First key idea for ISOMAP is that it preserves geodesic distances which captures the structure of the data manifold, despite its non-linearity.
   The second, is it approximates distances by creating a graph of nearest neighbors and computing shortest path distances.
   Lastly, it creates multidimensional scaling to the data into lower-dimension space while trying to preserve euclidean distances, which enables visualization and patterns of the manifold's structure.

4. (5 points) Explain how to decide k, the number of principle components, from data.
   One way to decide k value is to perform cross validation, by splitting data into training and validation datasets then performing PCA with different values of k evalutaing performance on the validation dataset.

5. (5 points) How do outliers affect the performance of PCA? You can create numerical examples to study and show this.
   Outliers impact the performance of PCA by skewing the the statistical structure of the data. It can affect the variance which can affect principal components.

2. PCA: Food consumption in European countries [20 points].

The data food-consumption.csv contains 16 countries in Europe and their consumption for 20 food items, such as tea, jam, coffee, yogurt, and others. We will perform principal component analysis to explore the data. In this question, please implement PCA by writing your own code (you can use any basic packages, such as numerical linear algebra, reading data, in your file).

First, we will perform PCA analysis on the data by treating each country's food consumption as their "feature" vectors. In other words, we will find weight vectors to combine 20 food-item consumptions for each country.

(a) (10 points) For this problem of performing PCA on countries by treating each country's food consumption as their "feature" vectors, explain how the data matrix is set-up in this case (e.g., the columns and the rows of the matrix correspond to what). Now extract the first two principal components for each data point (thus, this means we will represent each data point using a two-dimensional vector). Draw a scatter plot of two-dimensional representations of the countries using their two principal components. Mark the countries on the plot (you can do this by hand if you want). Please explain any pattern you observe in the scatter plot.

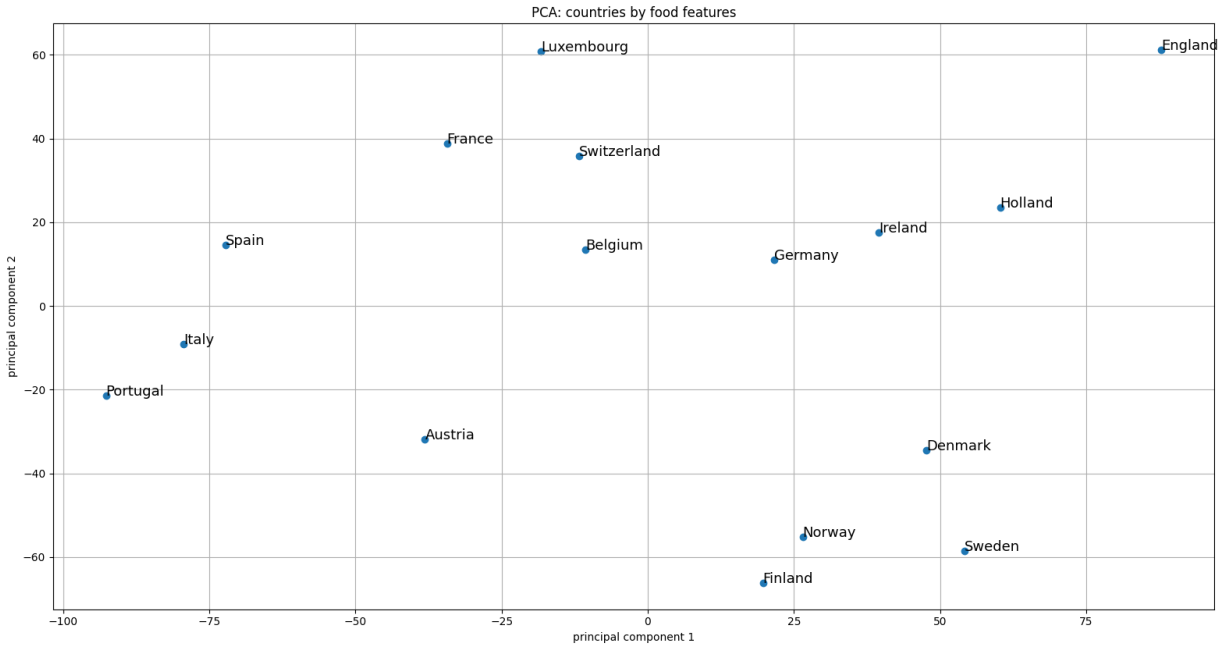

Figure 1: PCA output

From the output plot, the data shows how countries resemble eachother in terms of food consumption. For example, England and Portugal show the most unique features in food consumption in comparison to the other countries, as the they are more correlated with their neighboring points.

(b) (10 points) Now, we will perform PCA analysis on the data by treating country consumptions as "feature" vectors for each food item. In other words, we will now find weight vectors to combine

country consumptions for each food item to perform PCA another way. Project data to obtain their two principle components (thus, again each data point – for each food item – can be represented using a two-dimensional vector). Draw a scatter plot of food items. Mark the food items on the plot (you can do this by hand if you want). Please explain any pattern you observe in the scatter plot.

