Music Classification

Leif Wesche

2-22-2018

Abstract:

In this assignment, music from a range of genres and artists was classified using principle component analysis along with a variety of supervised classification methods. Data sets consisting of music signal data from a range of genres and artists were decomposed using singular value decomposition. The classification methods used include support vector machines, the Naive Bayes algorithm, and decision trees. The singular value spectrum was used to select a range of principle components used in the classification algorithms. Selected principle component projections corresponding to each song in each data set were used as training and test data for the supervised algorithms. It was found that the three algorithms effectively classified the data, but had varying success rates over different types of data sets. This result highlights the need for cross validation using a variety of methods when classifying data.

I. Introduction and Overview

The goal of this assignment was to develop an algorithm to classify music using principle component analysis and classification methods. The first test intended to classify songs from three different bands of different genres. Music from Nirvana, Chopin, and Gucci Mane, were used for this first test. The second test was to see how well songs by artists of the same genre could be classified. Three rock bands were chosen for the second test, including Nirvana, Mastadon, and the Red Hot Chili Peppers. The third test was to see how well music from a broad spectrum of artists in certain genres could be classified. For this test, a database of music was provided by George Tzanetakis. The three genres chosen in the third test were classical, hip hop and country music. The spectrogram of each song was calculated, stored into one large data matrix, then decomposed using singular value decomposition. Data from the resulting V matrix, corresponding to each songs projections onto the principle components, was used to train and test classification algorithms. Three classification algorithms were compared, including Support Vector Machines, Naive Bayes, and Decision Trees.

II. Theoretical Background

Singular value decomposition, or SVD, is a method of decomposing a matrix A into U, V, and Σ components, in the form of $A = U\Sigma V^T$. The U and V matrices are two separate unitary bases, while Σ is a diagonal matrix composed of the singular values of A in descending order. The singular values are closely related to the eigenvalues of the matrix A. The columns of V can be thought of as the projections of each column in the data set A onto the principle components of the data matrix. In other words, the SVD method was used because the V matrix allowed the songs to be compared by projecting the songs onto a common set of principle components.

The first classification algorithm used was support vector machines, or SVM. The SVM algorithm is a supervised algorithm, meaning it learns to classify data using a set of labeled training data composed of a known number of groups. Supervised algorithms are ideal for the task of classifying music, since the number of groups are known and music that can be used as labeled training data is readily available. Support vector machine algorithms seek to draw lines of discrimination in data by using the training data to search for optimal lower dimensional spaces to project the data onto where there exists a clear separation in the data groups.

The second classification method used was called the Naive Bayes algorithm. Like SVM, Naive Bayes is also a supervised classification method. The Naive Bayes algorithm scores each piece of test data based on probability distributions relating the test data to the training data.

The decision tree method, also referred to as classification and regression trees, consists of flowing chain of true or false questions that are tailored to a specific data set. The trees typically start with a single true or false question that is asked of each data point that it attempts to classify. The trees branch off farther and farther until the data reaches a terminal point, at which point it is classified. The decision tree method is also a supervised algorithm, which is build off a labeled training set.

III. Algorithm Implementation and Development

Because of varying availability of music, slightly different sized sets of data were used for each test. For the first test, where artists among different genres were classified, 67 songs from each artists were used for a total of 201 songs. For the second test, where artists from the same genre were classified, 44 songs from each artists were used for a total of 132 songs. In test 3, where different genres were classified using multiple artists from each genre, 100 songs from each genre were used for a total of 300 songs. Each song was truncated to roughly a 5 second clip in the middle of the song before the data was analyzed.

Initially, all of the songs were loaded into Matlab into using Matlab's "audioread" function. All audio files used were of mp3 format. Only the left channel data was used, since it was assumed to be very similar to the right channel in most cases. To expedite processing, the song data was sampled at every other value. A spectrogram of each song was calculated and the absolute value of each spectrogram was taken, so that only the real parts of the spectrogram were taken into account.

The spectrograms of each song were then arranged into a single column vector, and the column vectors were arranged row wise to form a single data matrix X. The groups of songs that made up the data matrix X were arranged row wise, meaning that the data matrix was split into thirds, where each third represented a single group. For example, in the genre classification test, the first third of the matrix consisted of country songs, the second third was classical music, and the final third was hip hop. The data matrix X was then decomposed using singular value decomposition to form three new matrices, U, Σ, and V.

The singular values of each data set were analyzed by plotting the diagonal values of the Σ matrix. The relative range of these values was used to determine the number of principle components used for classification. The singular values of each data set are shown in Figures 1, **??**, **??**.

