Creating a Feature Vector to Identify Similarity between MIDI Files

Joseph Stroud
2017 Honors Thesis
Advised by Sergio Alvarez
Computer Science Department, Boston College

Abstract

Today there are many large databases of music, whether used for online streaming services or music stores. A similarity measure between pieces of music is extremely useful in tasks such as sorting a database or suggesting music that is similar to a selected piece. The goal of my project is to create a feature vector using only the actual musical content of a piece, ignoring metadata like artist or composer names. This feature vector can be used to measure distance between pieces in a feature space. To analyze the information contained in the feature vector, clustering and cla1snfk

Contents

1.1 Overview of Feature Vector Approaches to Music...4

1.1.1 Musical Similarity...tu..4

1.1.2 Feature VeVectu0 0 1 534.1 672.1 Tm[(190.4 Fea)9(t)-4(u.............)] TJETBT1 0 0 1 289.85 672.1 Tm[(...81 Fe

1. Introduction

1.1.1 Musical Similarity

albums associated with a piece [10]. Most attempts at creating features, including features describing rhythm, melody, and harmony, are mainly mathematical functions of the sound waves created by musical performance [12].

correlation, I used the Pearson correlation coefficient, which computes a score between -1 and 1 describing how positively or negatively correlated two variables are, with no correlation getting a score of 0 [1][6]. I also used principal component analysis to prepare my features. Principal component analysis transforms a set of vectors to produce another set of vectors (principal components) which are linearly independent and linear combinations of the original set. These vectors can be used as features in a new feature vector. The set of all of the principal components captures the underlying data exactly. A given number of principal components will capture a greater portion of the variance in the data than any other set that contains that same number of vectors. The number of principal components to be retained can be selected by deciding what fraction of the total variance is to be captured [1][6].

I tested my features on a dataset made of 165 MIDI files representing unique pieces of popular music.

Country	Brad Paisley	Carrie Underwood	Dolly Parton	Lady Antebellum	Luke Bryan
Folk Rock	Crosby, Stills, Nash, & Young	Elton John	James Taylor	Simon & Garfunkel	

diminished, and augmented triads and major, minor, dominant, and diminished sevenths. I also used the percentage of time no defined chord was being heard as a feature.

Four

I also added features that describe the rhythm of each instrument in the piece. To do this, I used the cumulative distribution function of the note lengths in a piece. A cumulative distribution function gives the probability that a random variable will have a value less than the input to the function. To calculate the cumulative distribution function I assumed that the probability that a random note length would be less than a given value was equal to the percentage of measured note lengths less than that value. For features, I used the first and third quartile of the cumulative distribution function of the note lengths for each instrument type. Because there are 128 instrument types in MIDI, this adds 256 features. However, all instrument types not present in a given piece while have first and third quartile values of zero, meaning that most of these 256 features for any given piece will be zero.

In order to test the features described in section 2.2, I used the machine learning software Weka [7] to do clustering and classification on the dataset described in section 2.1. This contains both a user interface and a Java API, both of which I used in my project. I did clustering tasks using the k-means algorithm and classification tasks using the logistic regression classifier, described in sections 1.2.1 and 1.2.2 respectively. I did clustering on four different feature vectors: one with all features I developed, one with all the features except for the note length quartiles, one with features created by doing a principal component analysis, and one with features selected by the Pearson correlation coefficient to have a correlation with genre. I did classification only on the features selected to have a correlation with genre.

Figure 2: Number of songs from each genre in each cluster for k-

country songs, every song has a large number of instruments. In the cluster with mainly pop songs, every song has a high proportion of electronic instruments. In additional cluster, cluster three is almost entirely composed of slower songs. One example of a song in this cluster is *Scarborough Fair*, by Simon and Garfunkel. The other cluster, Cluster 0, only contains songs in a minor key, while also including every available song in a minor key. Although a musical characteristic, this is not immediately apparent to a listener and so is not a very useful result. However, all five clusters are associated with some musical characteristic, and four of those five clusters contain songs that have shared characteristics that are easily audible. This means that the pieces are arranged in the feature space in a meaningful way.

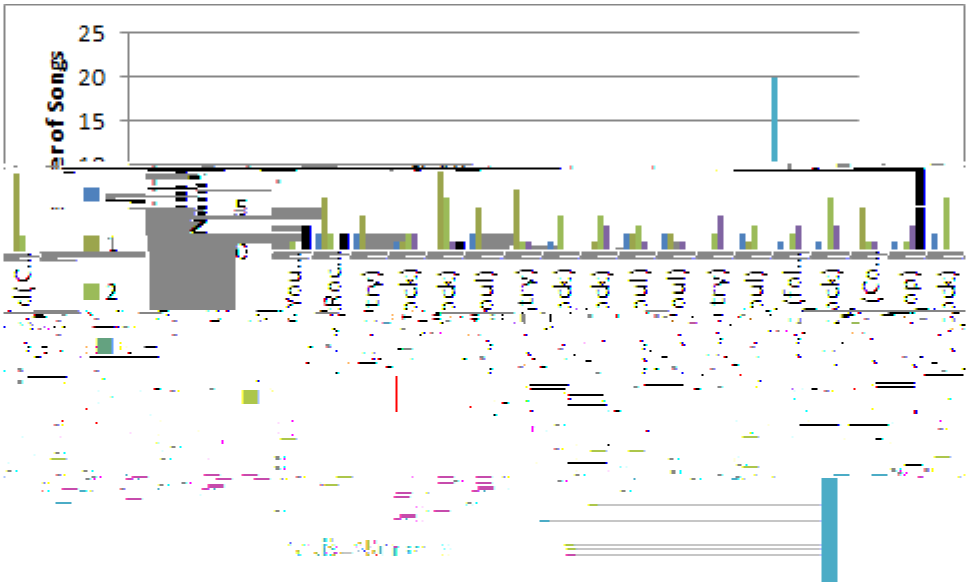

Figure 4: Number of songs by each artist in each cluster for k-means clustering where k = 5 with no note length quartile features