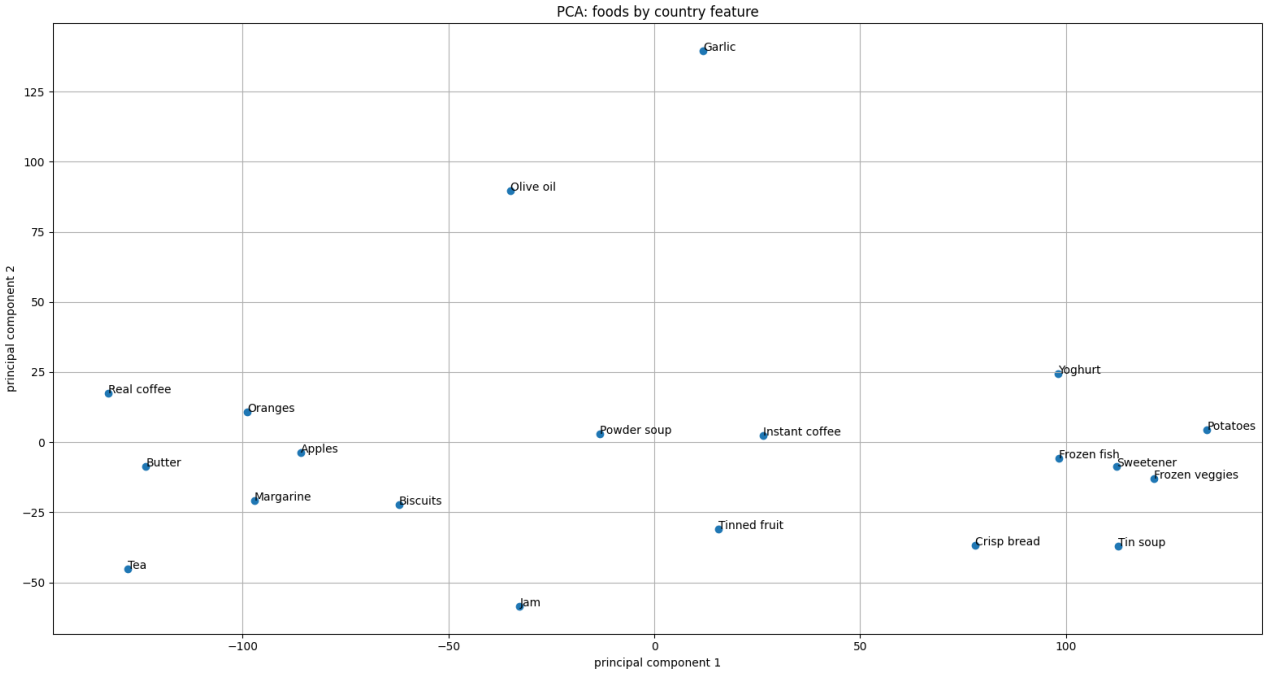

Figure 2: PCA output, B

From the output plot, the data shows how certain foods are distributed with given countries. Olive oil and garlic opposes most of the other food items, where the other food items have similar locations within each other which means these are more evenly distributed in the list of countries given.

3. Order of faces using ISOMAP [25 points]

This question aims to reproduce the ISOMAP algorithm results in the original paper for ISOMAP, J.B. Tenenbaum, V. de Silva, and J.C. Langford, Science 290 (2000) 2319-2323 that we have also seen in the lecture as an exercise (isn't this exciting to go through the process of generating results for a high-impact research paper!)

The file isomap.mat (or isomap.dat) contains 698 images, corresponding to different poses of the same face. Each image is given as a 64×64 luminosity map, hence represented as a vector in $\mathbb{R}^{4096}$. This vector is stored as a row in the file. (This is one of the datasets used in the original paper.) In this question, you are expected to implement the ISOMAP algorithm by coding it up yourself. You may find the shortest path (required by one step of the algorithm), using `https://scikit-learn.org/stable/modules/generated/sklearn.utils.graph_shortest_path.graph_shortest_path.html`.

Using Euclidean distance (i.e., in this case, a distance in $\mathbb{R}^{4096}$) to construct the ϵ-ISOMAP (follow the instructions in the slides.) You will tune the ϵ parameter to achieve the most reasonable performance. Please note that this is different from K-ISOMAP, where each node has exactly K nearest neighbors.

(a) (5 points) Visualize the nearest neighbor graph (you can either show the adjacency matrix (e.g., as an image), or visualize the graph similar to the lecture slides using graph visualization packages such as Gephi (https://gephi.org) and illustrate a few images corresponds to nodes at different parts of the graph, e.g., mark them by hand or use software packages).

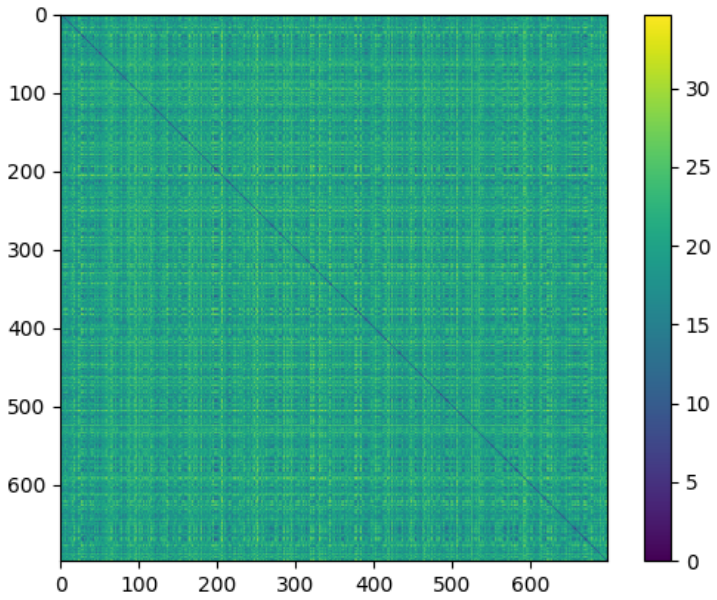

Figure 3: Adjacency matrix

(b) (10 points) Implement the ISOMAP algorithm yourself to obtain a two-dimensional low-dimensional embedding. Plot the embeddings using a scatter plot, similar to the plots in lecture slides. Find a few images in the embedding space and show what these images look like and specify the face locations on the scatter plot. Comment on do you see any visual similarity among them and their arrangement, similar to what you seen in the paper?

