Image Construction and Music Classification with SVD

Rachel Carroll*

03/13/2020

Abstract

This paper demonstrartes the role of singular value decomposition (SVD) in image construction and music classification. Images were analyzed to find correlations in structure across thousands of images in order to identify the significant attributes of facial images. Then music from various artist and genres were sampled and anayzed. The SVD was applied to spectrograms of the music samples to identify patterns in frequency associated with a given artist or genre. The results were used to create an algorithm that categorizes unknown music samples.

1 Introduction and Overview

This project is comprised of two parts, facial image construction analysis and music classification.

In part 1, two sets of black and white images of people's faces were analyzed. The first set contained images that were cropped and centered with neutral facial expressions but variations in lighting. The second set contained uncropped images of people with varying facial expressions (happy, sad, angry, etc) and oriented differently. The image data were initially held in matrix form, where the dimension of the matrix was the width and height of the image in pixels, and the values were associated with the pixel darkness. For both sets, all images were vectorized and stored in a matrix. Then the SVD was applied to identify significant patterns. Comparing the results of these two sets demonstrate both the power and limitations of using SVD to find correlations across images.

The second part highlights the use of SVD in data classification and clustering. This part comprised of writing a program that could classify a sample of music based on its sound wave data. The program was tested in the following three ways:

- classify artists of different genres
- calssify artists in the same genres
- classify a song sample by overall genres

2 Theoretical Background

Singular Value Decomposition (SVD)[1]:
The singular value decomposition is a great way identify correlations in large systems, and thus has the power to reduce redundencies in overrepresented systems. In short, it breaks down a matrix into fundamental elements, highlighting its most significant dynamics. Recall, the SVD breakdown form

$$A = U\Sigma V^*$$

*https://github.com/rachel-carroll/

[1]For additional background on SVD, see Principal_Component_Analysis in this repository

Where U and V are othonormal and contain the singular vectors of A and Σ is diagonal and contains the singular values. The most significant behavior is associated with the eigen vectors cooresponding to the relatively largest eigenvalues. The sections below will discuss how these useful properties of the SVD can be applied in many practices.

Low Rank Approximation
Let r be the rank of a matrix A and N be an integer such that $1 \leq N \leq r$. Then we can define A_N to be a rank N approximation of A as follows

$$A_N = \sum_{i=1}^{N} \sigma_i u_i v_i^*$$

where σ_i is the ith singular value, and u_i and v_i^* are the ith singular vectors. In the two norm, this is the best possible rank N approximation of A. Low rank approximation is essential in image compression because we can represent the full image very well using only a fraction of it's original data. We will see an example of this in the results section where an image is reconstructed extremely well using only 10 out of the original 168 columns.

Image Analysis with the SVD
As described above image compression is possible due to the fact that SVD can find correlated structures in a system. In this paper we use thousands of vectorized images of faces and use the SVD to find what structure, or patterns in pixel brightness, are associated with a human face. In this case, the columns of U provide patterns in relative darkness of pixels (orthogonalized). So every individual, unique picture in the system can be represented by a linear comination of the columns of U with coefficients from the columns of V. Since both U and V are orthogonalizes, Σ provides the pixel scaling. In image compression, the sign of the elements of V are significant because it determines if the associated pixels are relatively lighter or darker.

Music Classification with the SVD
In the music classification application, the SVD is still doing its same ol' thang. In this case we use it bit differently. The key to SVD analysis is knowing exactly what data you are measuring. In music data, we are measuring vectorized spectrograms of music samples. Therefore the vectors contain frequencies over time. Unlike in the image section, it doesn't make a lot of sense to look at the columns of U individually because they contain a hodgpodge of frequency clips stacked on top of eachother. Instead, we turn to the V matrix to categorize like pieces of music. In this case, a specific music piece is a linear combination of the columns of U, containing frequencies/time patterns. The columns of V contains the coefficients of this linear combination, which really is the code to get to the given music sample. This is why we look for patterns across the V columns to cluster like pieces of music.

