Implementing Principal Component Analysis (PCA) on Random and Standard Datasets (June 2023)

Bishwambhar Dahal[1] and Sirjana Bhatta[1]

[1]Department of Electronics and Computer Engineering, IOE, Thapathali Campus, Kathmandu 44600, Nepal
Corresponding author: Bishwambhar Dahal(dahalbishwambhar@gmail.com)

Abstract In this study, we explore the application of Principal Component Analysis (PCA) as a dimensionality reduction technique on random, Iris, and Wine recognition datasets. PCA is widely used to transform high-dimensional datasets into lower-dimensional representations while preserving crucial information. By implementing PCA from scratch, including steps such as zero-mean normalization, covariance matrix computation, and extraction of principal components and eigenvalues, we investigate its behavior and performance. Through our analysis, we gain insights into the underlying mechanisms of PCA and evaluate its effectiveness in reducing dimensionality while maintaining important information in the datasets. The results provide valuable insights into the impact of PCA on the random, Iris, and Wine recognition datasets, shedding light on its potential applications in data analysis.

Keywords *Covariance matrix, Principal Component Analysis (PCA), zero-mean normalization.*

I Introduction

The ”Curse of dimensionality” [1] refers the increase in computational complexity and performance deterioration that occurs with high-dimensional data. To mitigate this issue, it is important to reduce the dimensionality of the data. One widely adopted approach is Principal Component Analysis (PCA) [2]. PCA transforms high-dimensional data into a lower-dimensional representation without losing essential information. The new dimensions, referred to as principal components, are linear combinations of the original features. PCA aims to maximize the amount of information captured while minimizing redundancy and noise.The classes within the datasets remained distinguishable, just as they were in higher-dimensional data. Moreover, the reduced dimensionality of the data required less computational power, making it suitable for various algorithms. In addition to this, PCA helps for better data visualization as lower dimensional data can be easily analyzed and plotted in graph. PCA not only alleviate the curse of dimensionality but also enhance the efficiency and effectiveness of subsequent analysis on the datasets.

In our lab, we conduct experiments on three datasets: a randomly generated dataset, the Iris dataset and the Wine dataset. Random datasets allow us to examine the performance and behavior of PCA when applied to synthetic data with varying characteristics. On the other hand, standard datasets provide us with real-world examples to assess the effectiveness of PCA in practical scenarios.Through this technique, we observed that the datasets retained valuable information even in lower dimensions. Our objectives encompass two main aspects. Firstly, we demonstrate the effectiveness of PCA in reducing the dimensionality of random datasets while preserving valuable information. We will analyze how PCA performs in capturing the underlying structure of the data and assess the impact of different parameters, such as the number of principal components, on the results.Secondly, we explore the application of PCA on standard datasets, specifically focusing on datasets commonly used in the field. By applying PCA to these well-known datasets, we evaluate and analyze its ability to reveal the inherent patterns and relationships within the data, potentially leading to improved insights and decision-making. This lab aims to develop understanding of PCA and its practical implementation on different types of datasets. By evaluating its performance and exploring its applications, we hope to shed light on the benefits and considerations of using PCA as a dimensionality reduction technique in various data analysis tasks.

II Methodology

A Brief Theory

The key idea behind Principal Component Analysis (PCA) is to find a new set of orthogonal axes which are also known as principal components, that capture the maximum variance in the data. Higher eigenvalues indicate that the corresponding eigen vectors capture more variations in the data. The first principal component is associated with the highest eigen value and represents the most significant direction of variation present in the data. Other principal components capture the remaining variations. These components are chosen to be orthogonal to each other where each provides unique information about the data's pattern.

The covariance matrix is a square matrix that provides a measure of the covariance between pairs of variables in a dataset. It is used to quantify the relationships and interactions between variables.Covariance matrix provides information about relationship between different attributes. Diagonal terms of covariance matrix represent redundancy while off-diagonal terms represent variance. So, our goal is to minimize off diagonal and maximize diagonal values.

The eigen values can also be used to calculate the proportion of variance (variance ratio) explained by a certain number of principal components.It simply provides insights on amount of information retained by each principal component in Principal Component Analysis (PCA). Components with high variance proportions can be regarded as more important features in the dataset. Proportion of Variance helps to determine the optimal number of principal components that maintain a meaningful amount of variance while reducing the dataset's dimensionality. To obtain this proportion, each eigenvalue is divided by the sum of all eigenvalues.

B Mathematical Formulae

Let's assume we have a dataset 'X' with m records and n attributes i.e. having dimension $m \times n$.
For each attribute j, the mean can be calculated as:

$$\mu = \frac{\sum_{i=1}^{m} x_{ij}}{m} \tag{1}$$

Mean matrix ($\boldsymbol{\mu}$):

$$\boldsymbol{\mu} = (\mu_1, \mu_2, \mu_3, \ldots, \mu_n) \tag{2}$$

where
μ_1 is the mean of the first attribute in the dataset,
μ_2 is the mean of the second attribute in the dataset,
μ_3 is the mean of the third attribute in the dataset,
and so on, up to μ_n which is the mean of the nth attribute in the dataset.
To obtain the zero mean normalized dataset, we subtract the mean matrix $\boldsymbol{\mu}$ from each corresponding record $\mathbf{x}$ in the dataset:

$$\text{x\_normalized} = \mathbf{x} - \boldsymbol{\mu} \tag{3}$$

The covariance matrix for given dataset 'X' is calculated by:

$$\mathbf{S_X} = \frac{1}{m-1} \cdot (X^T \cdot X) \tag{4}$$

For a square matrix A, eigenvectors and eigenvalues satisfy the following equation:

$$A \cdot \mathbf{v} = \lambda \mathbf{v} \tag{5}$$

where $\mathbf{v}$ is an eigenvector and λ is the corresponding eigenvalue.
The above equation can be rearranged to form the given characteristics equation which can be used to find the values of λ:

$$(A - \lambda_i I) = 0 \tag{6}$$

where I is the identity matrix of the same size as A.
After calculating values of λ_i, we can calculate corresponding eigen vector using the formula below:

$$(A - \lambda_i I)v_i = 0 \tag{7}$$

The proportion of variance(PV) for each eigenvalue λ_i is calculated by:

$$\text{PV}_i = \frac{\lambda_i}{\sum_{j=1}^{n} \lambda_j} \tag{8}$$

Where $\sum_{j=1}^{n} \lambda_j$ represents the sum of all eigenvalues.

C System Block Diagram

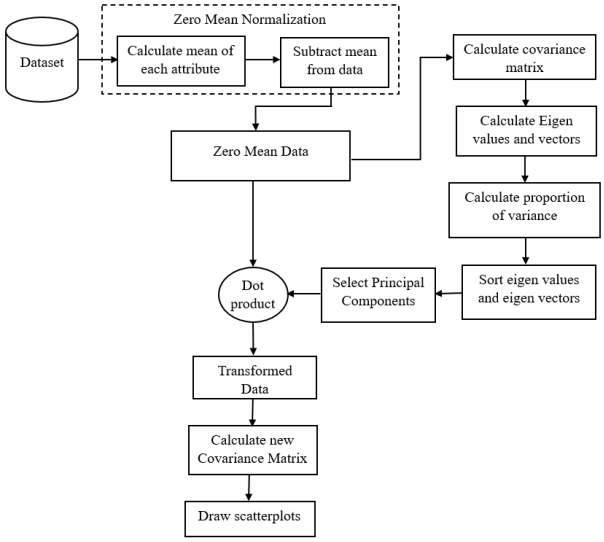

Figure 1: Block Diagram for Principal Component Analysis

D Working Principle

Principal Component Analysis (PCA) is a widely used dimensionality reduction technique. In our implementation using the Python programming language, we first carry out zero-mean normalization on the dataset. This involves calculating the mean of each attribute and subtracting it from the data matrix. By centering the data around a zero mean, we remove any bias present in the data. The normalized data is stored in a variable called 'X'.

Next, we calculate the covariance matrix of the normalized data. The covariance matrix captures the relationships and interactions between variables in the dataset. We also compute the eigenvalues and eigenvectors of the covariance matrix using the standard functions available in the NumPy library. The eigenvalues represent the amount of variance explained by each principal component, while the eigenvectors indicate the directions in the feature space that correspond to these components.

After sorting the eigenvalues in descending order, we select different permutations of eigenvalues and perform a change of basis. This involves taking the dot product between the transpose of the selected eigenvectors and the transpose of the zero-mean normalized data matrix 'X'. This process results in a new transformed dataset with reduced dimensionality. The dimension of the transformed data depends on the number of selected eigenvectors.

Finally, we calculate the covariance matrix of the transformed data and plot scatter graphs to visualize the data in 1D, 2D and 3D. These plots help us analyze the behavior and performance of PCA in reducing the dimensionality of the dataset.

E Instrumentation Details

In our study, we have used Python programming language for implementing PCA. We have implemented code in Jupyter Notebook, which is a web-based, interactive computing notebook environment. We have imported different libraries required for our experiment. First, the library we have imported is NumPy [3] , which is a powerful numerical computing library. By using this library, we have calculated the covariance matrix using the 'cov()' function, computed eigenvalues and eigenvectors using the 'linalg.eig()' function, and performed dot product calculations using the 'dot()' function. Additionally, NumPy is used for calculating sums using the 'sum()' function, generating random values using the 'random()' function, and sorting values using the 'argsort()' function.

The second library used is `sklearn.datasets` [4]. This library offers various datasets for machine learning and data analysis tasks. We have used this library to load the iris and wine datasets using the `load_iris()` and `load_wine()` functions, respectively.

The 'pandas' library is used to create data frames, which help us visualize our data more clearly.

Finally, we have used the 'matplotlib' library in our study for generating 1D, 2D and 3D scatter plots. We have utilized the 'scatter()' function of this library, which takes the values of individual columns and a class label to produce visually appealing plots. Other functions like 'xlabel()', 'ylabel()', 'title()', etc., are used to enhance the appearance of our graphs.

F Dataset Description

In our study of Principal Component Analysis (PCA), we have analyzed three datasets: a randomly generated dataset, the Iris dataset, and the Wine dataset. The first dataset is created by using the random function from the NumPy library. In this process, we first create a 20x2 matrix with normally distributed random values. We then multiply this matrix by a 2x2 matrix with uniformly distributed random values ranging between 0 and 1.The second dataset, the Iris dataset, is downloaded from the scikit-learn [4] library. It consists of 150 records with 4 attributes: sepal length, sepal width, petal length, and petal width. The 150 records in this dataset are divided into 3 classes: Iris setosa, Iris versicolor, and Iris virginica. The third dataset, the Wine dataset, is also downloaded from the scikit-learn library. It consists of 178 records with 13 attributes. These 178 records are divided into 3 classes. Both the Iris and Wine datasets are real-world datasets that are commonly used to examine the effectiveness of PCA in real-world scenarios. Overall, our study includes

a randomly generated dataset and two real-world datasets (Wine and Iris) to explore the application and effectiveness of PCA.