The songs classification algorithm worked by first extracting a number of principle component projections corresponding to each song. This was done by extracting values from the V matrix. The extracted projections were stored in separate matrices for each group of songs. For example, in the genre test where components 2 through 20 were used were used, the first 20 components of each column were extracted from the V matrix.

As a means to visualize the data, select components of the V matrix were plotted in 3 dimensional space. For example, the first three rows of each column of the V matrix may have been plotted as x, y, and z components in 3D space. The three groups were colour coded in order to distinguish clusters among them visually, so for example in the genre test, country was plotted in red, classical was plotted in blue, and hip hop was plotted in black. This was done to examine the data and see if clear areas of clustering could be identified. Figure 4 shows an example of one of these plots.

Next, training and test sets of data were constructed. Through experimentation, it was determined that classification worked well when 85% of the data was used for training and 15% was used for testing. So, 85% of the songs principle component projections were selected from the V matrix to use as training data, and the remaining projections were used as test data. To ensure that the distribution was even across all three groups of data, the V

matrix was separated into thirds column wise, and 85% of each section was used for training data and the rest was used for test data. For example, in the genres test, the first third of the test set data consisted of projections corresponded to 85 randoms country songs, the second third corresponded to projections of 85 random classical songs, and the final third corresponded to projections of 85 random hip hop songs.

This method ensured that each genre/artist group was represented evenly in both the training and test data, which was essential for both robustness and statistical evaluation of the sorting algorithm. A vector that was the same length as the training set was constructed to be used as labels for the training data. The first third of the data was labeled as "1", the second third was "2" and the final third was labeled "3". These labeled only served as a numerical input to the classification algorithms used later on.

The training data set and the corresponding vector of numeric labels were used as inputs to various Matlab commands in order to generate classification structures. These classification structures were then used with the test data to classify the test data. The support vector machine algorithm structure was compiled using Mablab's "fitcecoc" command and evaluated using the "predict" command. The Naive Bays method structure was compiled using the "fitcnb" command and evaluated using the "nb.predict" command. The descision tree structure was compiled from the test data with the "fitctree" command and evaluated with the "predict" command.

To test the accuracy of the classification, 100 trials were run for each test. In each trail, randoms songs were used as training data, and random songs were used as test data. To keep track of the accuracy, the each prediction was evaluated. If a prediction was correct, a value of 1 was added to a evaluation scalar associated with each group. After 100 trials, each evaluation scalar was divided by the number of trials and the number of songs in each groups test data. This resulted in a normalized evaluation scalar, which was then converted to a percentage. The average percentage classification accuracy was also computed for each algorithm. This percent accuracy data is shown in Figures 3, 4, and 5.

IV. Computational Results

First, the singular value spectrum for each test was plotted. In each case, the first singular value spectrum was normalized by dividing each value by the sum of the spectrum. The plots are shown in Figure 1 below, and in Figures **??** and **??** in Appendix A.

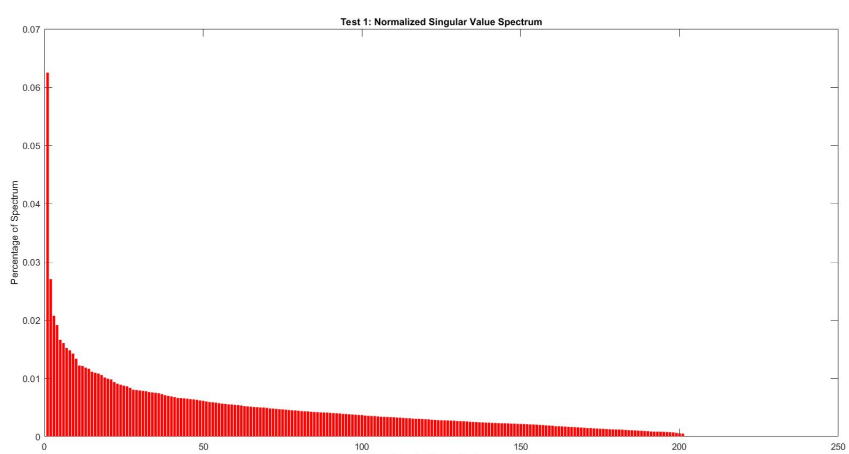

Figure 1: Test 1 normalized singular value spectrum.

The singular value spectrum in each test looked similar. In each case, the first value was by far the largest, representing over 6% of the total spectrum. However, the second value quickly dropped off to around 2-1.5% of the sum of the spectrum. From this point the values declined steadily. It was determined from the singular value spectrum that the first value corresponded to a principle component that was extremely prevalent in all the music data, thus would not be helpful in classifying the music. This hypothesis was confirmed by experimentation, when almost universally worse accuracy values were obtained when the first value was taken into account.