3 Algorithm Implementation and Development

3.1 Part 1: Image Analysis

The following steps were performed in MATLAB for both the cropped and uncropped data set

1. Read in images
2. Vectorize
3. Store vectorized images in a matrix, A
4. Caculate the SVD.
5. Reshape the first four columns of U into images, which we will refer to as "eigenfaces" as the represent rpimary facial characteristics common across the images
6. Run a low rank approximation to recreate individual and average faces

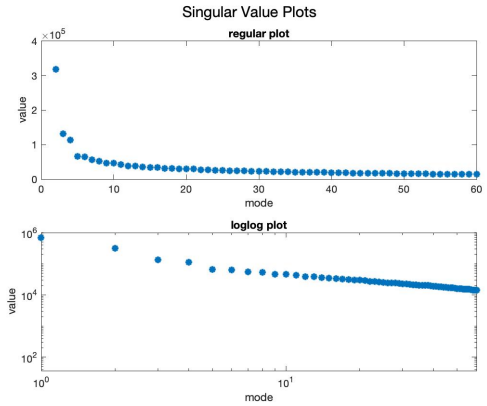
3.2 Part 2: Music Analysis

1. Read in music files
2. Take a series of 5 second samples across
3. Spetrogram and vectorize the samples
4. Split samples into testing and training data sets
5. Take the SVD of both sets
6. Using the V matrices of both sets, employ the "classify" function to predict the genre or artist in the testing sets

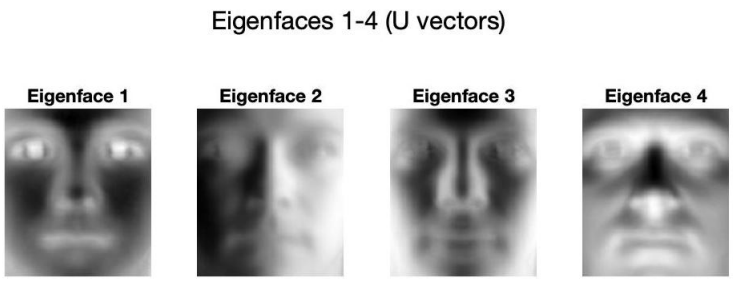
4 Computational Results

4.1 Part 1: Image Analysis - cropped

The Singuar value plots below give us an idea of how many significant modes exist to provide a good representation of the "face space" in which we are working. In an ideal case, there would be a clear distinction of significant and insignificant modes. Unfortunately in this case (and most cases) this distinction is up for interpretation and can be dependent on the overall goal of the anaysis. After the first few modes, the plots show a smooth descent in significance. However, the rate if steeper before 10 or 13 modes. The following sections will look into individual modes and interpret significant based on image reconstruction with low rank approximation.

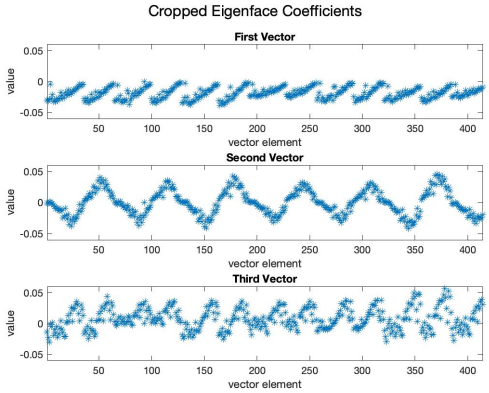


The following figure shows the images of the first four singular vectors $u_i \in U;\ i = 1, 2, 3, 4$. These faces display significant facial attributes associated with the given mode. For example, eyes and cheeks and forehead and significant in mode 1 and the bridge of the nose is significant in mode 4. Keep in mind, the original images are a linear combination of these eigenfaces. The coefficients of this linear comination is what creates the individual face out of these general facial building blocks.

