Audio Sampling and Classification with Singular Value Decomposition & Linear Discriminant Analysis

Joe Henthorn
University of Washington

Winter 2020

Abstract: The goal of this paper is to demonstrate how to use the singular value decomposition (SVD) and the Linear discrimination analysis (LDA) methods to obtain the Principle components of a system and to use them to classify audio data with spectrograms based on genres and artists.

1 Introduction

This document is a guide for Proper Orthogonal Decomposition (PCA), and the Linear discrimination Analysis (LDA). The examples in this paper use a few functions to acquire five second audio clips from songs of different artists and genres. We first use the Fast Fourier transform **(fft)** to generate spectrograms of the audio clips and then we train a classifier with these spectrograms by using the SVD method to determine the Principle components of each artist or genre. This produces a line of best fit for which we can use to project new testing data onto. Lastly, we run trials to test our classifier and determine it's accuracy.

2 Theoretical Background

The core of this method is to use Principle components of a data set to build a classifier that can guess music artists and genres. In order to achieve this we must use the LDA method to project our test data onto our trained model. In particular, we need a projection of best fit such that it maximizes the distance between the inner-class data, while also minimizing the between-class data. This projection $\vec{\boldsymbol{w}}$ is the eigenvector of the largest eigenvalue, whose projection is of interests as the best fit for this LDA. It is also worth normalizing this vector by its two norm.

$$\vec{\boldsymbol{w}} = argmax_{\boldsymbol{w}} \frac{\boldsymbol{w}^{\intercal} S_B \boldsymbol{w}}{\boldsymbol{w}^{\intercal} S_W \boldsymbol{w}} \tag{EQ:1}$$

This is also known as the Reyleigh Quotient. For a two class case we have the following scatter matrices:

$$S_B = (\mu_2 - \mu_1)(\mu_2 - \mu_1)^T \tag{EQ:2}$$

and

$$S_W = \sum_{j=1}^{2} \sum_{x} (x - \mu_j)(x - \mu_j)^T \tag{EQ:3}$$

$\vec{\boldsymbol{w}}$ can be found by the generalized eigenvalue problem below, Where w is the eigenvector of the largest eigenvalue, whose projection is the best fit line for this LDA.

$$S_B\boldsymbol{w} = \lambda S_W\boldsymbol{w} \tag{EQ:4}$$

It is also worth noting that our projection onto the PCA components can be calculated from the SVD by:

$$proj(PCA) = \Sigma V^T \tag{EQ:5}$$

Where U and V are unitary matrices and Σ is a diagonal matrix. The diagonal values of Σ are the singular values σ_j ordered from largest to smallest.

Here, the U matrix represents the rotation matrix with eigenvectors as columns. For our classification application, we take the first 1-N values, where N is the number of features we retain. To evaluate our test-set data (T), we use this truncated U matrix to project onto.

