GE 461 Introduction to Data Science
Project W9: Dimensionality Reduction and Visualization
İlknur Baş
21601847

LIBRARIES

The libraries that I have used in this project (some of them are also mentioned in the .py file with comments):

- **scipy.io** is used to read .mat files [1].
- **NumPy** is used when working with arrays [2].
- **train_test_split** from **sklearn.model_selection** library is used in order to split the given digit data into two subsets as train and test randomly [3].
- **PCA** from **sklearn.decomposition** [4] library is used in order to do principal component analysis. Its methods and attributes that are used in this project is listed as follows: fit (X, [, Y]), transform(X), mean_, explained_variance
- **matplotlib** is used to plot the desired things written in the assignment [5].
- **LinearDiscriminantAnalysis** from **sklearn.discriminant_analysis** [6] library is used to perform LDA. Its methods and attributes that are used in this project is listed as follows: fit(X,y), coef_
- **GaussianNB** from **sklearn.naive_bayes** [7] library is used to perform Gaussian Naive Bayes.

QUESTION 1

1)

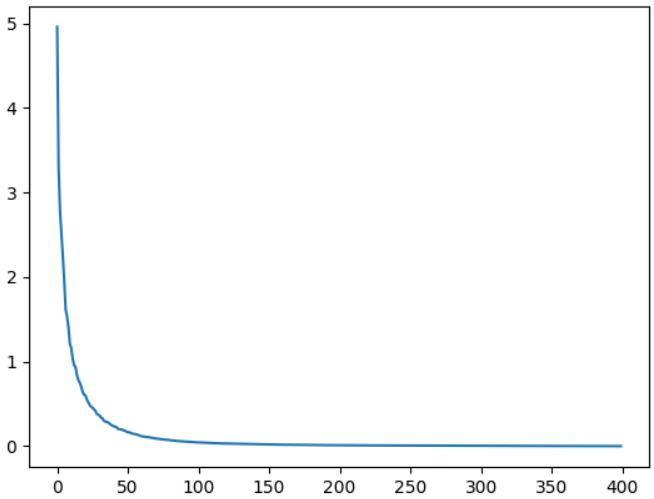

Figure 1: Eigenvalues in descending order

GE 461 Introduction to Data Science
Project W9: Dimensionality Reduction and Visualization
İlknur Baş
21601847

There were 400 components as it is mentioned in the assignment, and the plot shows the eigenvalues with the help of *explained_variance* component (see the library from LIBRARIES section). The highest eigenvalue demonstrates the highest variance in the digit data. Between 40 and 50 components (since the intervals are not very precise, I cannot write the exact number of components), the line starts to be close to 0. Selecting components larger than around 60-70 won't do much since their variance is decreasing as it goes. Hence, I would select 50 components.

2)

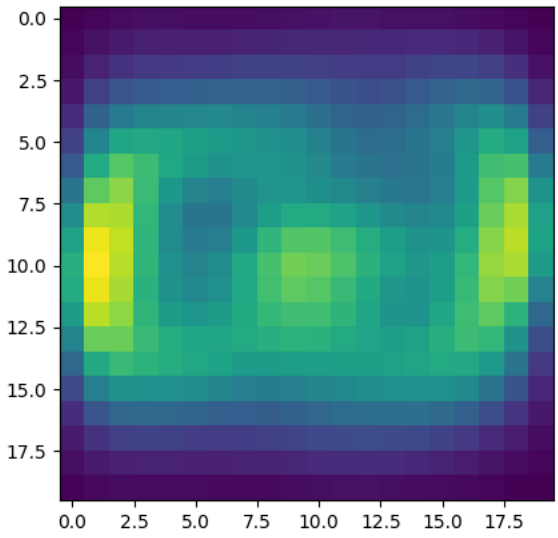

Figure 2: Sample mean as an image

We can say that the mean of the sample shows us like the average of the data. We can identify the mean as it is similar to the digit 8 from Figure 2. Since it is not very clear, we can also say that it could be similar to digit 3 or 6. The similarity can also be seen from the given digits.mat file. Also, since the mean of the whole training data is like that, I expect eigenvectors that I selected to be similar to this image since I selected highest eigenvalues.