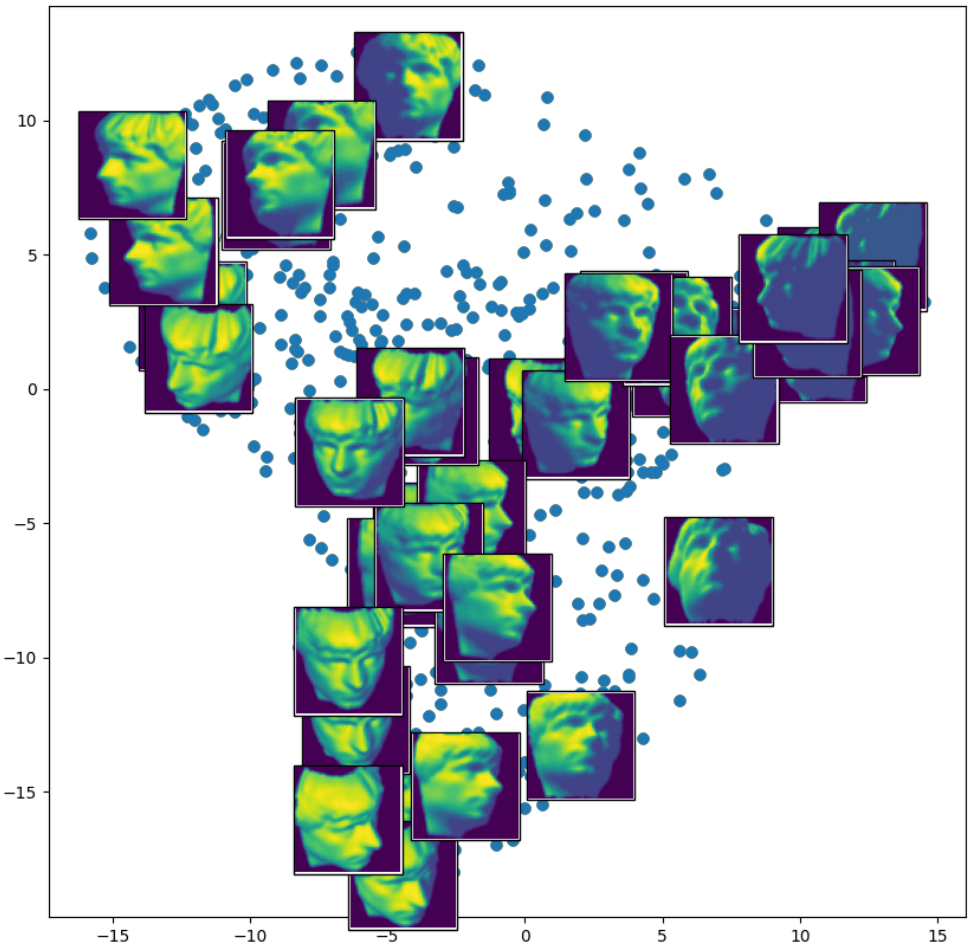

Figure 4: 2d embedded

yes they are similar to the images seen in the professors lecture, in exception that my images are in color and are somewhat larger than they should be however, where the direction of faces that they are pointing towards make sense in terms of the location in the scatter plot.

(c) (10 points) Perform PCA (you can now use your implementation written in Question 1) on the images and project them into the top 2 principal components. Again show them on a scatter plot. Explain whether or you see a more meaningful projection using ISOMAP than PCA.

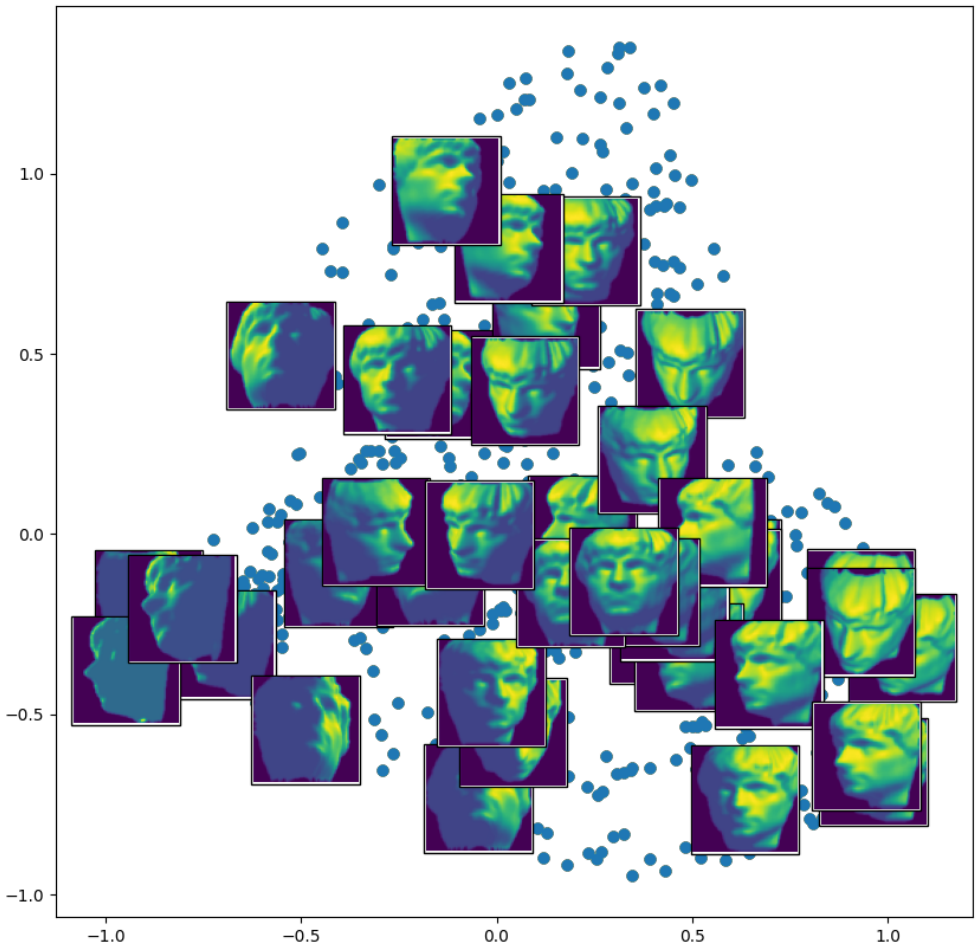

Figure 5: pca

It seems using pca the projected images on the plot are less accurate, as they are all closer together and it is hard to distinguish maybe due to the saturation in the images instead of the actual direction of the faces.