$$proj(SVD) = UT \tag{EQ:6}$$

Furthermore, we can find the LDA projection by

$$proj(LDA) = \boldsymbol{w} \cdot proj(SVD) \tag{EQ:7}$$

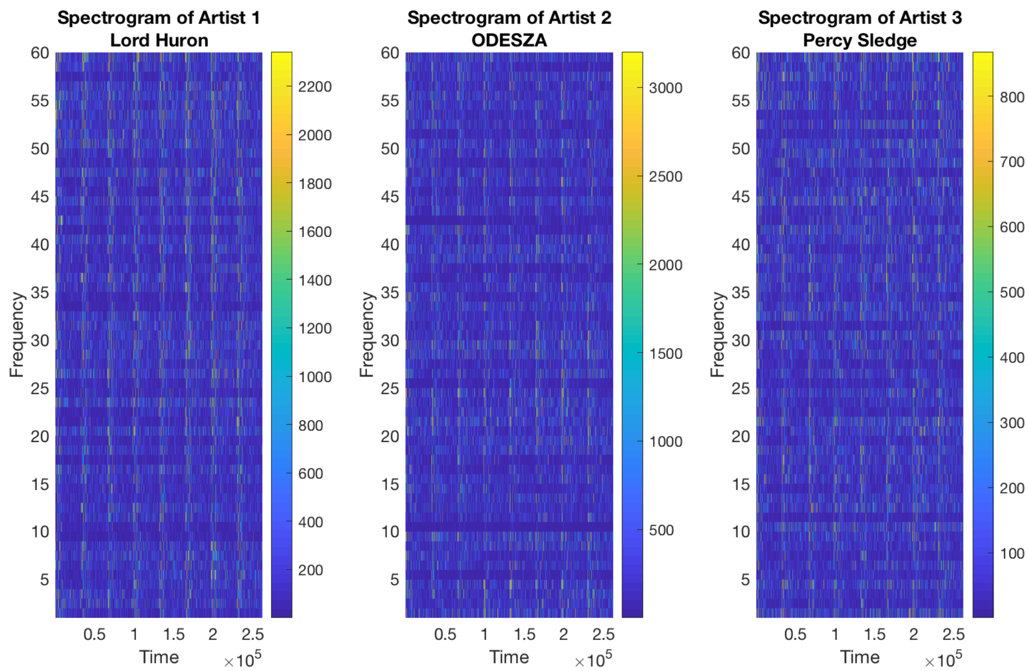
$$= \boldsymbol{w}UT \tag{EQ:8}$$

3 Methods & Algorithm Implementation

For this example, we first need to acquire a lot of audio data from various artists and genres. This was done by writing a function that loaded a audio file and sub-sampled it. Then a five second segment was clipped out for the sample data. Using the Fourier Transform **fft**, With the **specrogram** command in MatLab, we create a spectrogram for every sample then stored it in a csv file. Once all audio-frequency data is collected, another function is used to find the top three artists or genres with the most sample files, then the larger two are truncated to match the size of the third fewest sample files .

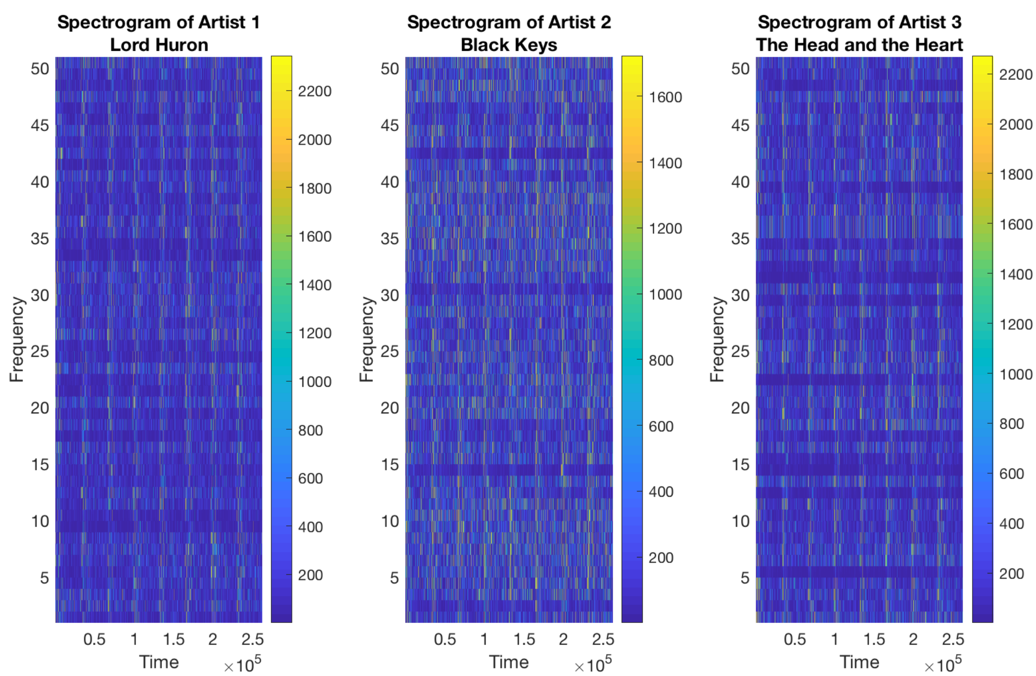
Next, we load in the data for each test case, and then randomly selected ten sample files as test files, and the rest are used to train our LDA algorithm. A function was created to take the SVD of the data and then to find **w** with the generalized eigenvalue problem stated above.

