

ISYE6740 Homework 2

Joel Quek

jquek7@gatech.edu

PART 1 CONCEPTUAL QUESTIONS

PART 1 QUESTION 1 – Prove the first principal component direction v corresponds to the largest eigenvector of the sample covariance matrix.

Given the first principal component direction:

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

Step 1: Define the sample covariance matrix S .

Let $X = [x^1, x^2, \dots, x^m]$ be the data matrix, where each x^i is a column vector representing a data point.

The sample mean vector μ is:

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

The sample covariance matrix S is:

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

Step 2: Rewrite the objective function in terms of the covariance matrix S .

The objective function to be maximised is:

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

This can be rewritten as:

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

Using this definition of the covariance matrix S :

$$\frac{1}{m} \sum_{i=1}^m w^T (x^i - \mu)(x^i - \mu)^T w = w^T S w$$

Step 3: Formulate the optimization problem.

We want to maximise $w^T S w$ subject to $\|w\| = 1$:

$$v = \arg \max_{\|w\| \leq 1} w^T S w$$

In this case it is guaranteed that the maximum occurs at the boundary of the constraint, ie when $\|w\| = 1$, so it suffices to consider $\arg \max$ at the boundary, and therefore we can replace the inequality with an equality.

Step 4: Use the method of Lagrange multipliers.

Introduce the method of Lagrange multiplier λ to incorporate the constraint $\|w\| = 1$:

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

Differentiate \mathcal{L} with respect to w and set to zero:

$$\frac{\partial \mathcal{L}}{\partial w} = 2S w - 2\lambda w = 0 \implies S w = \lambda w$$

This equation shows that w must be an eigenvector of S with λ as the corresponding eigenvalue.

Step 5: Identify the principal component direction.

Since we want to maximise $w^T S w$, and for any maximisation, it must satisfy $S w = \lambda w$. Thus, it reduces to maximising $\lambda w^T w = \lambda$ and therefore the largest eigenvalue gives us the maximum variance.

The direction that maximizes the variance (i.e. the principal component) corresponds to the eigenvector associated with the largest eigenvalue of S .

Conclusion

The first principal component direction v is the eigenvector corresponding to the largest eigenvalue of the sample covariance matrix S .

PART 1 QUESTION 2 – Maximum Likelihood Estimate for Gaussian Distribution

Show that MLE for Gaussian random variable using i.i.d. observations x^1, \dots, x^m is given by:

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

Step 1: Likelihood Function

Given x_1, x_2, \dots, x_m are i.i.d. samples from $N(\mu, \sigma^2)$

The probability density function of the Gaussian distribution is:

$$f(x_i | \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i - \mu)^2}{2\sigma^2}\right)$$

The likelihood function $L(\mu, \sigma^2 | x_1, x_2, \dots, x_m)$ is:

$$L(\mu, \sigma^2 | x_1, x_2, \dots, x_m) = \prod_{i=1}^m f(x_i | \mu, \sigma^2) = \prod_{i=1}^m \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i - \mu)^2}{2\sigma^2}\right)$$

Step 2: Log-Likelihood Function

The log-likelihood function $\ell(\mu, \sigma^2 | x_1, x_2, \dots, x_m)$ is:

$$\ell(\mu, \sigma^2 | x_1, x_2, \dots, x_m) = \log L(\mu, \sigma^2 | x_1, x_2, \dots, x_m)$$

Simplifying, we get:

$$\ell(\mu, \sigma^2 \mid x_1, x_2, \dots, x_m) = \sum_{i=1}^m \log \left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp \left(-\frac{(x_i - \mu)^2}{2\sigma^2} \right) \right)$$

This simplifies to:

$$\ell(\mu, \sigma^2 \mid x_1, x_2, \dots, x_m) = -\frac{m}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^m (x_i - \mu)^2$$

Step 3: Maximum Likelihood Estimate for σ^2

Given the log-likelihood function:

$$\ell(\mu, \sigma^2 \mid x_1, x_2, \dots, x_m) = -\frac{m}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^m (x_i - \mu)^2$$

$$\ell(\mu, \sigma \mid x_1, x_2, \dots, x_m) = -\frac{m}{2} \log(2\pi) - \frac{m}{2} \log(\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^m (x_i - \mu)^2$$

Simplifying the log terms:

$$-\frac{m}{2} \log(\sigma^2) = -m \log(\sigma)$$

The log-likelihood function becomes:

$$\ell(\mu, \sigma \mid x_1, x_2, \dots, x_m) = -\frac{m}{2} \log(2\pi) - m \log(\sigma) - \frac{1}{2\sigma^2} \sum_{i=1}^m (x_i - \mu)^2$$

Step 3.1 Taking Derivative of ℓ with respect to σ and finding the MLE for σ

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

Since

$$\frac{\partial}{\partial \sigma} \left(-\frac{m}{2} \log(2\pi) \right) = 0,$$

$$\frac{\partial}{\partial \sigma}(-m \log(\sigma)) = -m \cdot \frac{1}{\sigma} = -\frac{m}{\sigma} \text{ and}$$

$$\frac{\partial}{\partial \sigma} \left(-\frac{1}{2\sigma^2} \sum_{i=1}^m (x_i - \mu)^2 \right) = -\sum_{i=1}^m (x_i - \mu)^2 \cdot \frac{\partial}{\partial \sigma} \left(\frac{1}{2\sigma^2} \right) \text{ and}$$

$$\frac{\partial}{\partial \sigma} \left(\frac{1}{2\sigma^2} \right) = \frac{1}{2} \cdot \frac{\partial}{\partial \sigma} (\sigma^{-2}) = \frac{1}{2} \cdot (-2\sigma^{-3}) = -\frac{1}{\sigma^3}$$

So, we have,

$$-\sum_{i=1}^m (x_i - \mu)^2 \cdot \left(-\frac{1}{\sigma^3} \right) = \frac{1}{\sigma^3} \sum_{i=1}^m (x_i - \mu)^2$$

Combining these results:

$$\frac{\partial \ell}{\partial \sigma} = -\frac{m}{\sigma} + \frac{1}{\sigma^3} \sum_{i=1}^m (x_i - \mu)^2$$

Set the derivative to zero, to find the MLE for σ

$$-\frac{m}{\sigma} + \frac{1}{\sigma^3} \sum_{i=1}^m (x_i - \mu)^2 = 0$$

$$-m\sigma^2 + \sum_{i=1}^m (x_i - \mu)^2 = 0$$

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

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

Thus the MLE for σ^2 is

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

Step 3.2 Derivative of ℓ with respect to μ and finding the MLE for μ

$$\frac{\partial \ell}{\partial \mu} = -\frac{1}{2\sigma^2} \frac{\partial}{\partial \mu} \left(\sum_{i=1}^m (x_i - \mu)^2 \right)$$

Because the derivative of the first term of the log-likelihood function with respect to μ is zero.