III EXPERIMENTAL RESULTS

A Problem 1: PCA on Random Data

We apply PCA on random dataset to analyze the behaviour and effectiveness of PCA on dimensional reduction on synthesized data.Through various scatter plots we analyzed the data's dimensionality reduction and variance explained by the principal components. We take both the principal components one by one to visualize data when reduced to 1D and both principal components to visualize data when reduced to 2D.

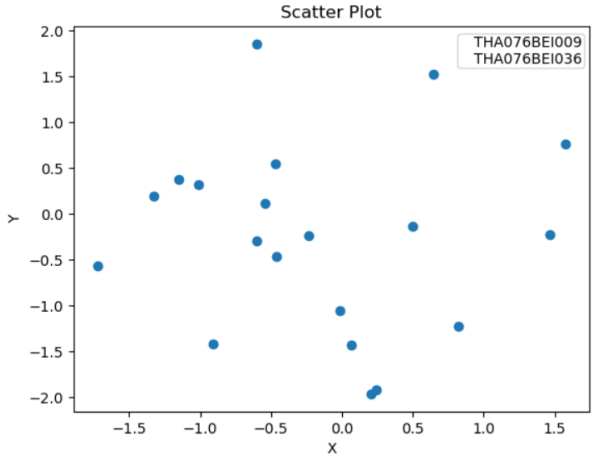

Figure 2: Plot of Normal(Gaussian) data

Figure 2 is simply a scattterplot of randomly generated normally distributed sample of size 20*2.

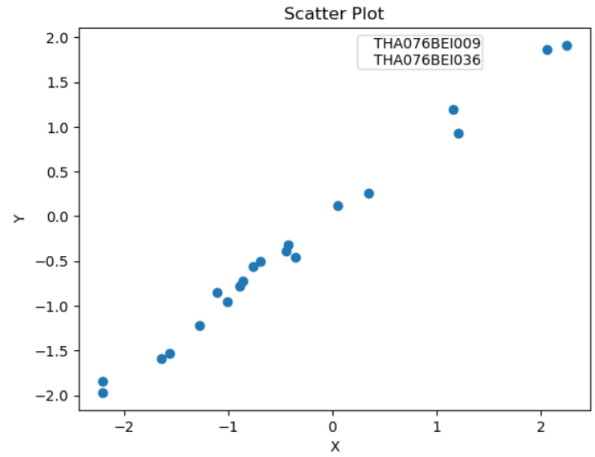

Figure 3: Plot of Resultant data

Above plot(Figure 3) is the scatterplot of the data generated by multiplying the randomly generated dataset and 2*2 matrix filled with random values from a uniform distribution having values between 0 and 1.This multiplication step allows us to create a dataset with specific patterns and relationships among the variables, which can help illustrate the behavior of PCA in different scenarios.

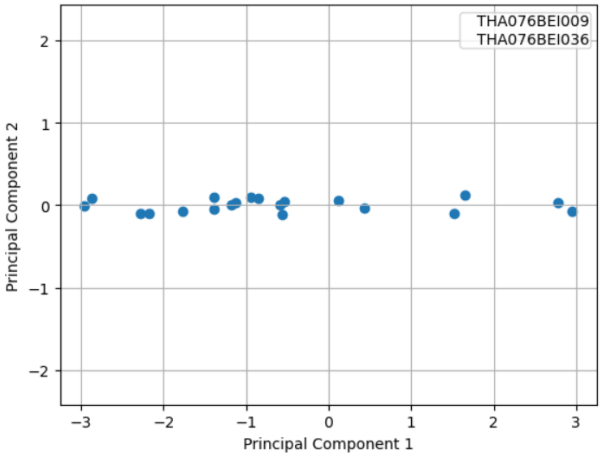

Figure 4: Plot of data by taking both eigen values

Above figure(i.e. Figure 4) is simply the rotation of the data since we take both the eigen values and here PCA only changes the basis but doesn't reduce the dimensionality.

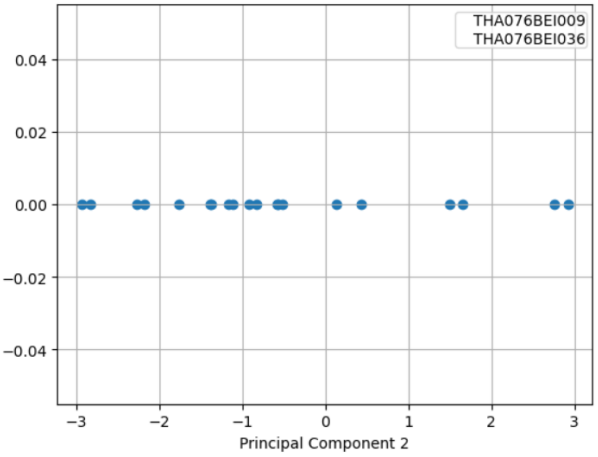

Figure 5: Plot using PC1

In figure 5, the 2D data is now transformed into 1D data where we take the only one most important principal component.