First First S_B and S_W were calculated and used to determine **w**, the projection of best fit. This was achieve with the **eig** command in MatLab.
Next, the SVD projection was used to find the LDA projection of the training data and test data. The LDA training data was used to determine the thresholds of the classifier. The LDA testing data was used to evaluate the accuracy of the classifier. This method was run for 30 times and then the average accuracy was calculated.



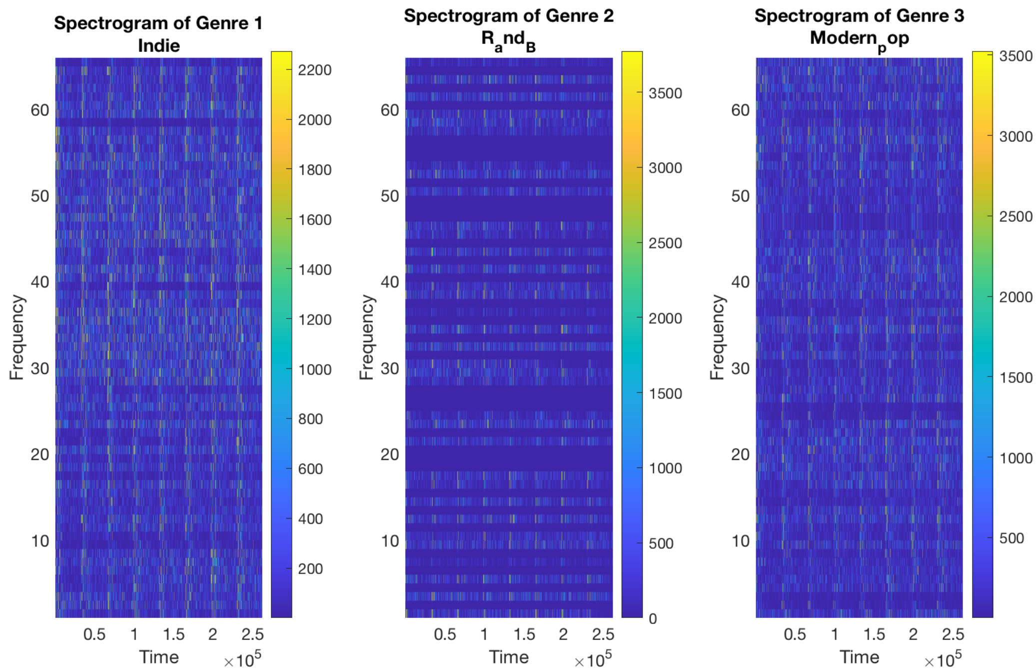
.

Figure 1: Plot of spectrogram for three artists of different genres.



.

Figure 2: Plot of spectrogram for three artists of the same genre (Indie rock).

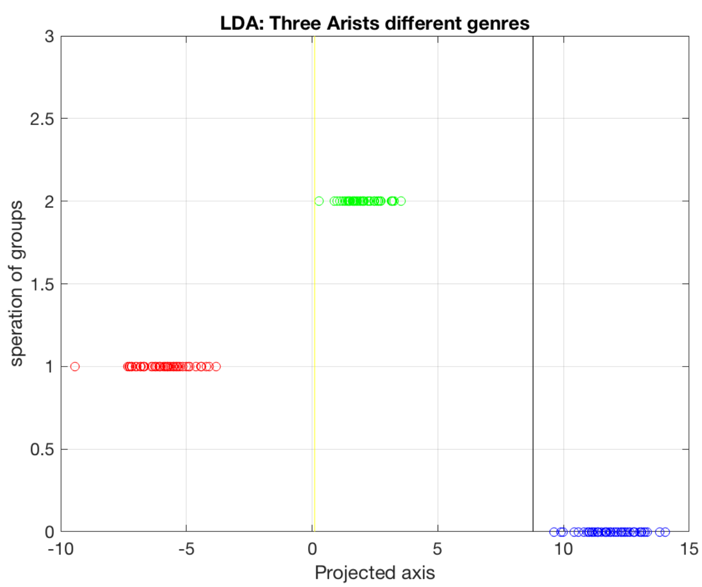


.