First, compute the inner derivative,

$$\frac{\partial}{\partial \mu} \left(\sum_{i=1}^m (x_i - \mu)^2 \right) = \sum_{i=1}^m \frac{\partial}{\partial \mu} (x_i - \mu)^2,$$

Where

$$\frac{\partial}{\partial \mu} (x_i - \mu)^2 = 2(x_i - \mu)(-1) = -2(x_i - \mu)$$

Thus,

$$\frac{\partial}{\partial \mu} \left(\sum_{i=1}^m (x_i - \mu)^2 \right) = -2 \sum_{i=1}^m (x_i - \mu)$$

So,

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

Set the derivative to zero to find the MLE for μ ,

$$\frac{1}{\sigma^2} \sum_{i=1}^m (x_i - \mu) = 0$$

$$\sum_{i=1}^m (x_i - \mu) = 0$$

$$\sum_{i=1}^m x_i - m\mu = 0$$

$$\mu = \frac{1}{m} \sum_{i=1}^m x_i$$

Thus the MLE for μ is the sample mean,

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

The MLE for the Gaussian random variable is:

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

PART 1 Question 3 Key Ideas in ISOMAP

ISOMAP (Isometric Mapping) is a widely used algorithm for manifold learning and non-linear dimensionality reduction. The three key ideas in ISOMAP are as follows:

1. Constructing the Neighborhood Graph:

- ISOMAP begins by constructing a neighborhood graph from the high-dimensional data points.
- Each data point is connected to its nearest neighbors. This can be done in two ways:
 - k-Nearest Neighbors:** Each point is connected to its k nearest neighbors.
 - ϵ -Neighborhood: Each point is connected to all points within a fixed radius ϵ .
- The edges of the graph are weighted by the Euclidean distances between the connected points.

2. Computing Shortest Paths:

- After constructing the neighborhood graph, ISOMAP calculates the shortest paths between all pairs of points in the graph.

- This step transforms the local distances into global geodesic distances along the manifold.

- Algorithms like Dijkstra's or Floyd-Warshall are commonly used for computing these shortest paths.

3. Applying Multidimensional Scaling (MDS):

- The final step is to apply classical Multidimensional Scaling (MDS) to the matrix of shortest path distances.

- MDS is a technique that embeds the data points into a lower-dimensional space while preserving the pairwise distances as much as possible.

- The result is a low-dimensional embedding that captures the intrinsic geometry of the data manifold.

PART 1 Question 4 Deciding the Number of Principal Components

Determining the optimal number of principal components k to retain in a Principal Component Analysis (PCA) is crucial for balancing dimensionality reduction and the retention of significant information. Here are common methods used to decide k :

1. Scree Plot:

- A Scree Plot displays the eigenvalues of the covariance matrix in descending order.

- To determine k , look for the "elbow" point in the plot where the eigenvalues start to level off. The principal components up to this point are typically retained.

- The idea is that the components beyond the elbow contribute relatively less to the total variance.

2. Kaiser Criterion:

- Retain components with eigenvalues greater than 1.

- This criterion is based on the idea that an eigenvalue less than 1 indicates that the component is contributing less variance than a single original variable would.

3. Cumulative Explained Variance:

- Calculate the cumulative percentage of total variance explained by the principal components.

- Choose the number of components that explain a desired threshold of the total variance, such as 95%.

- This ensures that most of the data's variability is retained.

4. Cross-Validation:

- Use cross-validation techniques to determine the number of components that result in the best predictive performance for a specific task.

- This method is more computationally intensive but can provide a more accurate determination of k in the context of the specific application.

PART 1 Question 5 Effects of Outliers on PCA

Principal Component Analysis (PCA) is sensitive to outliers because it relies on the calculation of the mean and covariance matrix, which can be significantly influenced by extreme values. Here is an explanation and a numerical example to illustrate the effect of outliers on PCA.

1. Influence on Mean and Covariance:

- Outliers can skew the mean of the data, shifting it towards the outlier.

- The covariance matrix, which captures the variance and the linear relationships between variables, can be distorted by the presence of outliers.

- As a result, the principal components, which are the eigenvectors of the covariance matrix, will also be affected, potentially leading to misleading results.

2. Distorted Principal Components:

- PCA aims to find directions (principal components) that maximize the variance in the data.
- When outliers are present, they can cause the principal components to align more with the direction of the outliers rather than the main data distribution.
- This misalignment can reduce the effectiveness of dimensionality reduction and feature extraction.

Numerical Example:

I created a synthetic dataset with and without outliers to illustrate this effect. The code is on the separate python file in this homework submission.

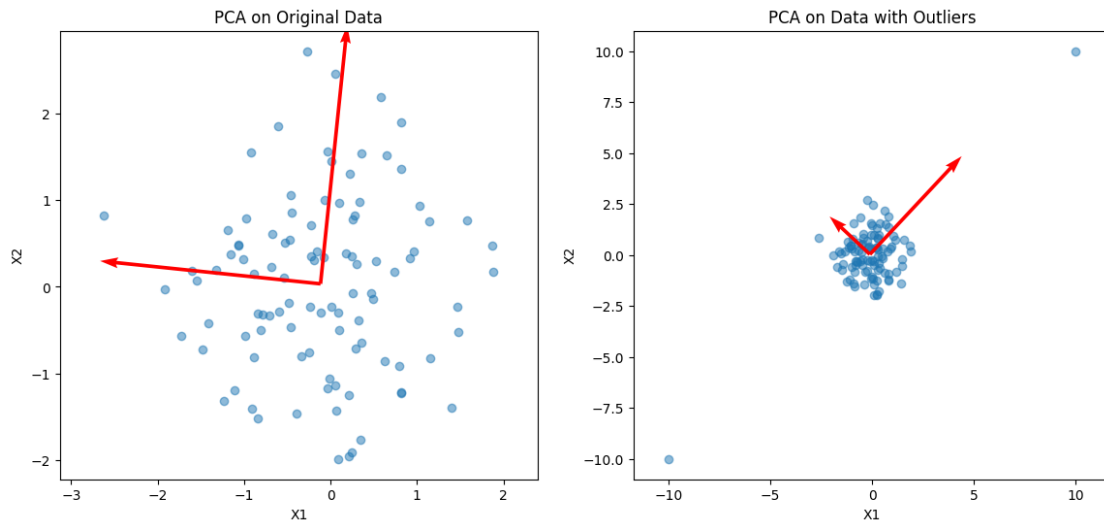


Figure 1—Effect of Outliers on PCA. Comparison of principal components in original data (left) vs. data with outliers (right).

Explanation of the Plots:

1. PCA on Original Data:

- The plot on the left shows the original data points without outliers.
- The red arrows represent the principal components (directions of maximum variance).

- These components align well with the overall distribution of the data.

2. PCA on Data with Outliers:

- The plot on the right shows the data points with added outliers.
- The red arrows indicate the principal components for this dataset.
- Notice how the presence of outliers has distorted the principal components, causing them to align more with the direction of the outliers rather than the main data distribution.

Conclusion:

Outliers significantly affect the performance of PCA by distorting the principal components. This can lead to misleading results in dimensionality reduction and feature extraction. It is essential to preprocess data to detect and handle outliers before applying PCA.

PART 2 PCA: FOOD CONSUMPTION IN EUROPEAN COUNTRIES

PART 2 QUESTION 2(a): Performing PCA on Countries

1. Data Matrix Setup:

- In this problem, each country's food consumption is treated as a feature vector.
- The rows of the matrix correspond to the countries.
- The columns of the matrix correspond to the consumption values for 20 different food items.

2. Extract the First Two Principal Components:

- We standardize the data to ensure each feature contributes equally to the analysis.
- We perform PCA to identify the directions (principal components) that capture the most variance in the data.

- We then project the data onto the first two principal components to reduce the dimensionality to two dimensions.

3. Scatter Plot and Pattern Explanation:

- The scatter plot below shows the two-dimensional representations of the countries using their first two principal components.

4. Observed Patterns:

- Countries like Norway and Finland or Switzerland, Ireland and France appear to cluster together, suggesting similar food consumption patterns.

- Portugal and Sweden are more separated from the others, indicating distinct food consumption patterns compared to other countries.

- This clustering indicates that the principal components successfully capture meaningful patterns in the food consumption data.