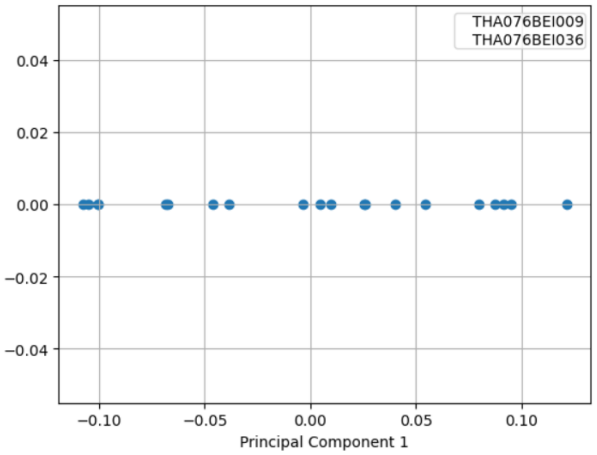

Figure 6: Plot using PC2

In figure 6, the 2D data is now transformed into 1D data where we take the only one lesser important principal component

B Problem 2: PCA on Iris Dataset

We apply PCA on iris dataset to analyze the behaviour and effectiveness of PCA on dimensionality reduction on real-world data.Through various 1D,2D and 3D scatterplots, we analyzed the data's dimensional reduction and variance explained by the principal components. Iris dataset consists of 4 attributes. We take various combinations of principal components to visualize dimensionality reduction in Iris dataset.

1 1D Plots of Iris Dataset

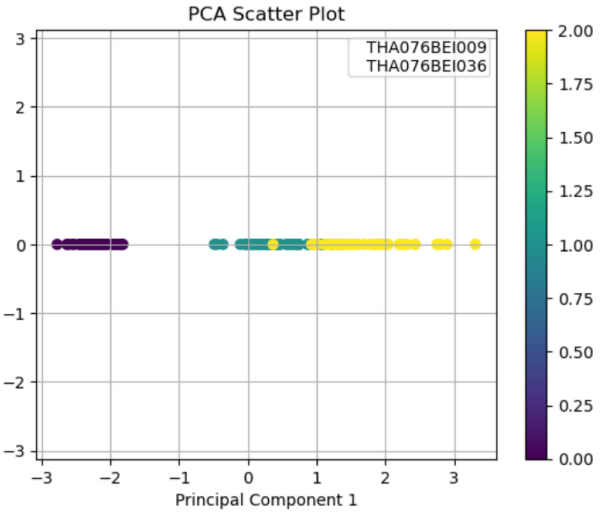

Figure 7: Plot using PC1

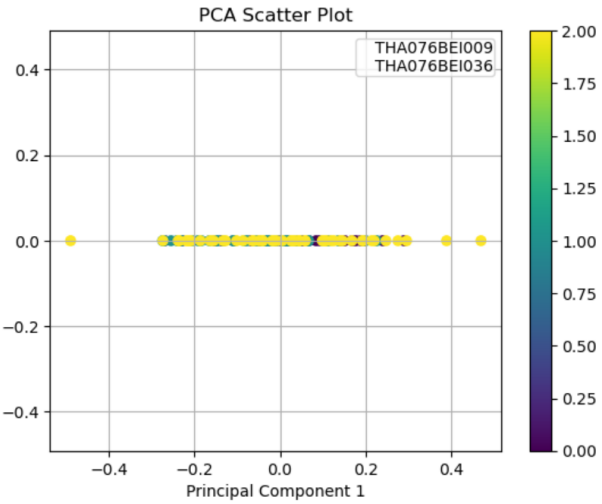

Figure 8: Plot using PC4

In figure 7 and Figure 8, we have used only one principal component to draw scatterplot of the data. For figure 7, we have taken the first principal component corresponding to the highest eigen value to draw 1D scatterplot and for figure 8, we have taken the last principal component corresponding to least eigen value to draw a 1D scatterplot. In the two figures, we can see that the classes in figure 7 are separable than classes in figure 8. This provides the insight that if we reduce the dimensionality of the dataset to 1D by taking the first(most important) principal component, then the new data's class can be determined more easily than in the case where we reduce the dimensionality of the dataset to 1D by taking the lesser important principal component.