Figure 3: Plot of three different genres. Indie Rock, RB, and Modern pop.

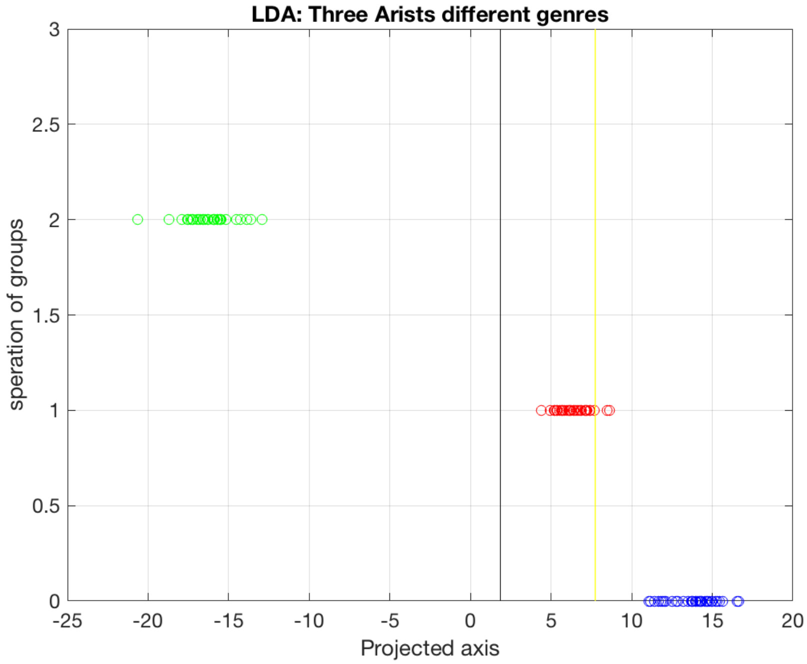
4 Computational Results

In this example we computed the singular values and PCA's for each test case.



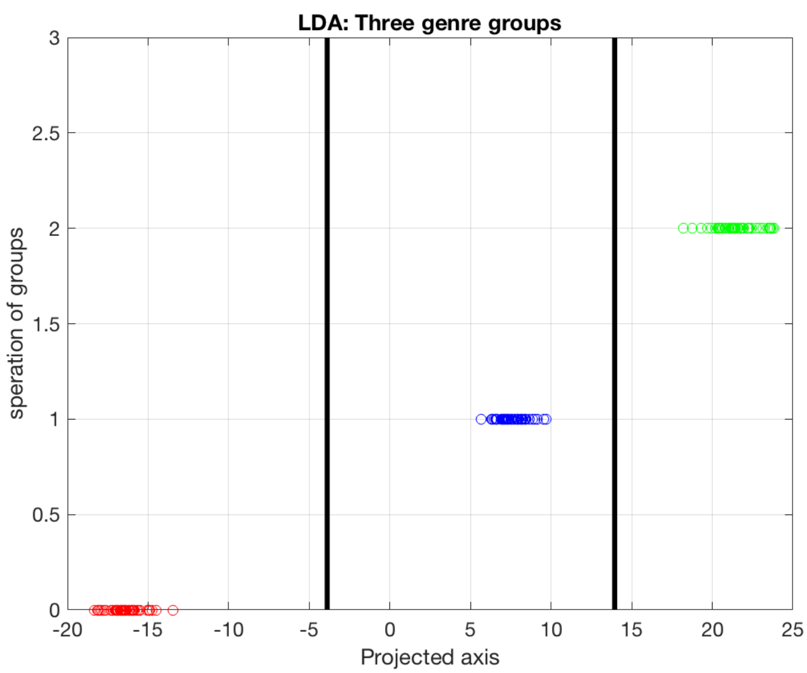
.