4. Density estimation: Psychological experiments. [30 points]

In *Kanai, R., Feilden, T., Firth, C. and Rees, G., 2011. Political orientations are correlated with brain structure in young adults. Current biology, 21(8), pp.677-680.*, data are collected to study whether or not the two brain regions are likely to be independent of each other and considering different types of political view **For this question; you can use third party histogram and KDE packages; no need to write your own.** The data set n90pol.csv contains information on 90 university students who participated in a psychological experiment designed to look for relationships between the size of different regions of the brain

and political views. The variables amygdala and acc indicate the volume of two particular brain regions known to be involved in emotions and decision-making, the amygdala and the anterior cingulate cortex; more exactly, these are residuals from the predicted volume, after adjusting for height, sex, and similar body-type variables. The variable orientation gives the students' locations on a five-point scale from 1 (very conservative) to 5 (very liberal). Note that in the dataset, we only have observations for orientation from 2 to 5.

Recall in this case, the kernel density estimator (KDE) for a density is given by

$$p(x) = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{h}K\left(\frac{x^i - x}{h}\right),$$

where x^i are two-dimensional vectors, $h > 0$ is the kernel bandwidth, based on the criterion we discussed in lecture. For one-dimensional KDE, use a one-dimensional Gaussian kernel

$$K(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}.$$

For two-dimensional KDE, use a two-dimensional Gaussian kernel: for

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2,$$

where x_1 and x_2 are the two dimensions respectively

$$K(x) = \frac{1}{2\pi}e^{-\frac{(x_1)^2+(x_2)^2}{2}}.$$

(a) (5 points) Form the 1-dimensional histogram and KDE to estimate the distributions of amygdala and acc, respectively. For this question, you can ignore the variable orientation. Decide on a suitable number of bins so you can see the shape of the distribution clearly. Set an appropriate kernel bandwidth $h > 0$.