As the value spectrum progressed the values decreased, meaning the principle components corresponding to them were less and less prevalent within the range of songs. Theoretically, once the singular values got too small, the principle components corresponding to the values would begin to be more descriptive of noise or irregularities in certain songs than the musical trends as a whole. It was determined that a good range to cut off the singular value spectrum when classifying songs was around the 0.5% mark, which occurred around singular value 20-50, depending on the data set.

By analysis of the singular value spectrum, the ideal range of principle component projections to take into account for classifying the songs was determined and verified through experimentation. This range was found to be between principle components 2 trough 40 for test 1, 2 through 20 for test 2, and 2 through 50 for test 3. These varying ranges make sense considering the variable size of song databases and varying similarities between data in each test.

A set of three principle components were plotted in x, y, and z space for each test for the purpose of visualizing the data. An example of one of these such plots for test 2 is shown in Figure 4 below.

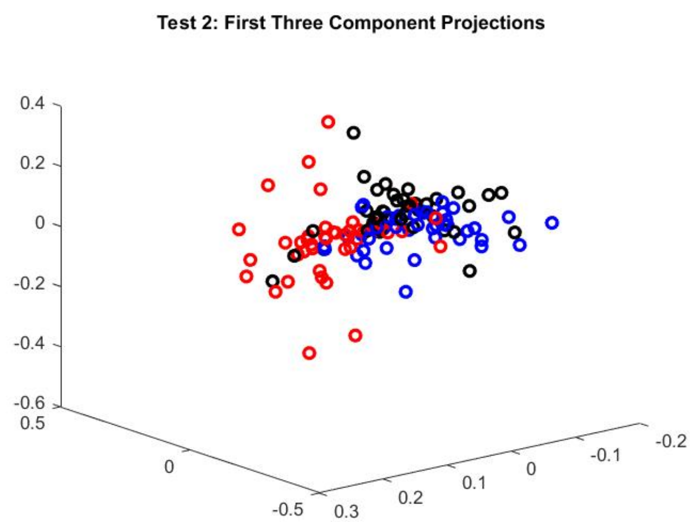

Figure 2: Test 2 first three principle component projections.
Red = Mastadon
Blue = Nirvana
Black = Chili Peppers

The goal of the classification methods is to determine a way to classify songs based on their position of their "n" principle component projections in "n"-dimensional space. In Figure 4, some notable trends in the clustering of the three artists can be observed in "n=3" dimensional space, but there is still a large area of crossover in the center of the plot. It was determined that significantly more than three principle component projections will be necessary to accurately classify the songs.

Finally, the songs were classified based on a select number of their principle components. In test one, where music from 3 artists of different genres was classified, it was found that the decision tree algorithm was overall most effective in classifying songs. In Figure 3 below, the outcomes of the SVM classification are shown in blue, the outcomes of the naive Bayes method are shown in green, the decision tree outcome is shown in yellow, and the overall accuracy scores of each method are shown in red. For each method and each test, 100 trials were run and the percent accuracy data shown in Figures 3, 4, and 5 was recorded by taking the average accuracy over all 100 tests.

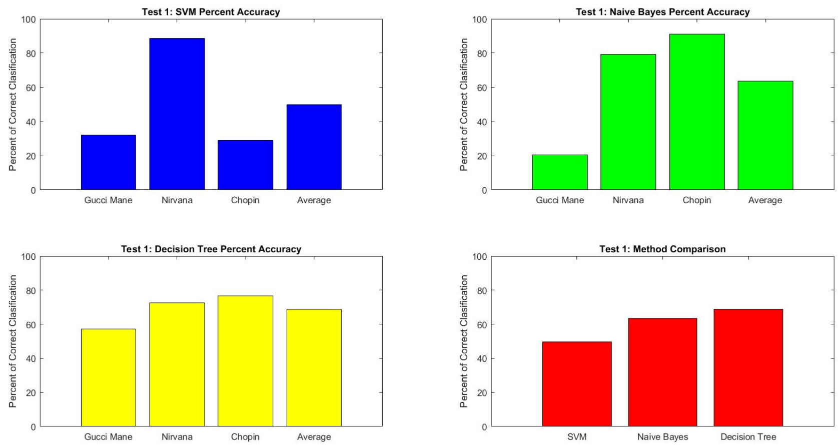

Figure 3: Test 1 classification results.