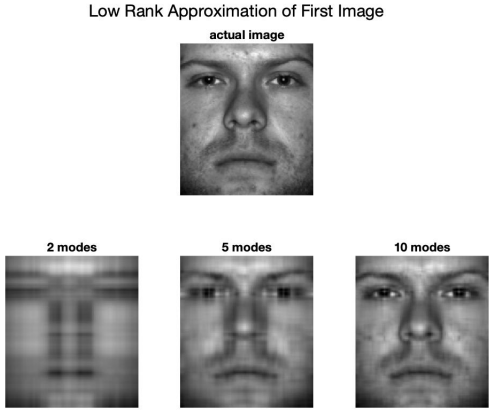


The figure below shows the coefficients fo the linear combination described. Note that the vectors are cut off after 450. Thereare actually 2414 images in total, but the pattern of coefficients continue as shown. Looking at the first plot, notice that all of the coefficients are negative. This makes sense because the image of the first U vector looks like an image negative. Each point in the plot represents relatively how much the first eigenface represents the associated original image (e.g. the first point in the first plot shows the representation of the first eigenface in the first image from the matrix A).

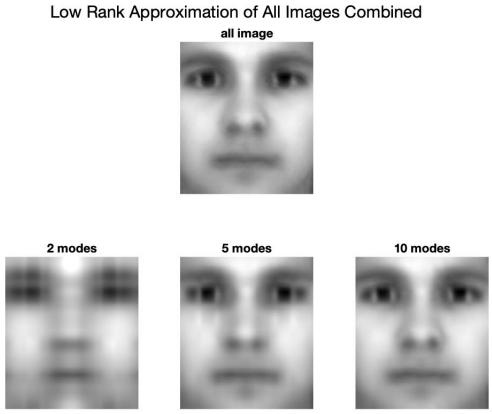
The second vector has both positive and negative coefficients. This aligns with the second eigenface. In the sample, some faces have even lighting and some have stronger lighting on the right or left side of the face. So the sign and magnitude of the second V vector plays into the lighting intensity on the sides of the face.



Going back to the idea of significant modes, we have below a low rank approximation of the first cropped image using 2, 5 and 10 modes. Notice that after 10 modes we have a good idea of what this person looks like. However the image is still a bit blurry and missing some finer details. Looking back at the singular value plot, this makes sense since the most significant modes appear to be below 10, yet the singular values between 10 and 20 are relatively higher than the ones past 20.

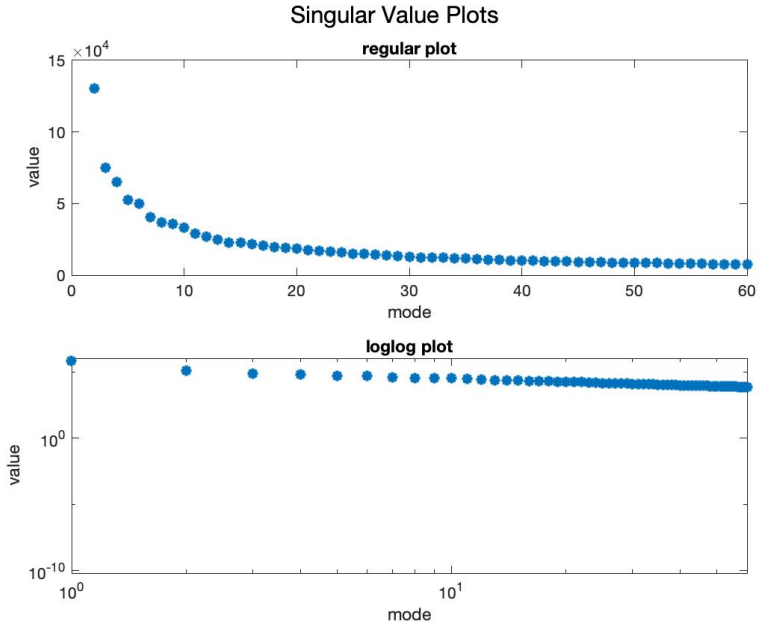


Below is a low rank approximation of the average of all faces. It turns out that the "average face" of a group of adult males looks more like a child or teen. The result is interesting and not too surprising that taking the average smooths over distinguishing facial details such as blemishes and wrinkles.