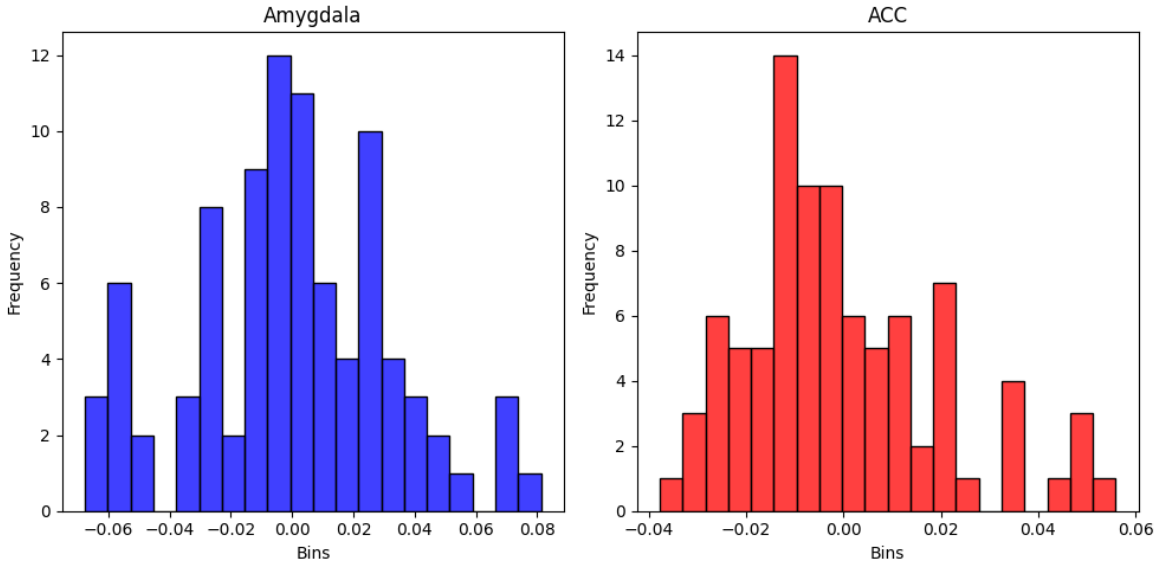

Figure 6: Histograms

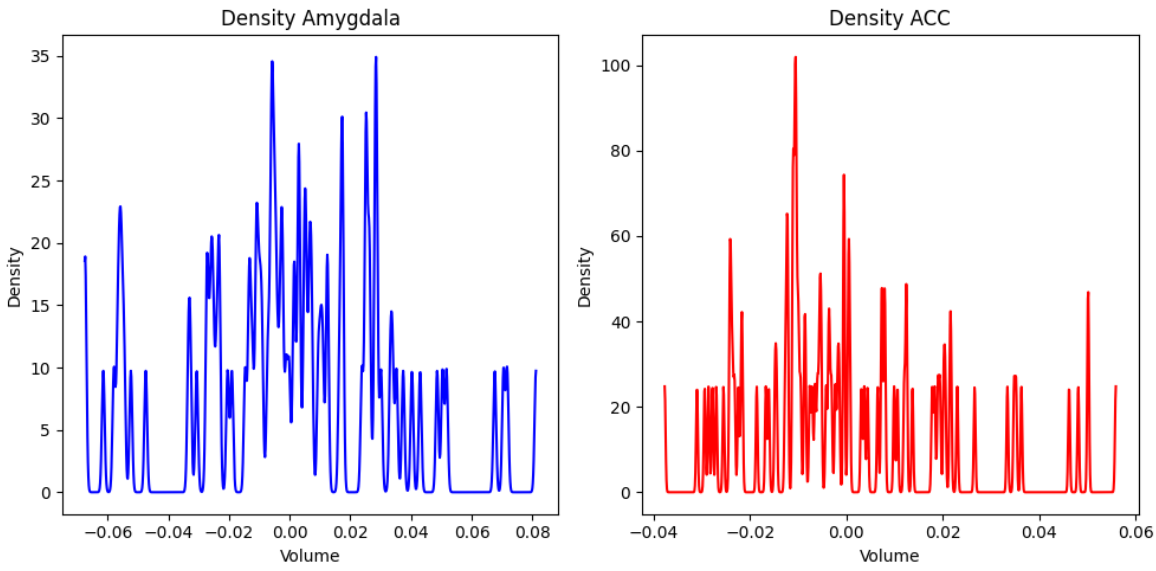

Figure 7: KDE

(b) (5 points) Form 2-dimensional histogram for the pairs of variables (amygdala, acc). Decide on a suitable number of bins so you can see the shape of the distribution clearly.

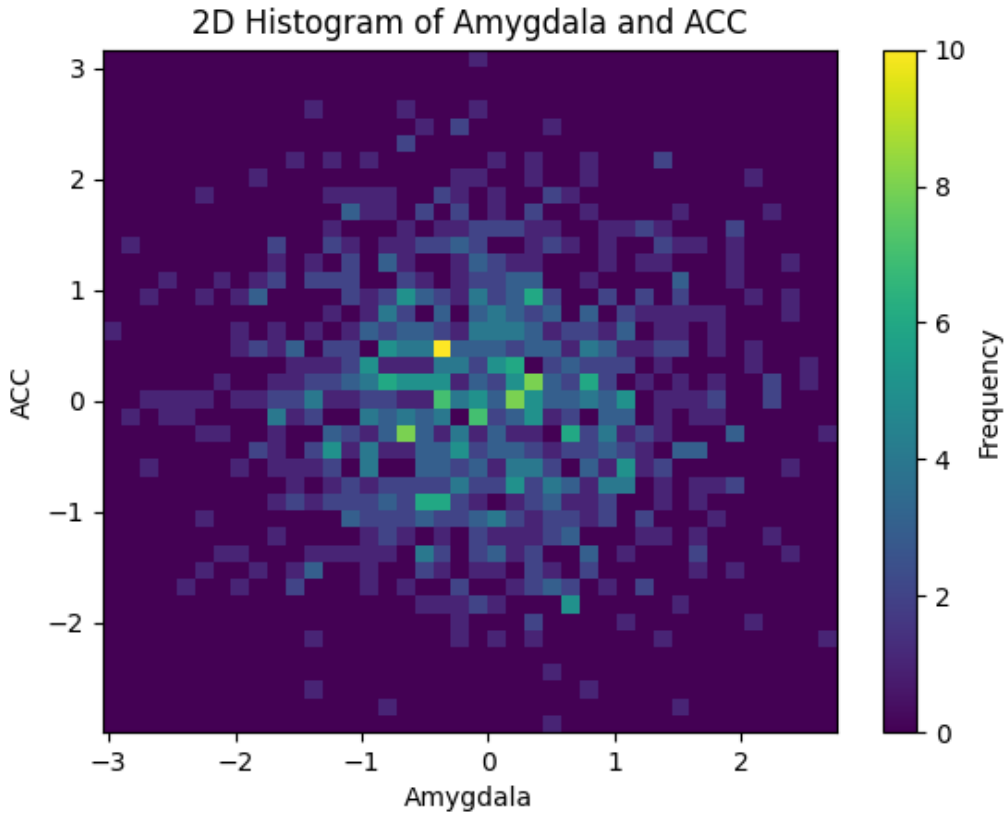

Figure 8: 2D Histogram

(c) (5 points) Use kernel-density-estimation (KDE) to estimate the 2-dimensional density function of (amygdala, acc) (this means for this question, you can ignore the variable orientation). Set an appropriate kernel bandwidth $h > 0$.

Please show the two-dimensional KDE (e.g., two-dimensional heat-map, two-dimensional contour plot, etc.)