GE 461 Introduction to Data Science
Project W9: Dimensionality Reduction and Visualization
İlknur Baş
21601847

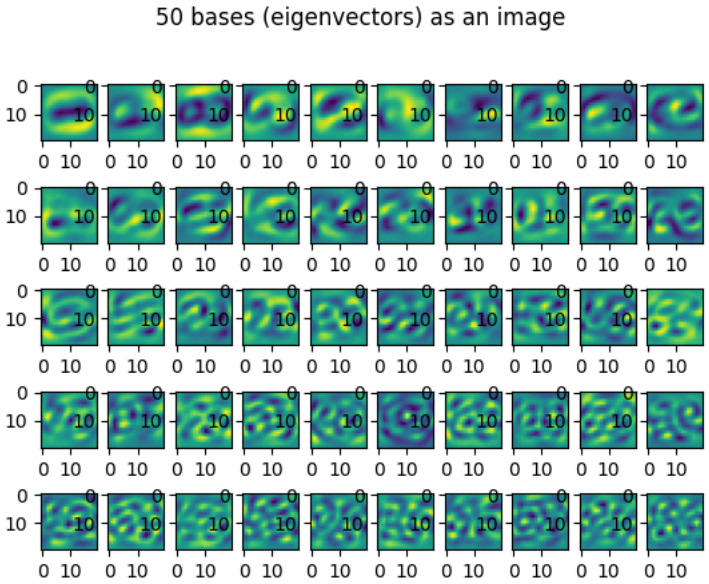

Figure 3: 50 eigenvectors as an image

Since, I choose 50 components in the Part 1, I performed PCA with 50 components to display eigenvectors. I think, some eigenvectors do not tell very much. But in the majority, we can see the shapes like digit 8. In fact, in some of them, such as in last column 4th row, we can encounter with a digit like 6. I wrote that sample mean is similar to digits 8,6 or 3. Having the digits similar to 8,6 or 3 (meaning similar to mean) in the Figure 3 shows us that first 50 components indeed have higher variance so that their image happened to be similar to mean.

3-4)

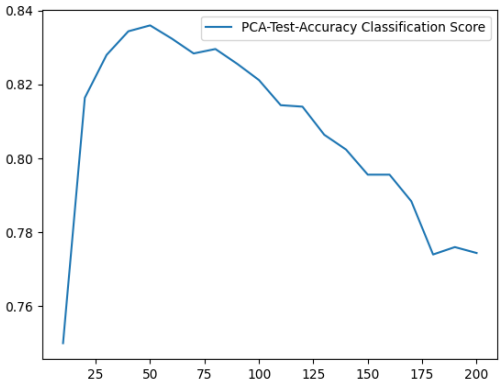

Figure 4: Test Classification Accuracy

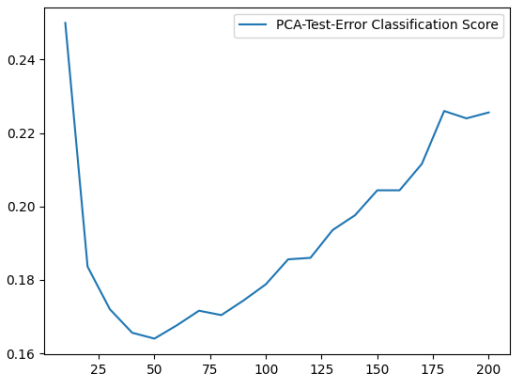

Figure 5: Test Classification Error

GE 461 Introduction to Data Science
Project W9: Dimensionality Reduction and Visualization
İlknur Baş
21601847