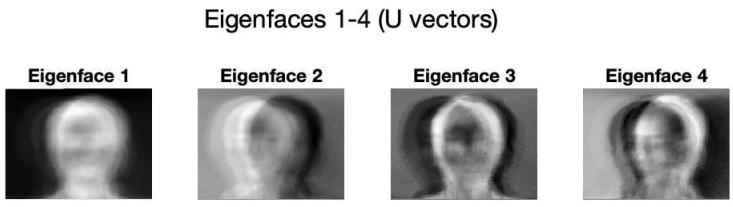


4.2 Part 1: Image Analysis - uncropped

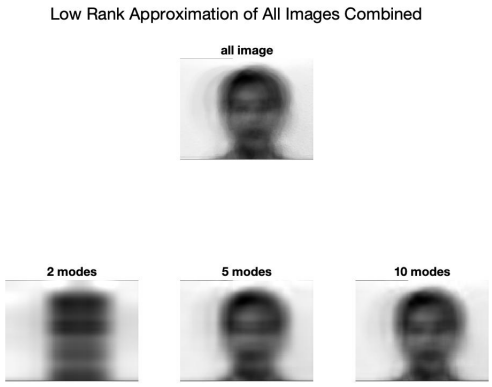
The singular vlues below follow a similar pattern as the cropped ones, but notice that the smooth decrease in significanse happens after the first node rather than the third.



Below the eigenfaces demonstrate that without the stardization of the images, the significant facial attributes are heavily influenced by the position of the head. Compared to the cropped versions, there is mush less detail on the detailed facial structures.

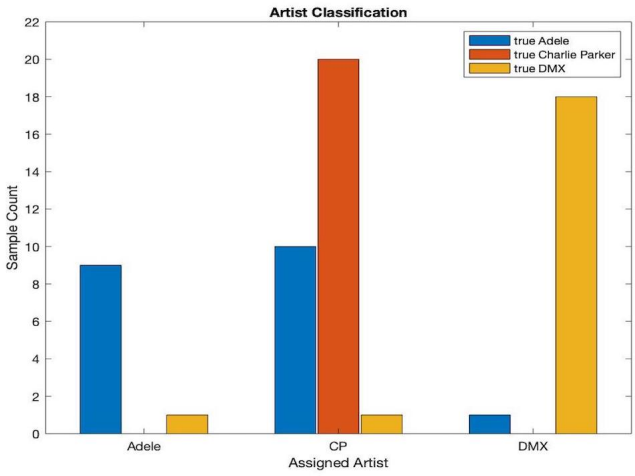


The low rank approximation of all faces, as expected from the eigenfaces, does not give much detail and is very blurry.