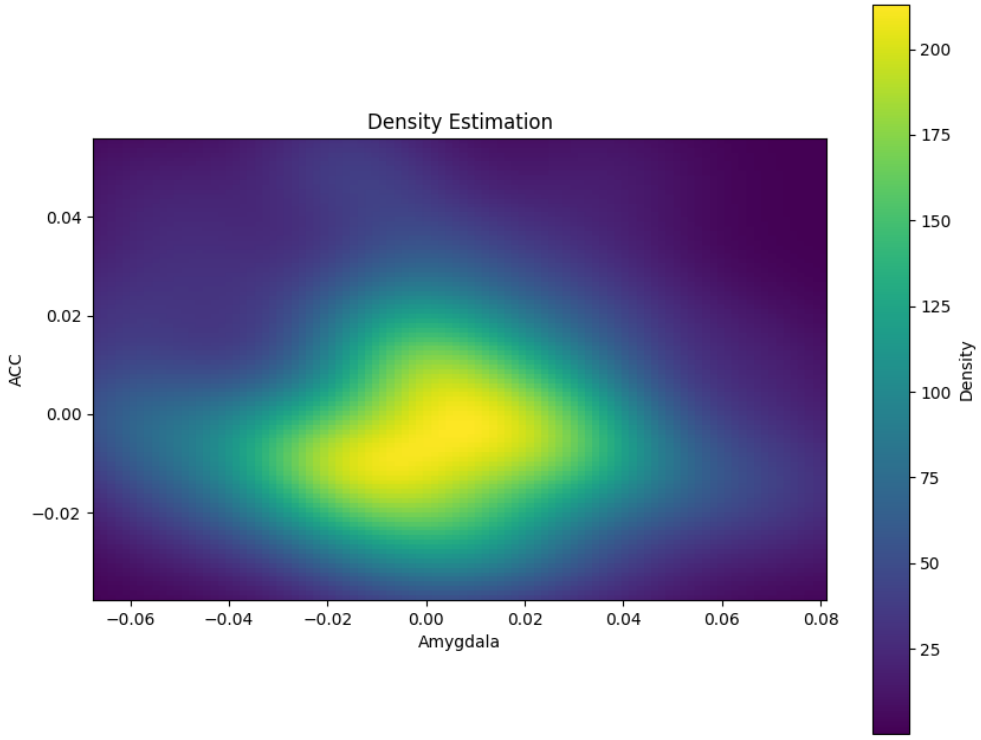

Figure 9: KDE density

Please explain what you have observed: is the distribution unimodal or bi-modal? Are there any outliers?

The distribution visualized is bimodal and some outliers are present.

(amygdala, acc)to be independent or not? Please support your argument with reasonable investigations.

ACC and amygdala are not independent because the density plot above shows how they are associated with each other.

(d) (10 points) We will consider the variable orientation and consider conditional distributions. Please plot the estimated conditional distribution of amygdala conditioning on political orientation: $p(\text{amygdala}|\text{orientation} = c)$, $c = 2, \ldots, 5$, using KDE. Set an appropriate kernel bandwidth $h > 0$. Do the same for the volume of the acc: plot $p(\text{acc}|\text{orientation} = c)$, $c = 2, \ldots, 5$ using KDE. (Note that the conditional distribution can be understood as fitting a distribution for the data with the same orientation. Thus you should plot 8 one-dimensional distribution functions in total for this question.)

Now please explain based on the results, can you infer that the conditional distribution of amygdala

and acc, respectively, are different from $c = 2, \ldots, 5$? This is a type of scientific question one could infer from the data: Whether or not there is a difference between brain structure and political view.

Now please also fill out the *conditional sample mean* for the two variables:

	2	3	4	5
Amygdala	0.019062	0.000588	-0.00472	-0.005692
ACC	-0.014769	0.001671	0.00131	0.008142

Figure 3: Figure 10: Conditional sample means

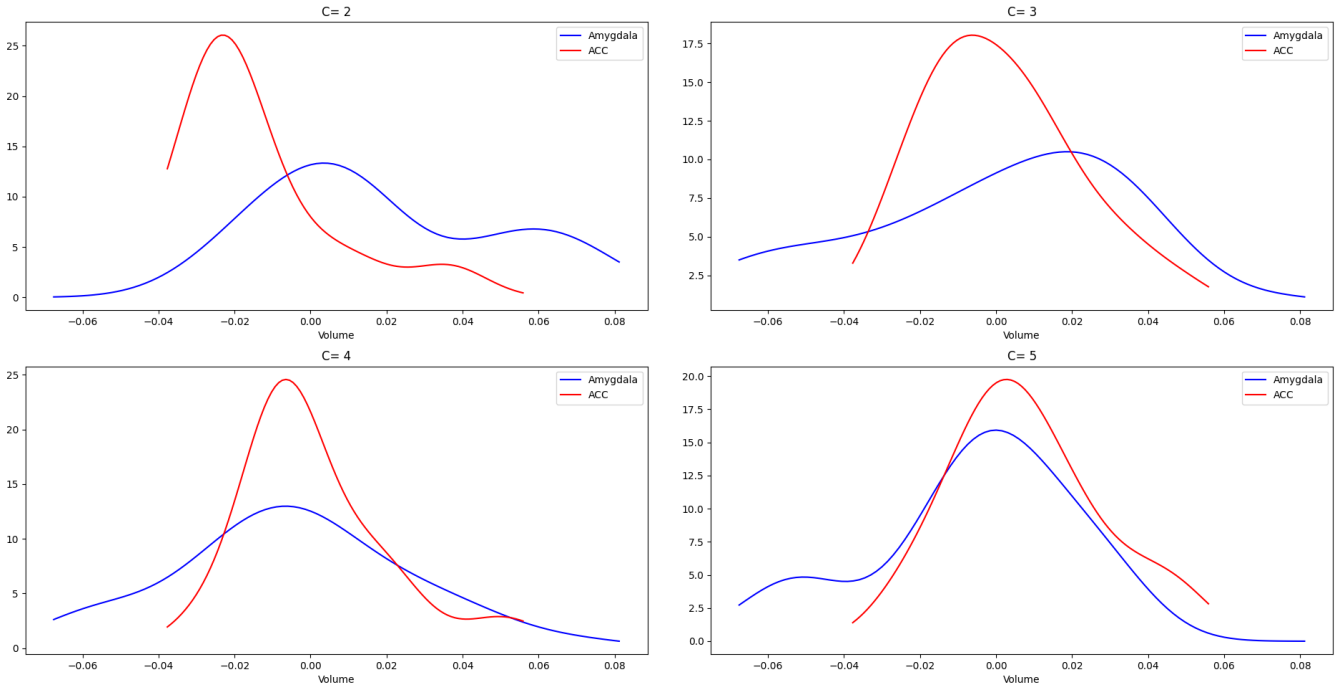

Figure 4: Figure 11: Conditional KDE plots

Remark: As you can see this exercise, you can extract so much more information from density estimation than simple summary statistics (e.g., the sample mean) in terms of explorable data analysis.

(e) (5 points) Again we will consider the variable orientation. We will estimate the conditional *joint* distribution of the volume of the amygdala and acc, conditioning on a function of political orientation: $p(\text{amygdala}, \text{acc}|\text{orientation} = c)$, $c = 2, \ldots, 5$. You will use two-dimensional KDE to achieve the goal; et an appropriate kernel bandwidth $h > 0$. Please show the two-dimensional KDE (e.g., two-dimensional heat-map, two-dimensional contour plot, etc.).