2 2D Plots of Iris Dataset

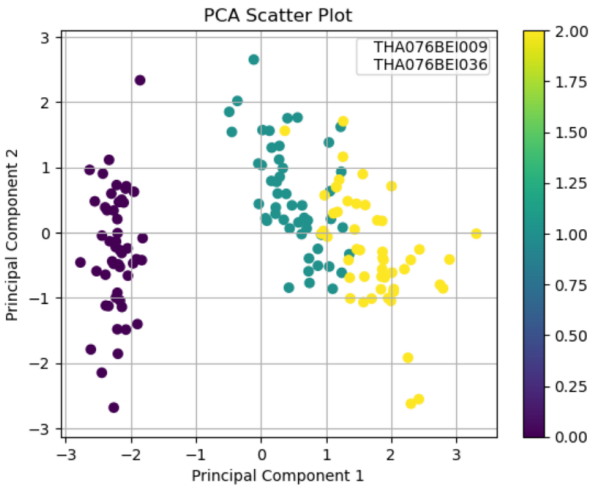

Figure 9: Plot using PC1 and PC2

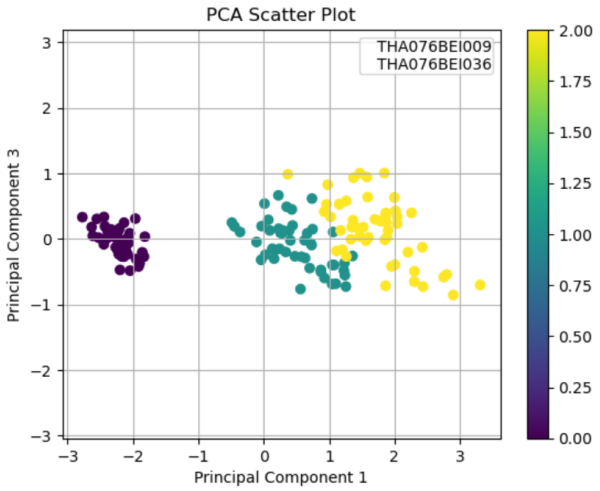

Figure 10: Plot using PC1 and PC3

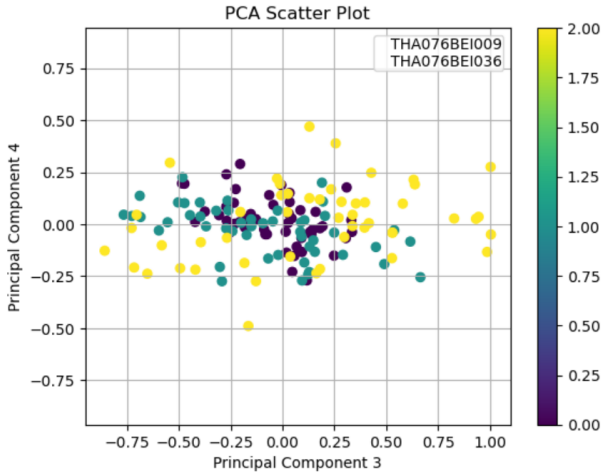

Figure 11: Plot using PC3 and PC4

Figure 9,10 and 11 are the scatterplots of the data when reduced to 2-dimension. Figure 9 is the scatterplot of the data when we take the highest two principal components i.e. PC1 and PC2. Similarly in figure 10, we have taken PC1 and PC3 and in figure 11 we have taken the least important principal components i.e. PC3 and PC4. When comparing all the three, we can see that the separability of classes is high in figure 9 and 10 than that of figure 11. This verifies that dimensionality reduction is more efficient for class separability when we take more important principal components than taking lesser important principal components. If we compare figure 9 and figure 10, Figure 10, where classes appear to be more separable compared to Figure 9, indicates that there may be other principal components beyond the topmost ones that better capture the discriminatory information in the data.So we can say that the combination of the topmost principal components is not always the best for separability.

3 3D Plots of Iris Dataset

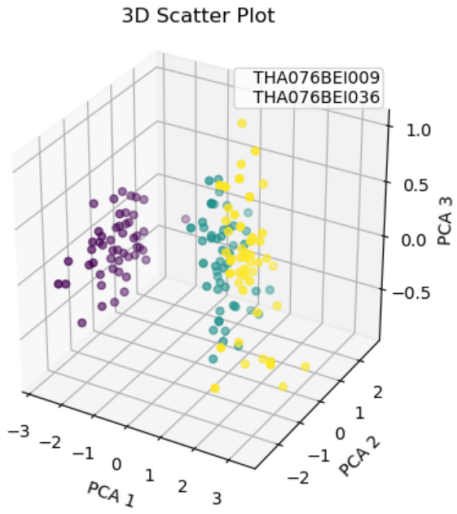

Figure 12: 3D Plot using PC1, PC2 and PC3

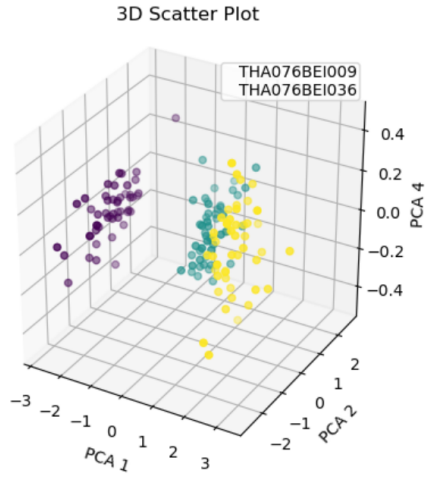

Figure 13: 3D Plot using PC1, PC2 and PC4

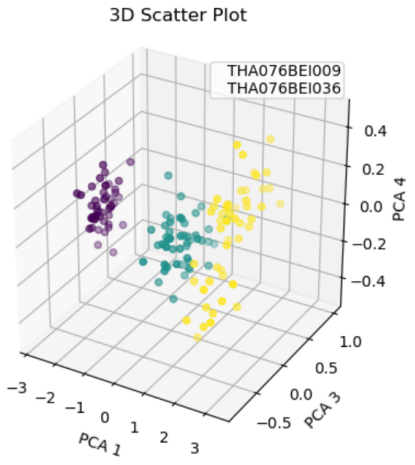

Figure 14: 3D Plot using PC1 ,PC3 and PC4

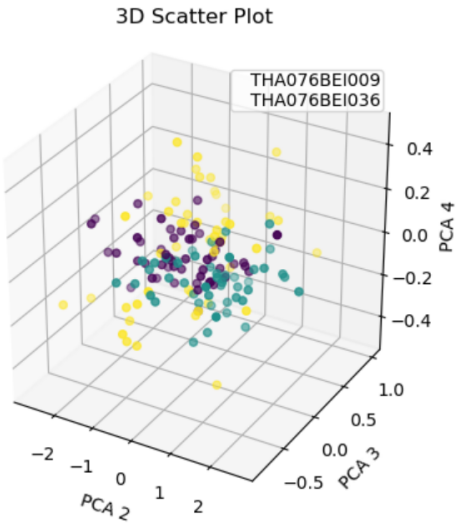

Figure 15: 3D Plot using PC2, PC3 and PC4

Figure 12, 13, 14 and 15 are the scatterplots of the data when the dimension is reduced to 3. In general, we can see that the separability of class is least for the last figure which is generated by taking the least important three principal components. In figure 12, 13 and 14 we can see that the three classes are more separable in figure 14 which is the plot when we take principal components PC1, PC3 and PC4. This observation highlights the limitations of relying solely on the top-ranked principal components for separability. Eventhough the top-ranked components captures the most variance in the dataset, they may not always correspond directly to the most distinguishing features or class boundaries. Other principal components, even if they have lower associated eigenvalues, may still capture important discriminative information for better class separation .