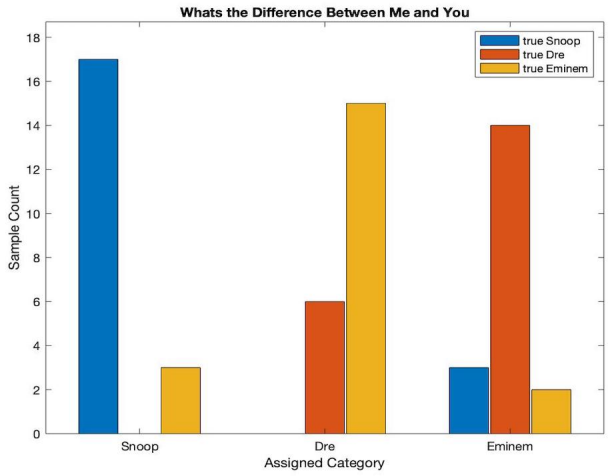
4.3 Music

The following figures show the classification results of the three different sampling runs. The first test is below, where three artists of different genres are classified. The results are very good for accurately identifying Charlie Parker and DMX music samples by artist. Adele was miscategorized as Charlie Parker slightly over half the time. Charlie Parker's music is consistantly high pitched and face paced. Adele on the other hand covers a variety of time, and frequency (voice and instrumentation) dynamics. Perhaps her faster and higher samples were classified as Charlie Parker because his music has a stronger correlation with this pattern in music.

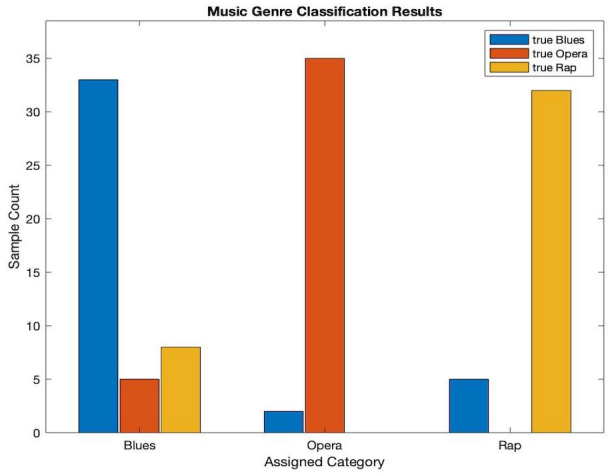


In Dr. Dre's Album, "2001" he collaborates with fellow artists Eminem and Xzibit to pose a fundamental question, "what's the difference between me and you?". As they lay down track seven, the artists describe mostly qualitative differences between themselves and other rappers in the game. At one point, Xzibit calculates an estimated difference of approximately five bank accounts, three ounces, and two vehicles. In this analysis we build on this idea by looking at the difference between Snoop, Dre, and Eminem based on their musical signatures in time and frequency content.

Snoop tracked very well but Eminem and Dre seemed to have been completely mixed up. The result is surprising because Eminem has a relatively higher frequency voice while Snoop and Dre emit lower frequencies. However, the Dre and Eminem samples have faster beats in general whereas Snoop maintains a chill flow. It is possible that the difference in frequency between Dre and Eminem was not enough to create a significant variance to be found by the SVD.



By far the best results came from overall music genre classification. This test used more samples from more artists. The distinct nature of these genres in time and frequency content as well as the larger data set both helped improve the results.



5 Summary and Conclusions

These experiements outline a couple of the countless areas to which SVD can be applied. Through these experiements we have taked a deep look into what the SVD can do in practice and how the three resultant matricies all play a role in extracting significant dynamics of a system. One reminder about the SVD is that is can be applied to any system. So understanding its behavior and significance is a powerful tool in any field that uses data.

6 Appendix A MATLAB functions used and brief implementation explanation

1. classify - The function takes a test data set, a training data set and a vector of labels associated with the rows of the test set. Then it labels the training data set and outputs a vector of the assigned labels.

2. five_sec_sample - custom made function for this project that takes an audio file name and and extracts a 5 second sample at a given point in the song. It outputs the song sample vector, sampling frequency,

time vector, and frequency domain vector

3. spec - custom made function for this project takes the spectrogram of an audio sample with given samping rate and gaussian filter width. There are also inputs to specify if the spectrogram should be displayed and if it should be putput in vectorized and matrix format.

4. vec_to_image - reshapes a vectorized image into given dimensions and outputs the image

7 Appendix B MATLAB codes

See the following repository for the MATLAB codes used:

- https://github.com/rachel-carroll/AMATH582

The repository contains the MATLAB files:

- images_cropped
- images_uncropped
- music_1_artist
- music_2_artistgenre
- music_1_genre