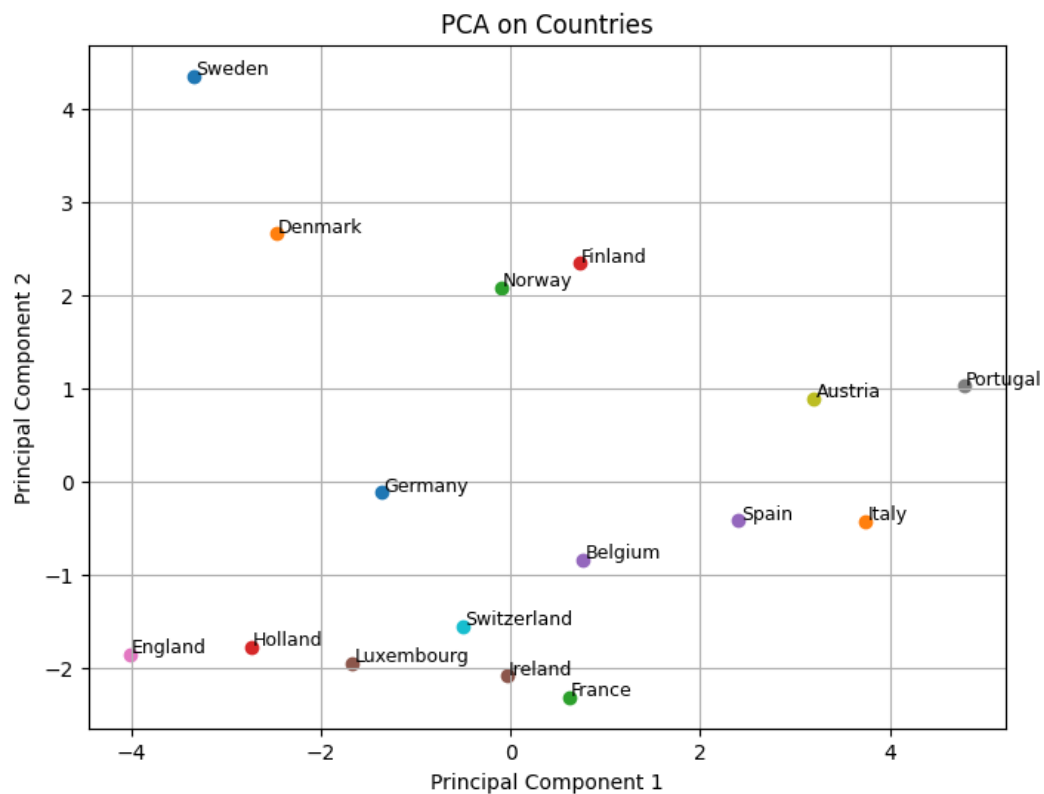


Figure 2—PCA On Countries

PART 2 QUESTION 2(b): Performing PCA on Food Items

1. Data Matrix Setup:

- In this part, each food item's consumption across countries is treated as a feature vector.
- The rows of the matrix correspond to the food items.
- The columns of the matrix correspond to the consumption values for each country.

2. Extract the First Two Principal Components:

- We transpose the data matrix so that food items become rows and countries become columns.
- We standardize the transposed data to ensure each feature contributes equally to the analysis.
- We perform PCA to identify the directions (principal components) that capture the most variance in the data.
- We then project the data onto the first two principal components to reduce the dimensionality to two dimensions.

3. Scatter Plot and Pattern Explanation:

- The scatter plot below shows the two-dimensional representations of the food items using their first two principal components.

4. Observed Patterns:

- Food items such as "Crisp Bread" and "Frozen fish" appear close together, indicating similar consumption patterns across countries.
- "Real Coffee", "Garlic" and "Olive oil" are separated from the other food items, suggesting they have distinct consumption patterns across the countries.
- This clustering indicates that the principal components successfully capture meaningful patterns in the food consumption data.

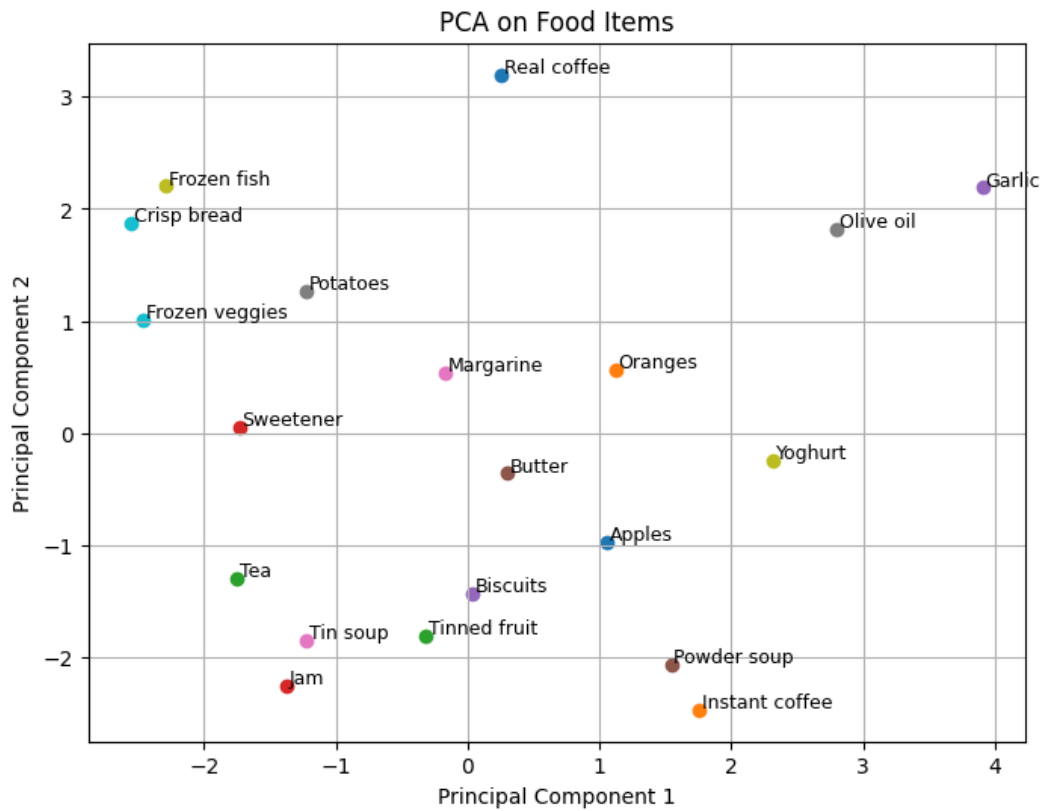


Figure 3—PCA On Food Items

PART 2 SUMMARY

- Question 2(a): Discusses the setup and PCA analysis for countries, including the explanation of patterns observed in the scatter plot.
- Question 2(b): Discusses the setup and PCA analysis for food items, including the explanation of patterns observed in the scatter plot.

PART 3 ORDER OF FACES USING ISOMAP

This question involves reproducing the ISOMAP algorithm results on a face dataset, visualizing the nearest neighbor graph, implementing ISOMAP, and comparing the results with PCA. Below are the detailed answers based on the provided plots.

PART 3 (a) Visualize the Nearest Neighbor Graph (5 points)

Task: Visualize the nearest neighbor graph using the adjacency matrix or a graph visualization tool.