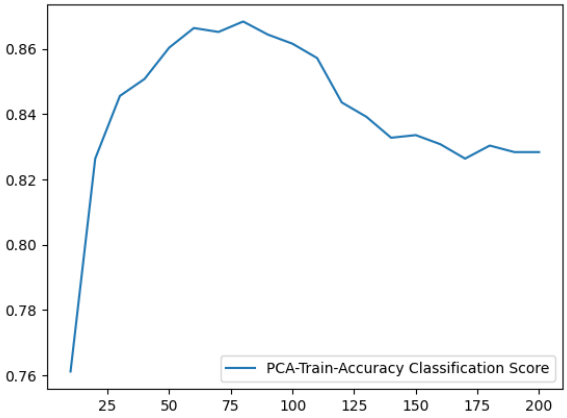

Figure 6: Train Classification Accuracy

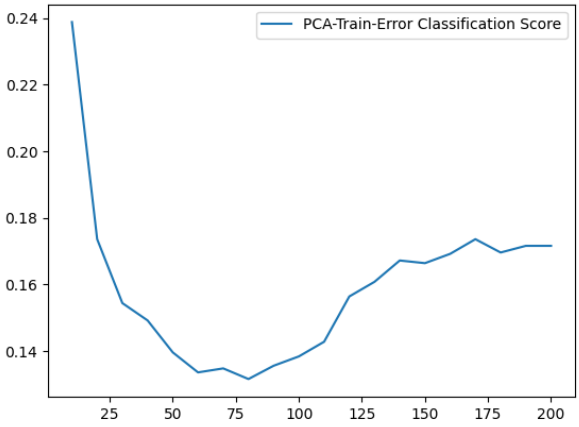

Figure 7: Train Classification Error

Values of accuracy and error for both test and train data can be seen from the output console of the .py file.

In Figure 4, the plot achieved is highest accuracy rate when there are 50-55 components. I expected to be in this way (I explained in Part 1-2), and also the plot fulfilled my expectations. This is also true for test data. It reaches its highest rate when there are 50-55 components.

QUESTION 2

1)

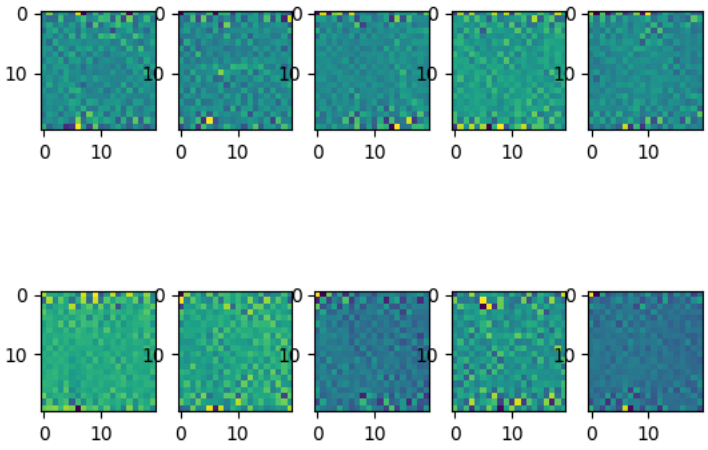

Figure 8: Set of bases

The set of bases can be seen from Figure 8, I have obtained all 10 of them since there can be at most 10. I think, the colors are not very distinct in general. We can say that the color of 2nd row 1st column is lighter compared to others and the color of 2nd row last column is darker compared to others. But in general, I expected to see a little more difference in color wise between the bases so that the digits can be identified more clearly. We can see more distinction especially in the 2nd row.

2-3)