Figure 4: Figure shows the Linear Discrimination Analysis projections for test 1. Black and yellow lines are the thresholds between each group.



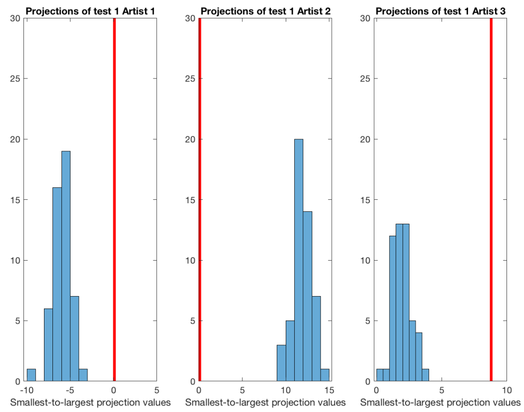
.

Figure 5: Figure shows the Linear Discrimination Analysis projections for test 2. Black lines are the thresholds between each group.



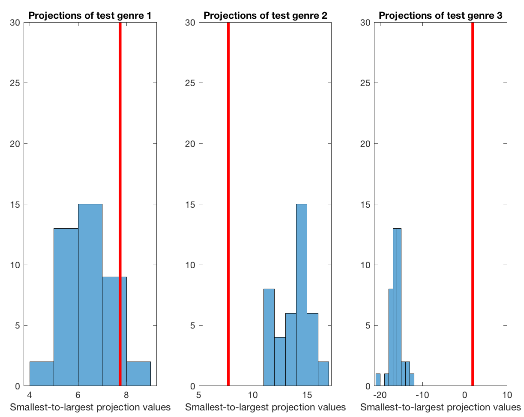
.

Figure 6: Figure shows the Linear Discrimination Analysis projections for test 3. Black lines are the thresholds between each group.



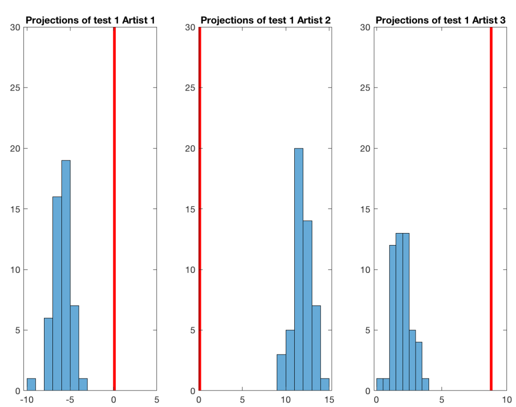
.

Figure 7: Plot shows the histogram from the test 1 LDA analysis and thresholds in red.



.

Figure 8: Plot shows the histogram from the test 2 LDA analysis and thresholds in red.



.

Figure 9: Plot shows the histogram from the test 3 LDA analysis and thresholds in red.

Test	Number of test files	Average Accuracy (%)
3 Artist different genres	66	28
3 Artist same genres	51	35
3 different genres	87	34

.

Figure 10: Table shows the number of test files and the average accuracy values for each Test case.

5 Conclusion

In this example we collected and processed audio data in order to generate a large number of five second clips. We took the spectrogram of each clip and build a large matrix of data for each genre or artist. We used the SVD algorithm to find Principle Components and we used them to find the Principle Components projection. We used this as a method for finding a projection of best fit for our data such that S_W was minimized, and S_B was maximize. This was the foundation of our Linear Discriminant Analysis. We then found thresholds the marked the boundary of each class and we used these thresholds to classify new data. Here we found that for all test there was not great at separation of classes with the the lines of projection we found. Each test set had upwards
Lastly,each test was ran 30 times and an average accuracy rate was found. This value tended to stabilize around a particular value. The average accuracy rate for each test was low, but more than random chance (33.33 percent for three classes). This could be improved by gathering much more training data.