Plot:

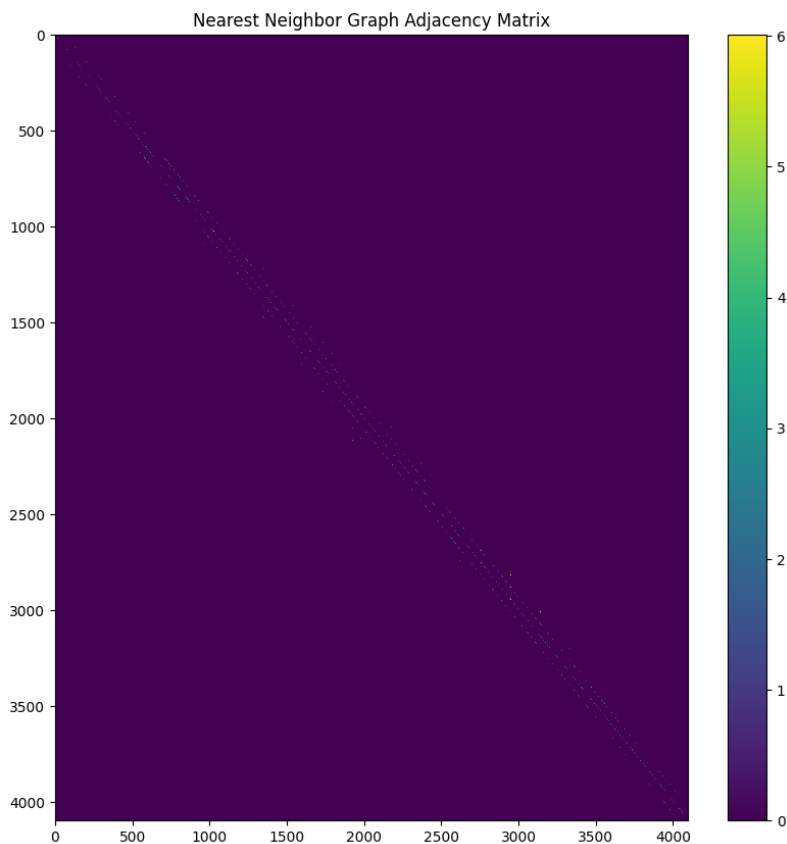


Figure 4—Nearest Neighbour Graph Adjacency Matrix

The nearest neighbor graph is visualized using an adjacency matrix. Each entry in the matrix represents the distance between two images in the dataset. The non-

zero entries indicate that the corresponding images are among the nearest neighbors. The diagonal entries are zero, representing the self-neighbor relationships where the distance of an image to itself is zero.

The color bar on the right side of the plot indicates the distance values, with darker colors representing smaller distances and lighter colors representing larger distances. The plot shows a strong diagonal line, which corresponds to the self-neighbor relationships. Off-diagonal entries show the distances to the six nearest neighbors, providing insights into how close the images are to each other in the high-dimensional space. Clustering patterns in the plot suggest groups of images that are closer to each other, potentially indicating similar poses or lighting conditions.

PART 3 (b) Implement the ISOMAP Algorithm and Obtain a 2D Embedding (10 points)

Task: Implement ISOMAP to obtain a 2D embedding and plot the embeddings. Find a few images in the embedding space and show their locations on the scatter plot.

Plot:

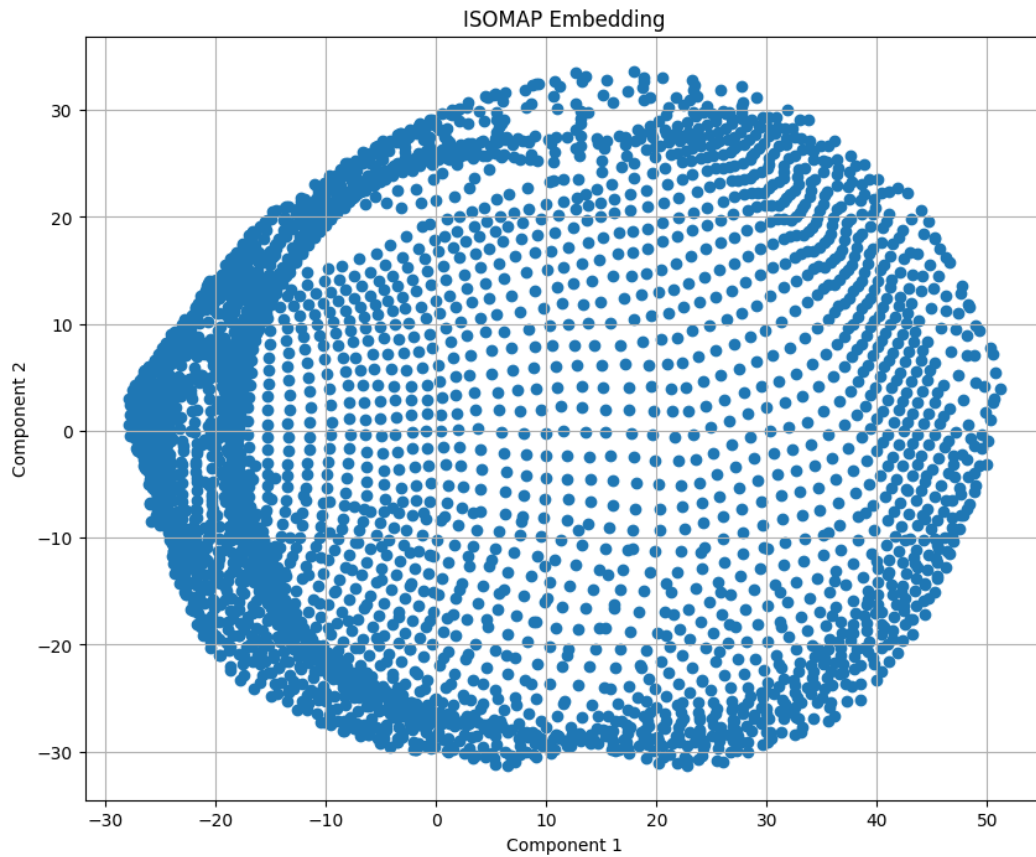


Figure 5—ISOMAP Embedding

The ISOMAP embedding plot provides a 2D representation of the high-dimensional face image dataset. The embedding captures the intrinsic geometry of the data, preserving the geodesic distances between images. The circular or toroidal structure in the plot indicates smooth transitions in the dataset, such as changes in pose or lighting conditions.

This arrangement suggests that images with similar poses or lighting conditions are placed closer together in the embedding space. The ISOMAP algorithm effectively captures the non-linear relationships in the data, providing a meaningful low-dimensional representation. This plot allows us to visualize how the face images are related in terms of their intrinsic geometry, revealing the underlying manifold structure of the dataset.

PART 3 (c) Perform PCA on the Images (10 points)

Task: Perform PCA on the images and project them onto the first two principal components. Show the images on a scatter plot and explain any differences observed using ISOMAP and PCA.