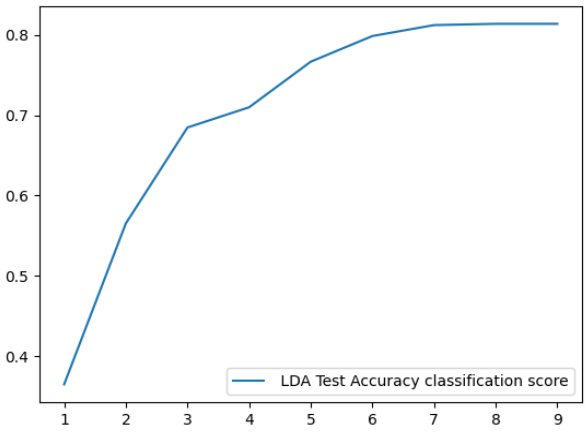

Figure 9: Test Classification Accuracy

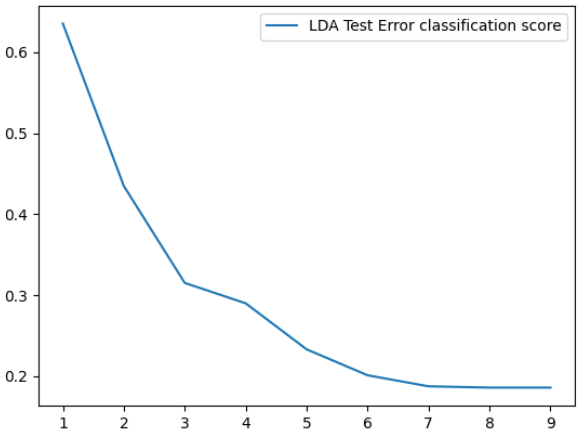

Figure 10: Test Classification Error

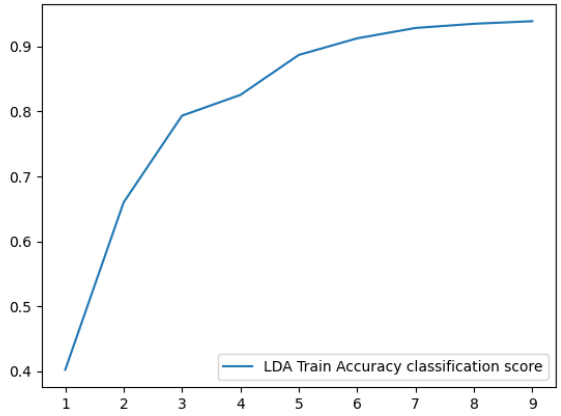

Figure 11: Train Classification Accuracy

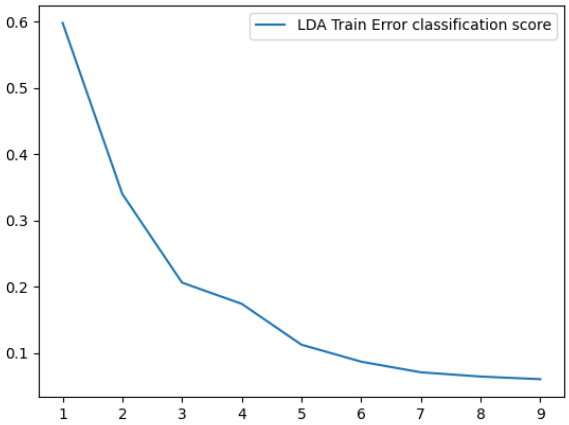

Figure 12: Train Classification Error

Values of accuracy and error for both test and train data can be seen from the output console of the .py file.

GE 461 Introduction to Data Science
Project W9: Dimensionality Reduction and Visualization
İlknur Baş
21601847

As the dimension number reaches to around 3-4, the accuracy rate is in rapid increase. After that there is a slow increase but still there is an increase. In my opinion, that means we can identify the digit data accurately (with little error) with 5 or 6 dimensions, no need for all of them.

REFERENCES

[1]"File IO (scipy.io) — SciPy v1.6.2 Reference Guide", *Docs.scipy.org*, 2021. [Online]. Available: https://docs.scipy.org/doc/scipy/reference/tutorial/io.html. [Accessed: 16- Apr- 2021].

[2] “numpy.append — NumPy v1.20 Manual", *Numpy.org*, 2021. [Online]. Available: https://numpy.org/doc/stable/reference/generated/numpy.append.html. [Accessed: 16- Apr- 2021].

[3] “sklearn.model_selection.train_test_split — scikit-learn 0.24.1 documentation", *Scikit-learn.org*, 2021. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html. [Accessed: 16- Apr- 2021].

[4] "sklearn.decomposition.PCA — scikit-learn 0.24.1 documentation", *Scikit-learn.org*, 2021. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html#sklearn.decomposition.PCA.transform. [Accessed: 16- Apr- 2021].

[5] “Pyplot tutorial — Matplotlib 3.4.1 documentation", *Matplotlib.org*, 2021. [Online]. Available: https://matplotlib.org/stable/tutorials/introductory/pyplot.html. [Accessed: 16- Apr- 2021].

[6] "sklearn.discriminant_analysis.LinearDiscriminantAnalysis — scikit-learn 0.24.1 documentation", *Scikit-learn.org*, 2021. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.discriminant_analysis.LinearDiscriminantAnalysis.html. [Accessed: 16- Apr- 2021].

[7] "sklearn.naive_bayes.GaussianNB — scikit-learn 0.24.1 documentation", *Scikit-learn.org*, 2021. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html. [Accessed: 16- Apr- 2021].