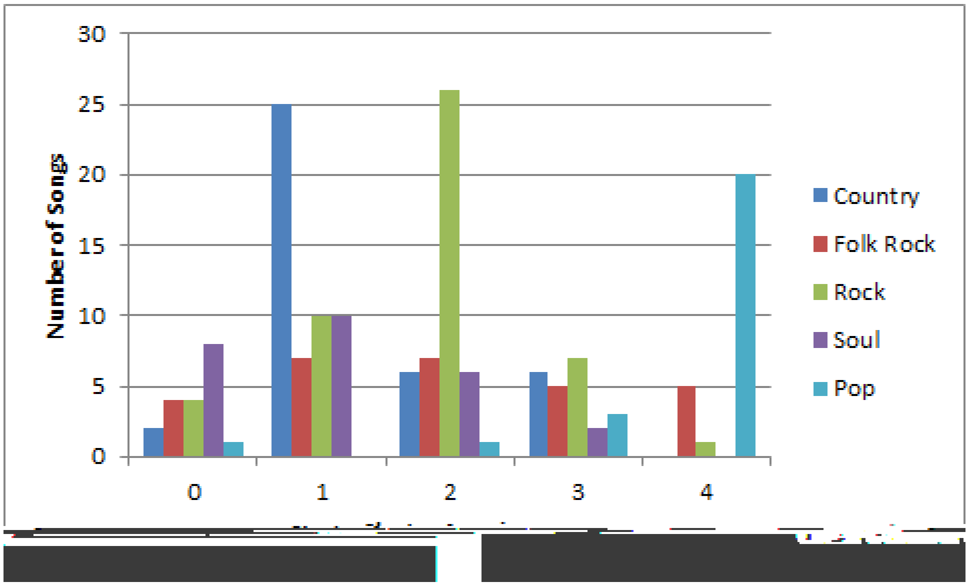

Figure 5: Number of songs from each genre in each cluster for k-means clustering where k = 5 with no note length quartile features

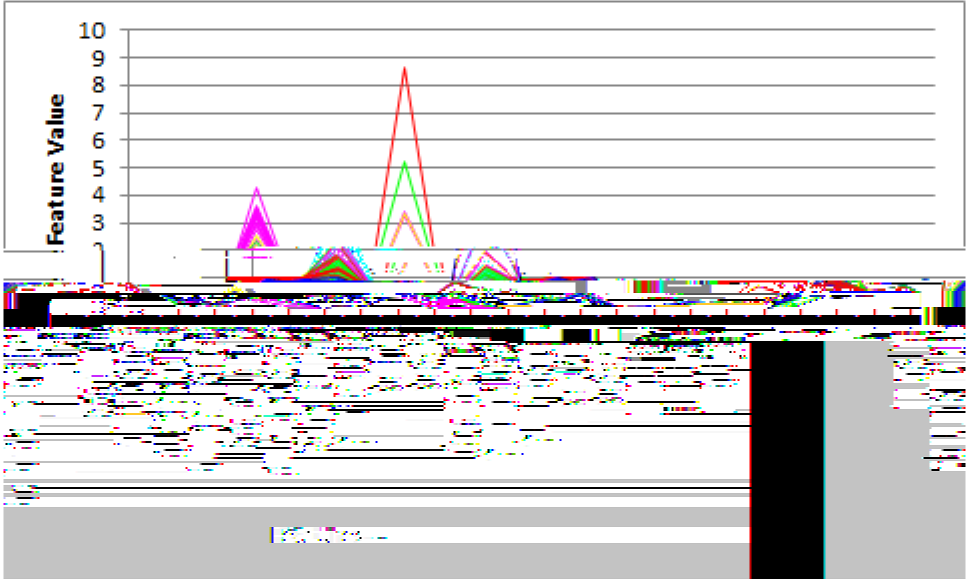

Figure 6: Parallel coordinate visualization of all non-note length quartile features colored by cluster

In the k-means clustering using features created by principal components analysis (see Figure 7), described in subsection 1.2.3, the results tended toward placing most pieces in a single cluster. This is not a successful use of feature reduction and does not contain very much musical information.

Figure 7: Number of songs from each genre in each cluster for k-means clustering where k = 5 with features created by principal component analysis

In the k-means clustering using features highly correlated with genre (shs

Feature	Pearson Correlation Coefficient
Time	0.1814
Pitch Range	0.1674
Number of Instruments	

Figure 10: Parallel coordinate visualization of features selected for correlation with genre colored by cluster

For classification, using a logistic classifier with 10 fold cross validation, the accuracy rate was about 56% (see Table 4). We can compare this with the expected accuracy if the classifier was placing songs into the five classes randomly (i.e., if the expected value of songs accurately classified per genre was one fifth of the total songs in that genre):

$$\mathbf{E}(\textit{Country songs correctly classified}$$

As the classifier performed much better than random chance, we can see that there is meaningful musical information encoded in the feature set made up of features highly correlated with genre.

	A	**B**	**C**	**D**	**E**
Country	25	5	2	6	1
Folk Rock	2	12	5	4	5
Pop Rock	0	1	23	1	0