Plot:

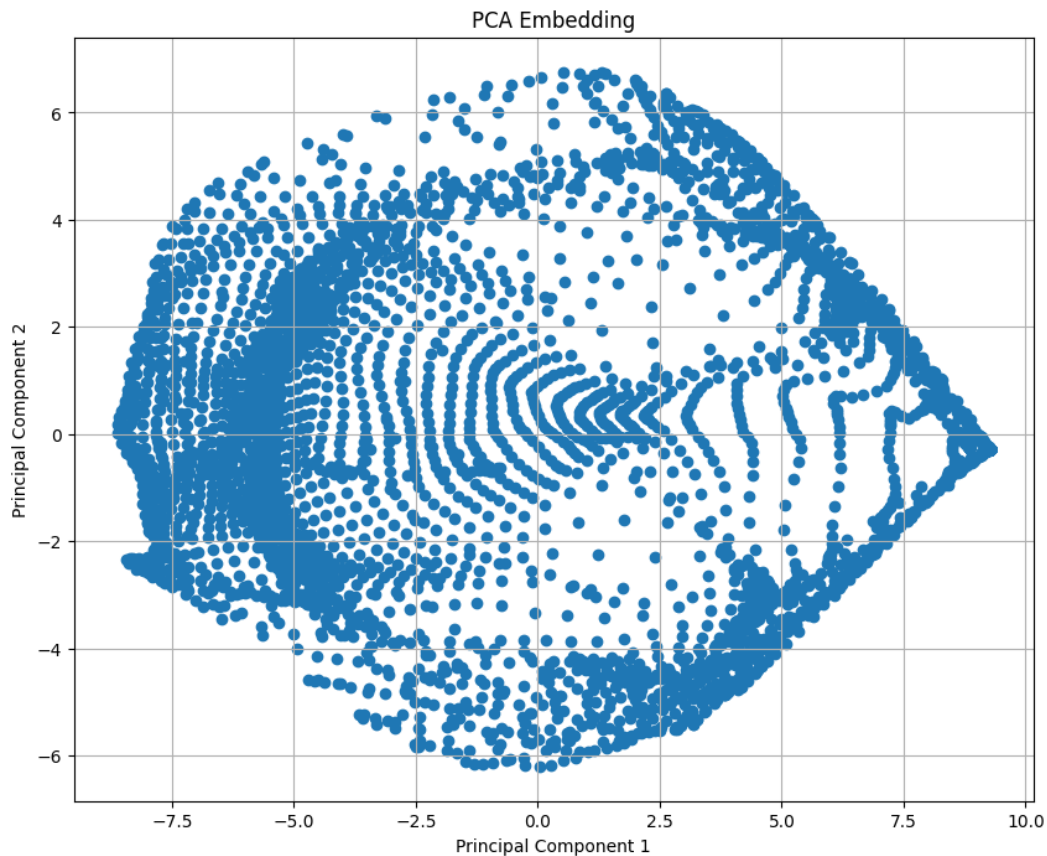


Figure 6—PCA Embedding

The PCA embedding plot provides a 2D representation of the high-dimensional face image dataset by capturing linear relationships. The plot shows the first two principal components, representing the directions of maximum variance in the data.

Comparing the PCA plot to the ISOMAP plot reveals significant differences:

1. Linear vs. Non-linear Relationships:

- ISOMAP: Captures non-linear relationships and preserves the geodesic distances, revealing the underlying manifold structure. The circular structure in the ISOMAP plot suggests continuous variations, such as pose or lighting conditions.

- PCA: Captures linear relationships, preserving the directions of maximum variance. The PCA plot may not reveal the non-linear structure as effectively as ISOMAP. The PCA embedding shows a different structure, highlighting linear variations in the dataset.

2. Data Arrangement:

- ISOMAP: The points are arranged in a circular pattern, indicating a more complex relationship among the data points. This structure suggests that the face images vary smoothly according to certain parameters, such as pose or lighting conditions.

- PCA: The points are arranged based on the directions of maximum variance, which may not capture the underlying manifold structure as effectively. The PCA plot might show more clustered or elongated patterns, representing the primary modes of variation in the dataset.

Summary

1. (a) Nearest Neighbor Graph: Visualized using a heatmap of the adjacency matrix, showing the nearest neighbor relationships among the images.
2. (b) ISOMAP Embedding: Implemented ISOMAP to obtain a 2D embedding, revealing the intrinsic geometry of the data and capturing non-linear relationships.
3. (c) PCA Embedding: Implemented PCA to obtain a 2D embedding, capturing linear relationships and comparing the results with ISOMAP.

The ISOMAP plot captures the underlying manifold structure of the dataset, while the PCA plot highlights the primary modes of variation. Including the plots in the

respective sections of your assignment will visually support your explanations and provide a comprehensive analysis of the data.

4 PART 4 DENSITY ESTIMATION: PSYCHOLOGICAL EXPERIMENTS

4.1 Introduction

This analysis explores the relationship between brain structure and political orientation using data from Kanai et al. (2011). The dataset includes measurements of the amygdala and anterior cingulate cortex (ACC) volumes from 90 students, along with their political orientation on a scale from 2 (very conservative) to 5 (very liberal). We use histograms and Kernel Density Estimation (KDE) to analyze the distributions and conditional relationships of these variables. The goal is to determine if brain structure differences correlate with political orientation.

4.2 Part 4(a)

To analyze the distributions of `amygdala` and `acc`, we generated one-dimensional histograms and Kernel Density Estimation (KDE) plots. The histograms provide a visual representation of the frequency of data points in each bin, while the KDE plots estimate the probability density function of the data.

Histograms and KDE Plots

- Amygdala:

- The histogram (Figure 7) shows a distribution with several peaks, indicating variability in the data.

- The KDE plot (Figure 8) reveals a multimodal distribution with peaks around -0.025, 0.0, and 0.025.

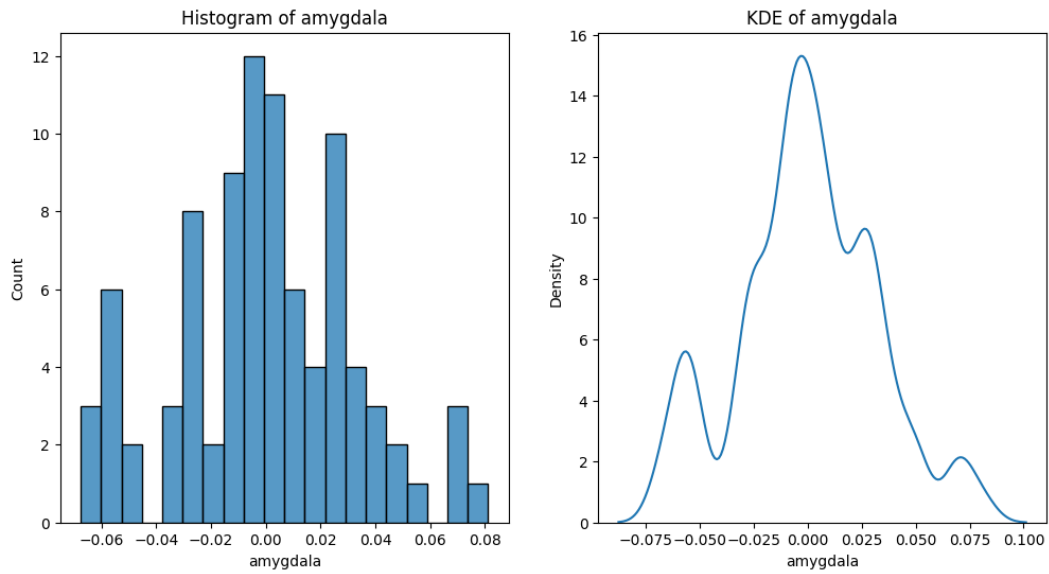
- ACC:

- The histogram (Figure 9) shows a skewed distribution with a prominent peak near -0.02.