Please explain based on the results, can you infer that the conditional distribution of two variables (amygdala, acc) are different from $c = 2, \ldots, 5$? This is a type of scientific question one could infer from the data: Whether or not there is a difference between brain structure and political view.

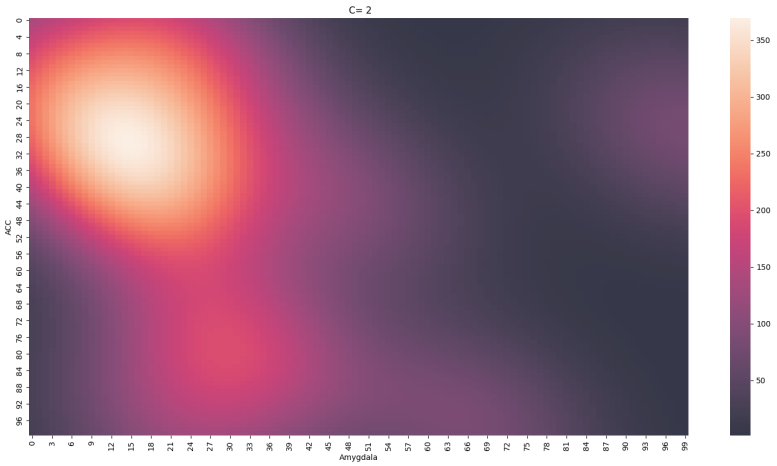

Figure 12: C=2

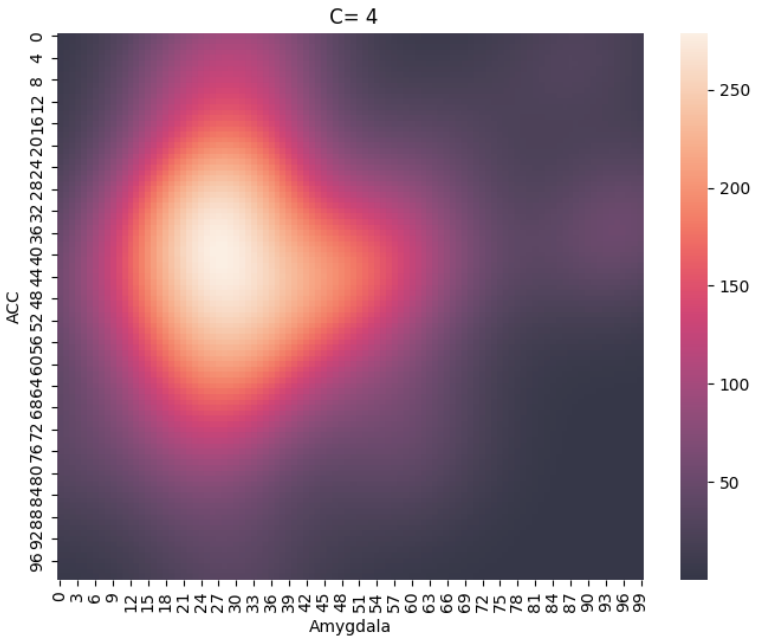

Figure 13: C=3

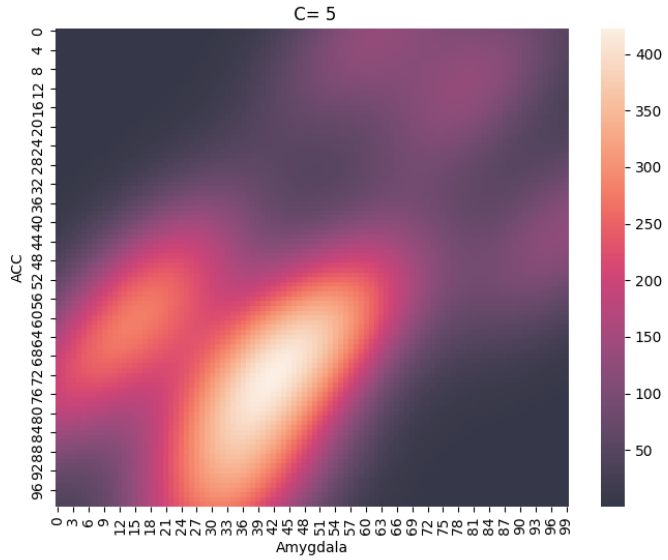

Figure 14: C=4

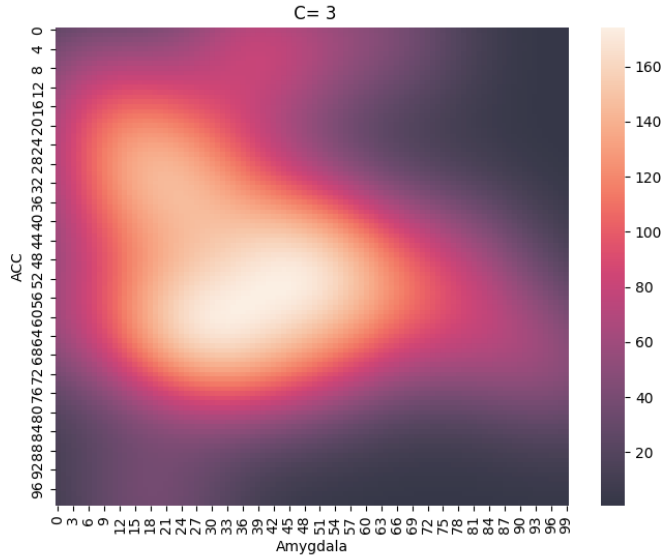

Figure 15: C=5

The plots show how different how the structure is based on politic view. however if exploring more plots we can see similarities as well.