If we look at the figure 12, 13 and 14, we can also notice that the class which is denoted by purple color is highly separable in figure 12 and its separability is decreasing gradually in figure 13 and 14. So if one class is highly separable when considering the top three principal components, while the other classes may not exhibit the same level of separability, it indicates that these components capture class-specific discriminative information. This insight suggests the need for deeper analysis to improve the separability of the classes.

C Problem 3: PCA on Wine Dataset

Wine dataset consists of 13 attributes. We are able visualize the plots only in 1D, 2D and 3D so, we reduce the dimensionality from 13D to 1D, 2D and 3D. As there is huge drop in dimensional, a lot of information might lost during PCA.

1 1D Plots of Wine Dataset

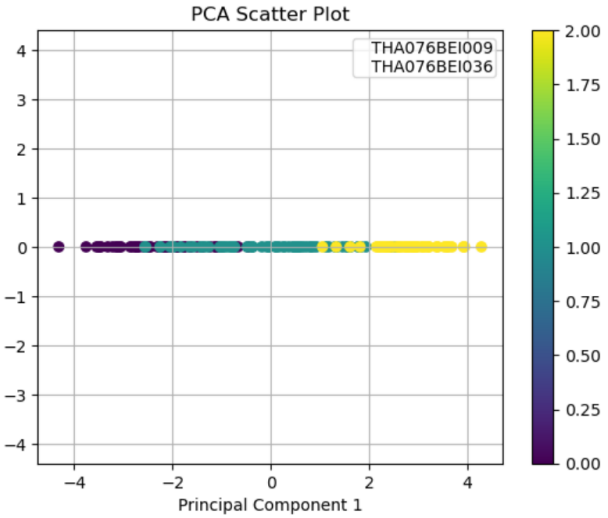

Figure 16: Plot using PC1

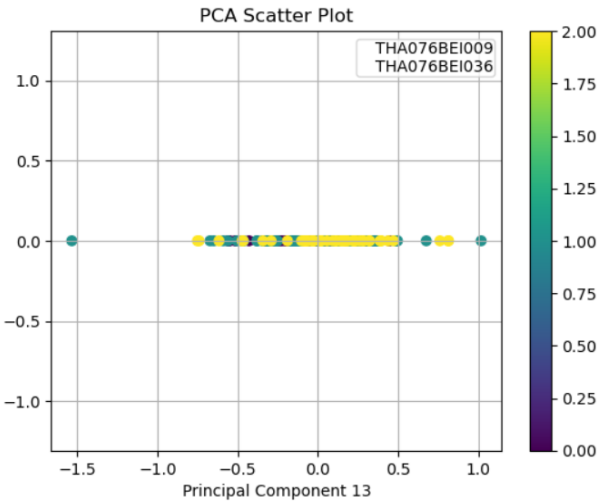

Figure 17: Plot using PC1 and PC3

In Figure 16 and Figure 17, the scatterplots for the Wine dataset show the results when reducing the data to 1-D using PCA. However, due to the significant reduction in dimensionality, the separability of the classes is not well preserved. Even the most important principal component does not sufficiently separate the classes. This highlights the importance of considering the loss of information when reducing dimensions. But also comparatively, the separatibility in figure 16 is comparatively higher than in figure 17 which indicates the significance of top-most principal component in PCA.