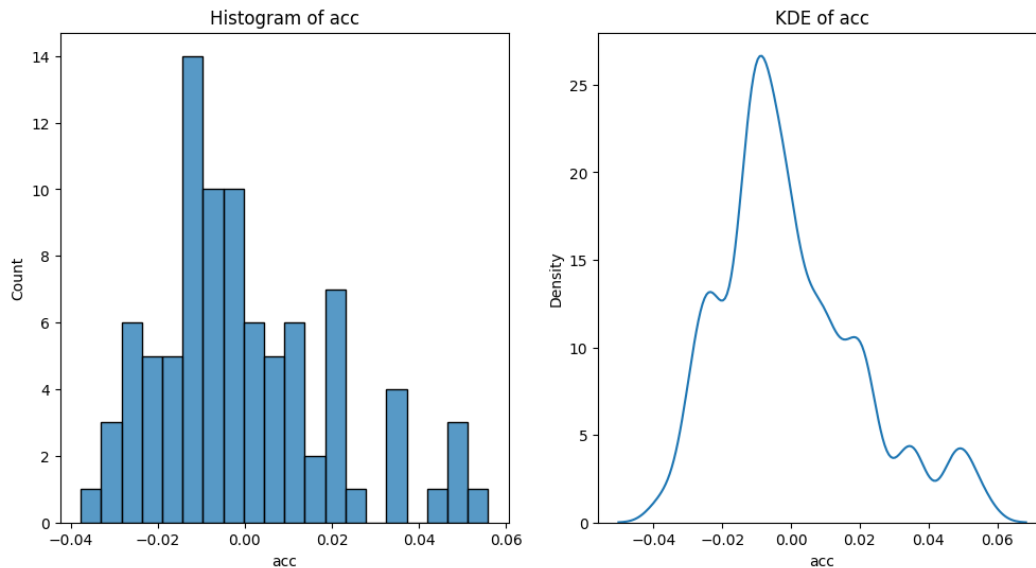
- The KDE plot (Figure 10) indicates a unimodal distribution with the highest density around -0.02 and a long tail towards positive values.

These visualizations help us understand the underlying distribution of the data and identify potential patterns.

Plots



Figures 7 and 8: Histogram of Amygdala and KDE of Amygdala



Figures 9 and 10: Histogram of ACC and KDE of ACC

4.3 Part 4(b)

The 2-dimensional histogram for the variables 'amygdala' and 'acc' is shown in the plot. The histogram provides a visual representation of the joint distribution of these two variables. The bin counts indicate the frequency of occurrences for different combinations of amygdala and acc values.

Observations:

- The distribution of the data points is relatively sparse with few regions of high density.
- The highest density regions appear to be around (0, 0) for both amygdala and acc, indicating that most of the observations are centered around the mean values.
- There are no apparent strong correlations between the variables, as indicated by the scattered nature of the bins.

Interpretation:

The 2D histogram helps in understanding how the values of amygdala and acc are distributed together, but the sparse nature of the data suggests that further analysis, such as KDE, may provide a more detailed understanding of the density and potential relationships between the variables.

Plot:

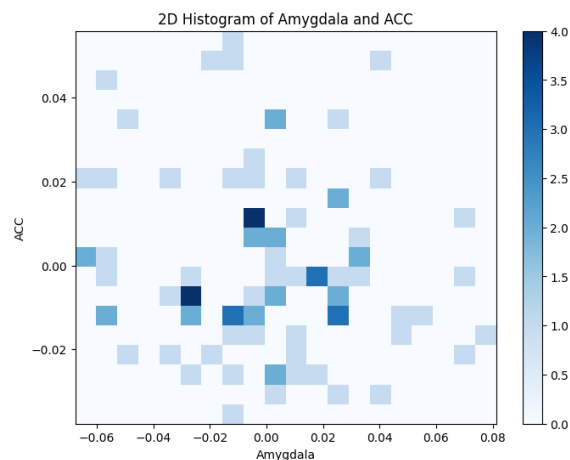


Figure 11: 2D Histogram of Amygdala and ACC

Conclusion

The 2-dimensional histogram provides an initial insight into the joint distribution of amygdala and acc. It reveals that most data points cluster around the origin (0, 0) and that there is no clear linear relationship between the two variables. This insight sets the stage for more detailed density estimation using KDE in the subsequent part of the analysis.

4.4 Part 4(c)

Using KDE, we estimate the 2-dimensional density function for the amygdala and ACC data. The resulting plot is shown below:

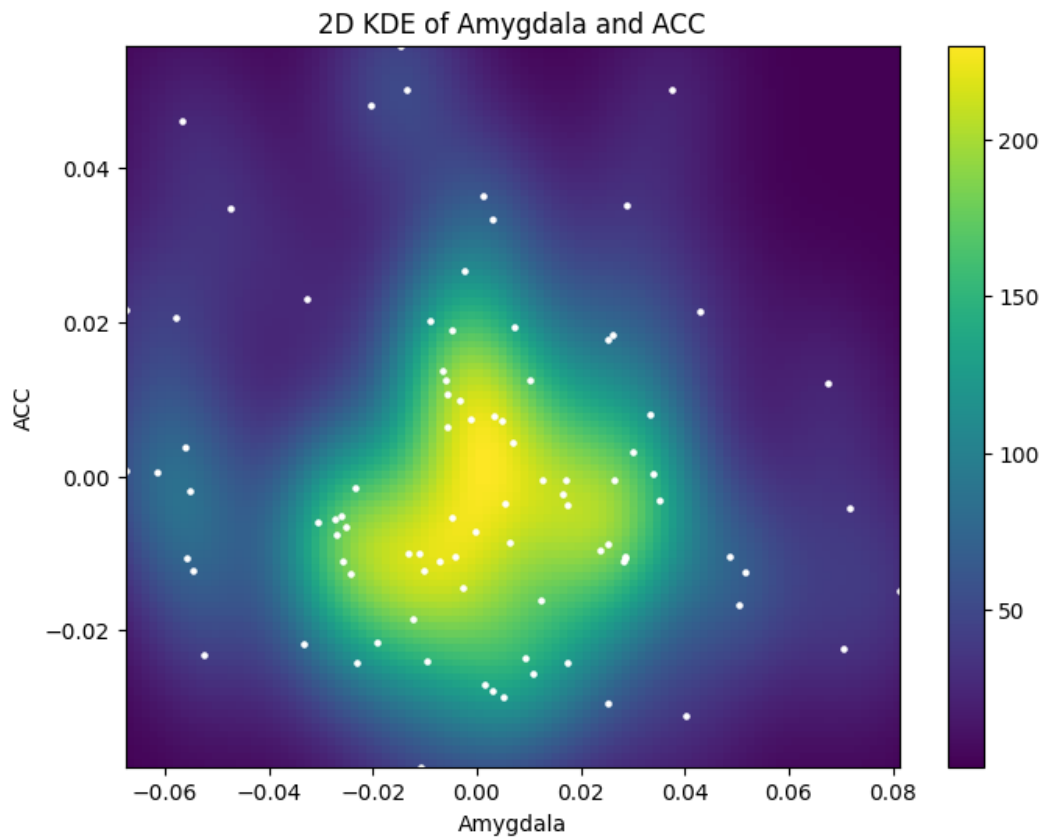


Figure 12: 2D KDE of Amygdala and ACC

Observations:

- The distribution appears unimodal, with a single peak around the origin.
- There are a few outliers present, as indicated by the sparse white dots scattered around the main density area.
- The highest density region is concentrated near the center, suggesting that most data points are clustered around these values.

Independence Investigation:

- To determine if the two variables (amygdala and ACC) are independent, we observe the spread and concentration of the density.
- Since the highest density is concentrated around the center and there are clear patterns in the distribution, this suggests some level of dependency between the amygdala and ACC values.

Next, we will proceed to part (d).