5. Eigenfaces and simple face recognition [Bonus, 10 points].

This question is a simplified illustration of using PCA for face recognition. We will use a subset of data from the famous Yale Face dataset.

Remark: You will have to perform downsampling of the image by a factor of 4 to turn them into a lower resolution image as a preprocessing (e.g., reduce a picture of size 16-by-16 to 4-by-4). In this question, you can implement your own code or call packages.

First, given a set of images for each person, we generate the eigenface using these images. You will treat one picture from the same person as one data point for that person. Note that you will first vectorize each image, which was originally a matrix. Thus, the data matrix (for each person) is a matrix; each row is a vectorized picture. You will find weight vectors to combine the pictures to extract different "eigenfaces" that correspond to that person's pictures' first few principal components.

(a) (5 points) Perform analysis on the Yale face dataset for Subject 1 and Subject 2, respectively, using all the images EXCEPT for the two pictures named subject01-test.gif and subject02-test.gif. **Plot the first 6 eigenfaces for each subject.** When visualizing, please reshape the eigenvectors into proper images. Please explain can you see any patterns in the top 6 eigenfaces?

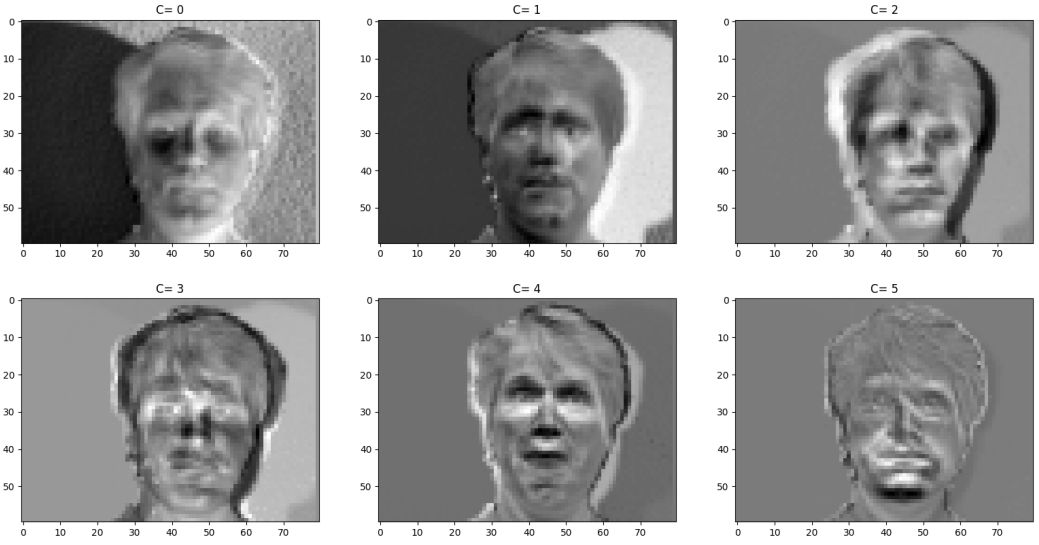

Figure 16: Subject 1

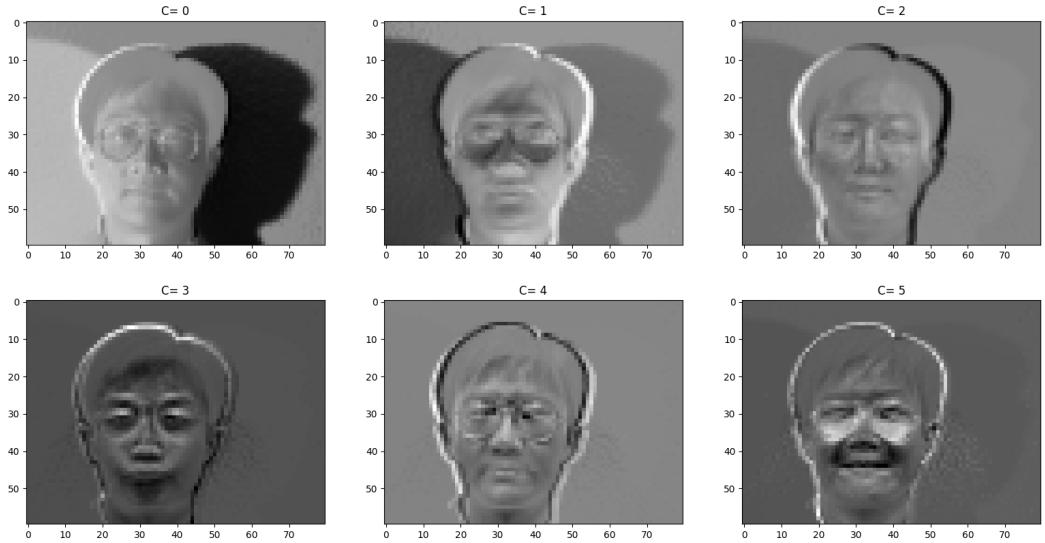

Figure 17: Subject 2
The subject pictures with higher eigenvalues are clearer to identify. Also, shadows impact the subjects ability to identify.

(b) (5 points) Now we will perform a simple face recognition task.

Face recognition through PCA is proceeded as follows. Given the test image subject01-test.gif and subject02-test.gif, first downsize by a factor of 4 (as before), and vectorize each image. Take the top eigenfaces of Subject 1 and Subject 2, respectively. Then we calculate the *projection residual* of the 2 vectorized test images with the vectorized eigenfaces:

$$s_{ij} = \|(\text{test image})_j - (\text{eigenface}_i)(\text{eigenface})_i^T(\text{test image})_j\|_2^2$$

Report all four scores: s_{ij}, $i = 1, 2$, $j = 1, 2$. Explain how to recognize the faces of the test images using these scores.

Comment if your face recognition algorithm works well and discuss how you would like to improve it if possible.