2 2D Plots of Wine Dataset

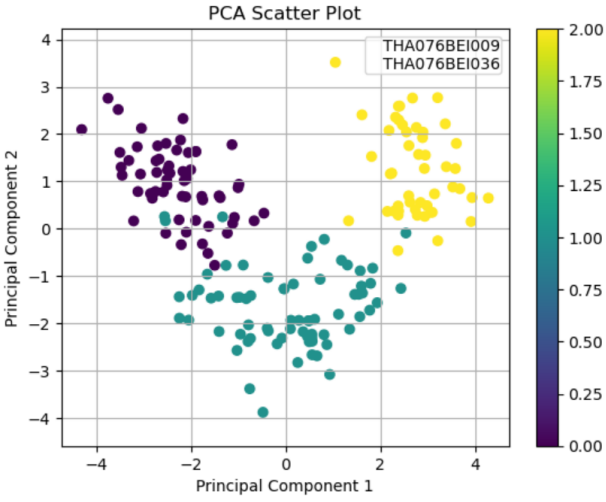

Figure 18: Plot using PC1 and PC2

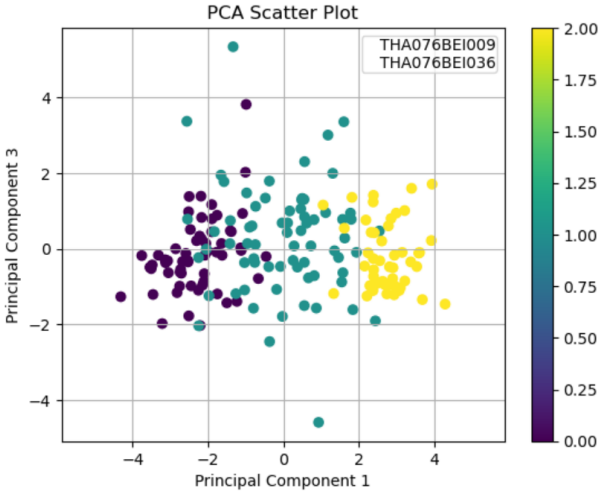

Figure 19: Plot using PC1 and PC3

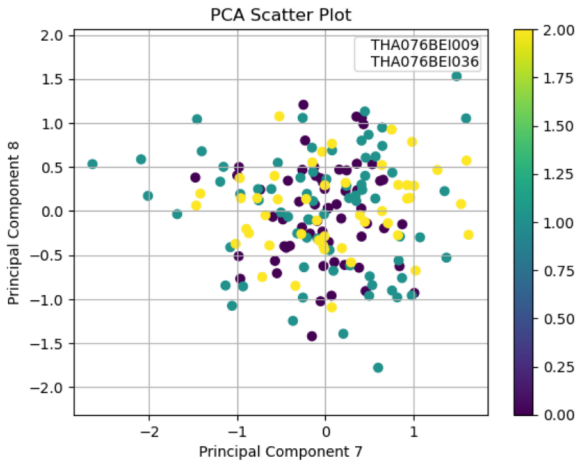

Figure 20: Plot using PC7 and PC8

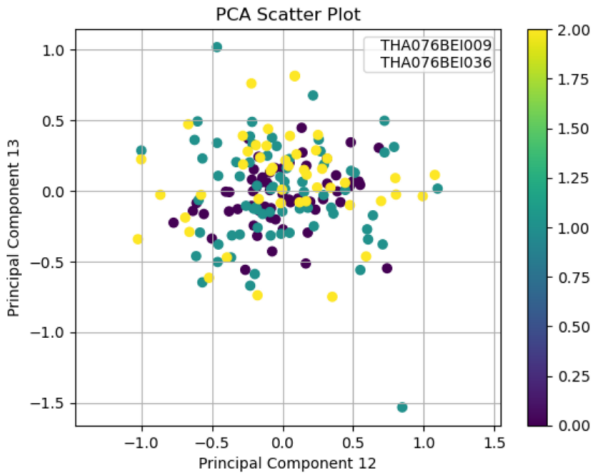

Figure 21: Plot using PC12 and PC13

Above 2D plots are generated for different combinations of two principal component. In figure 18, even though the dimension is reduced to large extent, there is significant separability when we take the top-most two principal components. This indicates that these two components capture significant information that enables effective class separation. We can also notice the decrease in separability for the other combinations of principal components and can visualize the worst separability when the lowest two components are taken i.e. in figure 21.

3 3D Plots of Wine Dataset

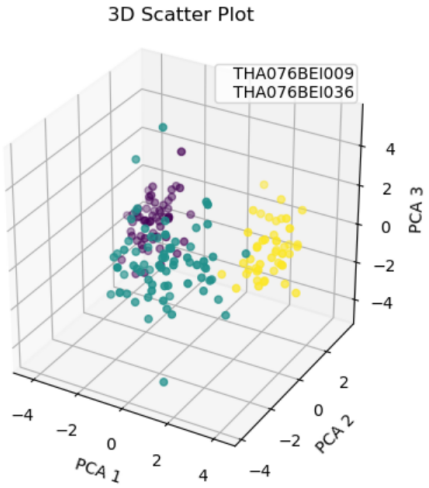

Figure 22: 3D Plot using PC1 , PC2 and PC3

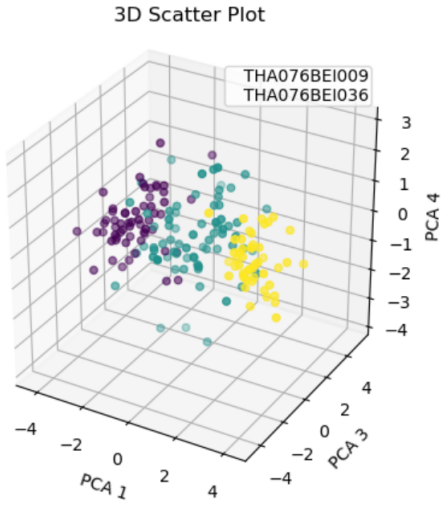

Figure 23: 3D Plot using PC1, PC3 and PC4

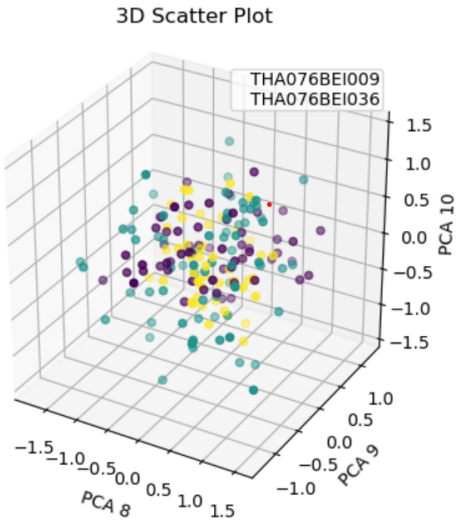

Figure 24: 3D Plot using PC8, PC9 and PC10

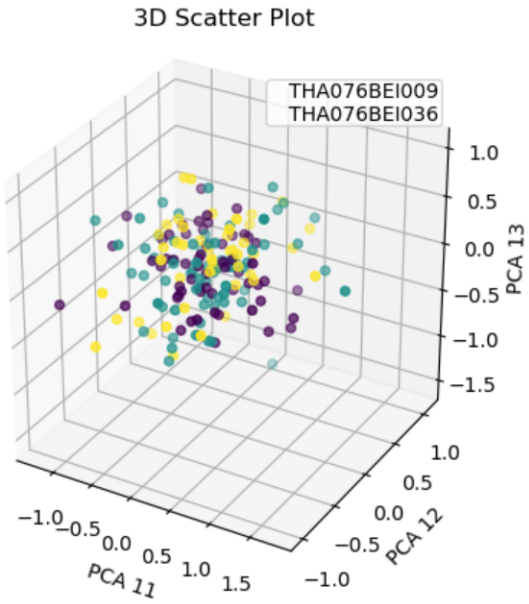

Figure 25: 3D Plot using PC11, PC12 and PC13

The most common property of decreased separability when considering other principal components instead of just the top-most ones can be visualized in 3D plots of wine dataset.