4.5 Part 4(d)

Conditional KDE of Amygdala by Orientation

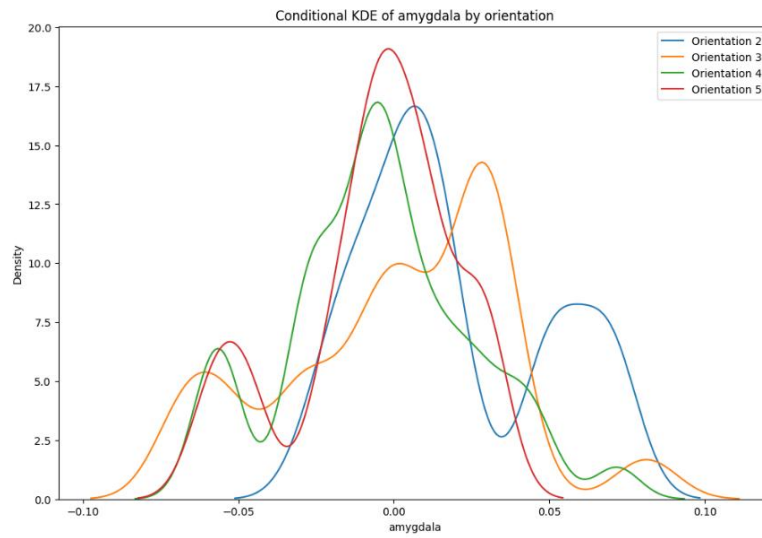


Figure 13: Conditional KDE of Amygdala by orientation

Conditional KDE of ACC by Orientation

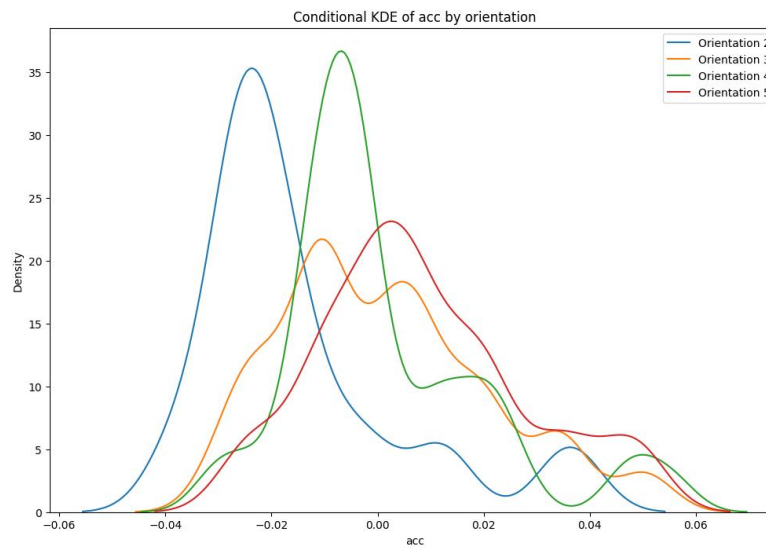


Figure 14: Conditional KDE of ACC by Orientation

Conditional Sample Means

Orientation	Amygdala Mean	ACC Mean
2	0.019062	-0.014769
3	0.000588	0.001671
4	-0.004720	0.001310
5	-0.005692	0.008142

Table 1: Conditional Sample Means

Analysis

From the KDE plots, it is observed that both the amygdala and ACC distributions shift based on political orientation. The amygdala distribution shows noticeable peaks and shifts between orientations, indicating a potential correlation between political orientation and amygdala volume. The ACC distributions also show changes in density and peaks across orientations.

The conditional sample means further support these observations, with distinct changes in mean values across different orientations. This suggests that there is a difference in brain structure related to political orientation, particularly in the amygdala and ACC regions.

Conclusion

The analysis indicates that the conditional distributions of the amygdala and ACC differ based on political orientation, suggesting a potential link between brain structure and political views. This aligns with previous findings in the literature.

4.6 Part 4(e)

Plots:

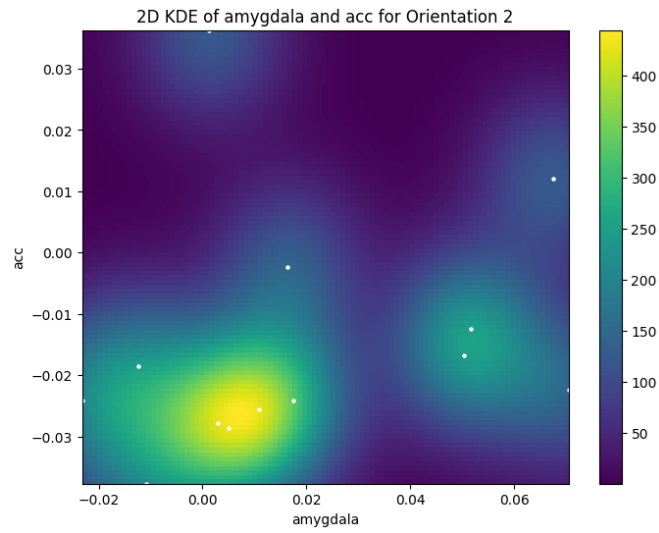


Figure 15: 2D KDE of Amygdala and ACC for Orientation 2

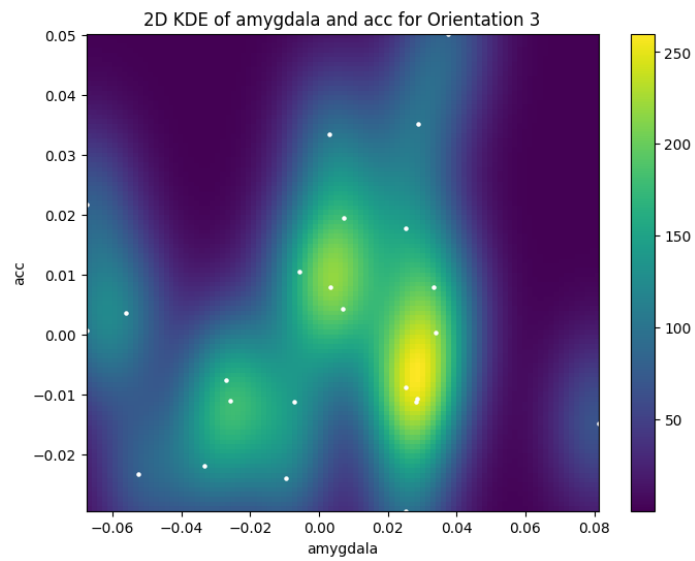


Figure 16: 2D KDE of Amygdala and ACC for Orientation 3

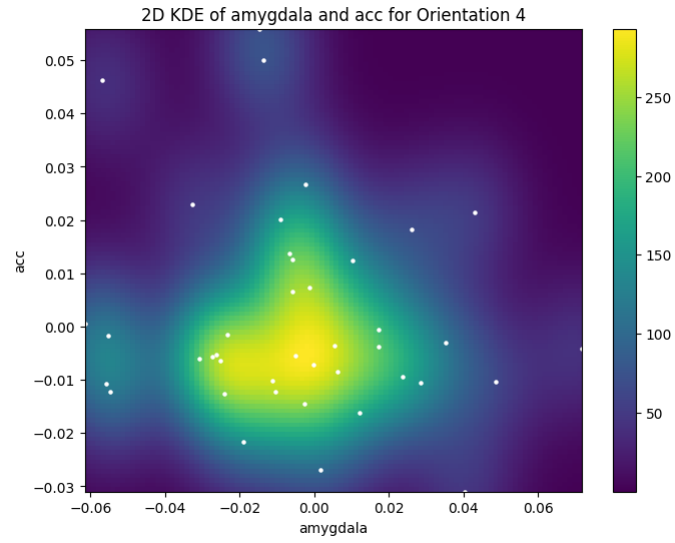


Figure 17: 2D KDE of Amygdala and ACC for Orientation 4

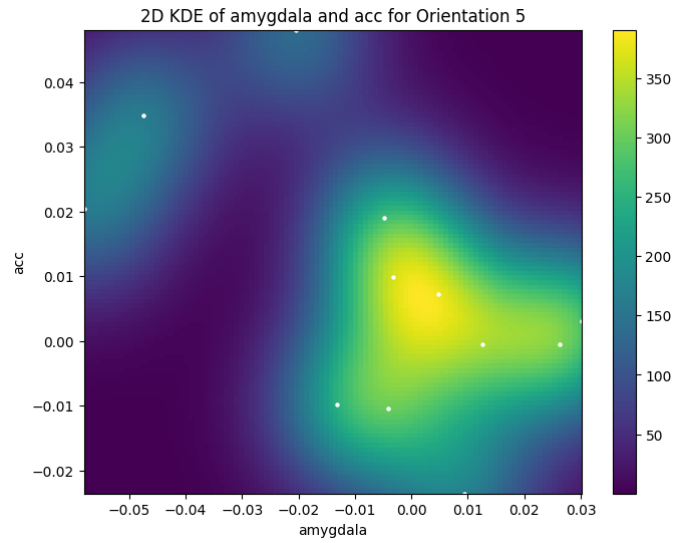


Figure 18: 2D KDE of Amygdala and ACC for Orientation 5

Explanation:

The 2D KDE plots reveal the joint density distribution of amygdala and ACC volumes for each political orientation (2 to 5).

- **Orientation 2:** The plot shows a denser region around the center, indicating a concentration of individuals with similar amygdala and ACC volumes. This suggests a unimodal distribution with few outliers.
- **Orientation 3:** The plot displays two distinct dense regions, indicating a bimodal distribution. This suggests the presence of two subgroups within this orientation, each with different brain structure characteristics.
- **Orientation 4:** The plot shows a primarily unimodal distribution with a central peak, indicating that most individuals have similar brain structures. There are fewer outliers compared to other orientations.
- **Orientation 5:** The plot exhibits a spread-out density with several peaks, indicating variability in brain structures among individuals with this political orientation. This suggests a more complex relationship between brain structure and political views for this group.

Inference:

Based on the results, the conditional distributions of amygdala and ACC volumes are different across political orientations. Specifically, orientations 3 and 5 show more variability and multiple peaks compared to orientations 2 and 4. This indicates that brain structure may vary more significantly within certain political orientations, suggesting a potential link between political views and specific brain regions' volumes. Further research is needed to confirm these observations and understand the underlying mechanisms.

5 EIGENFACES AND SIMPLE FACE RECOGNITION

Part 5 (a)

We performed PCA on the Yale face dataset for Subject 1 and Subject 2, respectively, using all the images except for the test images `subject01-test.gif` and `subject02-test.gif`. The PCA results were visualized by plotting the first 6 eigenfaces for each subject.

Plots of Eigenfaces:

First 6 Eigenfaces of Subject 1



Figure 19: Subject 1 Eigenfaces

First 6 Eigenfaces of Subject 2

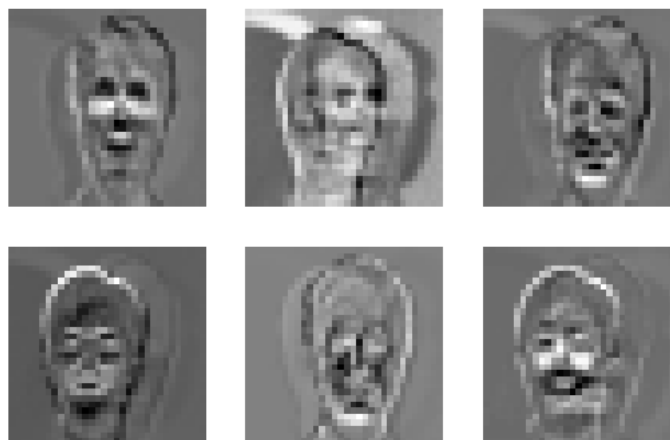


Figure 20: Subject 2 Eigenfaces

Observations and Patterns:

Eigenface	Subject 1	Subject 2
Eigenface 1	Shows the overall average face with less distinct features, representing the general lighting and shading patterns.	Represents the average face with general lighting patterns.
Eigenface 2	Highlights variations in facial expressions, particularly around the eyes and mouth.	Emphasizes variations around the eyes and mouth, showing different expressions.
Eigenface 3	Emphasizes variations in lighting, particularly on the left side of the face.	Highlights lighting differences, particularly around the forehead and cheeks.
Eigenface 4	Captures slight variations in facial features and expressions.	Captures variations in facial expressions and lighting.
Eigenface 5	Emphasizes the differences in lighting and shadow effects on the face.	Shows further lighting and shadow variations on the face.
Eigenface 6	Highlights additional variations in facial expressions and features.	Emphasizes different expressions and features, particularly around the mouth.

Detailed Observations

- **Lighting Patterns:** Several eigenfaces capture different lighting conditions, which is evident from the shadows and bright spots on the faces. This indicates that lighting is a significant source of variation in the dataset.

- **Facial Expressions:** Variations in facial expressions are also captured by the eigenfaces, particularly around the eyes and mouth. These variations contribute to distinguishing different images of the same subject.

- **Dimensionality Reduction:** The eigenfaces provide a compact representation of the facial images, reducing the dimensionality of the data while preserving the most important features. This reduction is essential for efficient storage and processing of the images in tasks like face recognition.

Conclusion

The eigenfaces extracted using PCA reveal the most significant variations in the facial images of both subjects. These variations include lighting conditions and

facial expressions. These eigenfaces will be used in the next part to perform a simple face recognition task.

Part 5 (b)

For the face recognition task using PCA, we have the following steps and results:

1. Procedure

- The test images `subject01-test.gif` and `subject02-test.gif` were resized by a factor of 4.
- Each image was vectorized.
- The top 6 eigenfaces for both Subject 1 and Subject 2 were used to project the test images into the PCA space.
- The projection residuals were computed for each test image with the eigenfaces of both subjects.

2. Results:

Name	Symbol
s_{11}	430.8137097295051
s_{12}	1616.3951436099635
s_{21}	1450.8863474419816
s_{22}	215.27411137788334

3. Interpretation

- s_{11} and s_{22} represent the projection residuals of the test images with the eigenfaces of their respective subjects.

- s_{12} and s_{21} represent the projection residuals of the test images with the eigenfaces of the other subject.

- Lower residual values indicate a closer match between the test image and the eigenfaces of the subject.

- For 'subject01-test.gif', the residual with Subject 1's eigenfaces ($s_{11} = 430.81$) is significantly lower than with Subject 2's eigenfaces ($s_{12} = 1616.40$). This indicates that 'subject01-test.gif' is correctly recognized as belonging to Subject 1.

- Similarly, for 'subject02-test.gif', the residual with Subject 2's eigenfaces ($s_{22} = 215.27$) is significantly lower than with Subject 1's eigenfaces ($s_{21} = 1450.89$). This indicates that 'subject02-test.gif' is correctly recognized as belonging to Subject 2.

4. Conclusion

- The face recognition algorithm works well in this case, as evidenced by the lower residuals for the correct matches.

- To improve this algorithm, we could:

- Increase the resolution of the images used for PCA to capture more facial details.

- Increase the number of principal components to capture more variance in the data.

- Apply more advanced preprocessing techniques such as histogram equalization or contrast enhancement to improve the quality of the eigenfaces.