For test 1, the first 2 through 40 principle component projections were used to classify the three different artists, Gucci Mane, Nirvana, and Chopin. Overall, the decision tree method worked the best, with an accuracy of around 70%. The decisions tree method classified each artists at roughly equal success rates, while the SVM and Naive Bayes method seemed to favour particular artists. The success rate of each method for test one was fairly high, which fits the expected result, since the three groups of music classified were of three distinctly different genres.

In test 2, the first 2 through 20 principle component projections were used to classify the three different artists, Mastadon, Nirvana, and the Red Hot Chili Peppers. The percentage accuracy data is shown below in Figure 4, and colour coded the same way as the test 1 data.

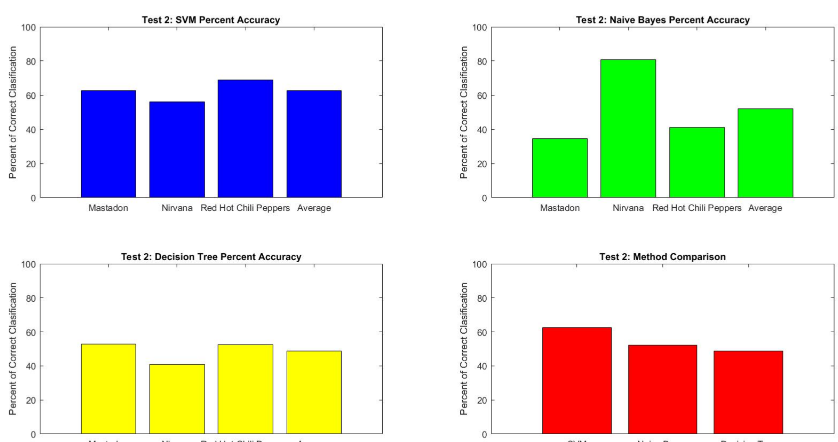

Figure 4: Test 2 classification results.

As expected the algorithms had a harder time sorting these three groups of music. This difficulty was expected due to the fact that the three rock bands compared in test 2 sounded much more similar than the groups compared in the other tests. However, the classification still worked fairly well, and the SVM method was found to yield the best overall results with a accuracy rate of about 60%. Interestingly enough, in test two the Naive Bayes method was found to favor one band over the others, while the decision tree and and SVM algorithms classified the songs at a fairly even success rate. This result is opposite to the result observed in test 1.

In test three, three genres of music were compared, with a variety of artists of the same genre in each group. The genres compared were country, classical, and hiphop. The percent accuracy data for test 3 was colour coded the same way as the other tests, and is shown below in Figure 5.

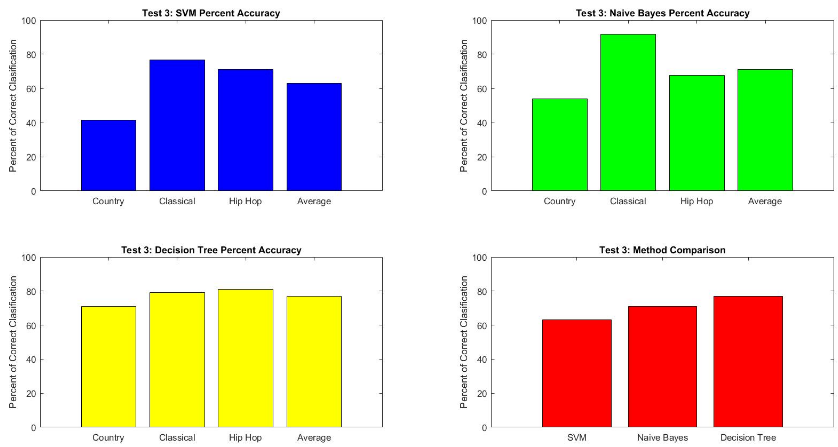

Figure 5: Test 3 classification results.

The three algorithms all worked best in test 3. The SVM and Naive Bayes algorithms might have slightly favoured some genres, they all classified the songs with a accuracy rate of over 60%, while the decision tree method worked best, with an accuracy rate of over 80%. The high accuracy of test 3 was most likely due to a combination of the facts that the most songs were used for classification in test 3, and that the groups used were all of the same genre but were slightly more broad than a single artists. This may have helped the algorithms accurately identify some songs that were within a genre but had slightly different features.

V. Summary and Conclusions

Principle component analysis along with supervised classification algorithms proved to be an effective method for classifying music. The singular value spectrum obtained by singular value decomposition was effectively analyzed to determine an appropriate range of principle components to use for classification. All three classification algorithms, support vector machines, Naive Bayes, and decision tree algorithms, were effective tools for classifying music, but there accuracy varied among different data sets. This highlights the fact that it is important to cross validate results using a variety of algorithms when classifying data.