If we compare the 2-D plot drawn by taking the top-most two and 3-D plot drawn by taking the top-most three components i.e. Figure 18 and Figure 22, we can visualize that the classes are less distinguishable when considering the three principal components compared to the top two. This signifies that in some cases,including the third principal component introduces additional information that may not contribute significantly to the separability of the classes. As a result, the separability decreases when considering the three topmost principal components compared to just the top two. This suggests that the third principal component may contain noise or less relevant information for class separation in this particular dataset. This emphasizes the importance of selecting the most informative components for achieving better separability in higher-dimensional datasets.

IV Discussion and Analysis

In the previous sections, we witnessed the successful application of Principal Component Analysis (PCA) for dimensionality reduction. Various graphs were plotted to demonstrate the effectiveness of PCA on the given dataset. By applying PCA, we obtained a transformed matrix with a lower dimension. During the computation of the covariance of the transformed matrix, we noticed a reduction in off-diagonal elements and an increase in diagonal elements. This outcome aligns with our goal of minimizing redundancy and maximizing variance between attributes.

When examining graphs of different principal components, we observed that classes are more easily distinguishable when plotted against principal components with higher eigenvalues. Conversely, it becomes more challenging to distinguish classes when using principal components with lower eigenvalues. This indicates that principal components with higher eigenvalues play a more significant role in defining the features of the dataset. Therefore, it is advisable to prioritize principal components with higher eigenvalues when performing PCA. But, we also encountered some cases where some other combination of principal components rather than the top-most ones gives higher class separability. So we should explore different combinations of pricipal components and evaluate their impact on class separability.

We also found that in some cases taking only two principal components have higher class separability than taking three top-most principal components.While the three top-most principal components may capture additional variations in the data, they might not necessarily align with the most significant discriminative features or class boundaries. As a result, the additional principal component may introduce noise or less relevant information, leading to lower class separability. In such cases, it is important to consider the specific characteristics of the dataset and carefully select the number and combination of principal components that optimize the separability of the classes.

While the theory of Principal Component Analysis (PCA) suggests that the transformed covariance matrix should ideally be diagonal, in simulation, due to computational limitations and numerical approximations, the off-diagonal elements are not found to be exactly zero but rather very close to zero.Non-zero values in the off-diagonal elements of the transformed covariance matrix in PCA are typically caused by rounding errors or small numerical discrepancies during the computation process. These errors can accumulate throughout the different steps of PCA, including zero-mean normalization, eigenvalue decomposition, and data transformation. These small non-zero values in the off-diagonal elements have negligible impact on the overall results and interpretation of PCA.

As previously mentioned, selecting the appropriate principal components enables effective data analysis in lower dimensions, equivalent to that in higher dimensions. This observation emphasizes that valuable information is re-

tained in the transformed matrix through the use of PCA. Overall, PCA provides a powerful technique for reducing dimensionality while preserving important information, as evidenced by the graphs and results presented.

V Conclusion

In this lab, we successfully performed PCA on a random dataset, Iris dataset and Wine dataset. By using various libraries of python and by implementing various steps of PCA, we successfully reduced the dimension of the datasets and analyzed the effectiveness of PCA. The previous sections of the analysis yield several key conclusions. First and foremost, Principal Component Analysis (PCA) has proven its prowess in feature extraction and dimensionality reduction across all three datasets. In iris and wine dataset PCA was able to reduce dimension of original data while preserving valuable information. Evidence of effectiveness of PCA can be seen in above graph where lower dimension data has also been able to distinguish its class. For better performance of PCA, principal components with higher eigen values need to be selected. Overall, PCA is a powerful technique for data analysis and can significantly contribute to improved understanding and interpretation of complex datasets.

REFERENCES

[1] M. Köppen, "The curse of dimensionality," in *5th online world conference on soft computing in industrial applications (WSC5)*, vol. 1, 2000, pp. 4–8.

[2] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[3] T. E. Oliphant *et al.*, *A guide to NumPy*. Trelgol Publishing USA, 2006, vol. 1.

[4] O. Kramer and O. Kramer, "Scikit-learn," *Machine learning for evolution strategies*, pp. 45–53, 2016.

Bishwambhar Dahal is a fourth-year student of Electronics, Communication, and Information Engineering. With a deep fascination for Artificial Intelligence (AI), He is driven by the potential of AI to transform industries and tackle complex challenges. His academic journey has equipped him with a strong foundation in AI concepts, including machine learning and data analysis. He possess a relentless curiosity and is always eager to explore the latest advancements in AI. His goal is to apply his knowledge and skills in AI to make meaningful contributions to research and development, pushing the boundaries of what is possible with intelligent algorithms.(THA076BEI009)

Sirjana Bhatta is a fourth-year student of Electronics, Communication, and Information Engineering with a keen interest in the field of Artificial Intelligence (AI). She possess a strong academic foundation and practical skills in machine learning, deep learning, and data analysis. Her passion lies in leveraging AI to revolutionize industries and solve complex problems. She is a motivated learner, constantly staying updated with the latest advancements in AI. She is seeking opportunities to contribute to AI research and development and make a positive impact on society. (THA076BEI036)