Music Classification and Machine Learning

Student ID: xxxxx

May 13, 2013

Mathematics and Computer Science
Department

Biola University

I. Introduction

The technological revolution has completely changed modern society and culture, and continues to progress, even in third world and impoverished regions of the world. Its affects are nowhere more evident than in the ways that people now access and enjoy music. Physical media is quickly heading towards extinction, replaced by digital distribution services such iTunes,[1] while the likes of Pandora[2] and Spotify[3] provide subscription-based access to vast collections of music. Listeners are now given an unprecedented level of control over the availability and customization of their audio content and listening experience. With such power given to the consumers, content providers face increasing demands and expectations. To begin with, music content must be organized in a way that allows users to easily find what they want to listen to. In addition, users now desire recommendations and customized programming suited to their personal tastes and preferences.

The solution? *Music classification*: describing and categorizing music in a manner that is meaningful and comprehensible. With the vast amount of music content that already exists and is continually being created, it is impractical and near impossible for this to be done manually. Modern services and engines instead use *automatic classification* via *machine learning*, employing computers and algorithms to classify the content. Because of the immense depth and breadth of this cross-discipline field, known as *Music Information Retrieval*, a thorough analysis would be beyond the scope of this discussion. Instead, this paper will instead seek to survey

[1] http://itunes.com

[2] http://www.pandora.com

[3] http://www.spotify.com

some of the aspects and methodologies of automatic music classification to see how machines can be applied in the seemingly nebulous field of music.

II. Feature Extraction

Before a music file can be classified, there must be some way of describing and representing the characteristics of its contents. This concept is called *feature extraction*. The raw data contained in the audio files must be processed and analyzed in a way that allows machines to describe and understand the music. Rhythm, for example, refers to patterns and structures of the time-based movement in music, often buried in the complex textures of timbres and melodies. Rhythmic descriptors can be extracted using computational techniques such as the *beat histogram*, a predominant tool in music classification methods. The audio is first split into various frequency bands, which effectively isolates particular groups of instruments at various points in the music. After further processing and filtering the audio, the results are passed through an “autocorrelation function” to “detect the dominant periodicities of the signal;” in other words, it discerns where the strong beat or pulse of the music lies.[4] From this information, descriptors such as *tempo*, given in terms of beats-per-minute, can be extracted. Characteristics of the beat histogram itself also turn out to be useful in classifying genre. Figure 1 shows the beat histograms of two classical pieces (left column) and two pop pieces (right column). Pop music tends to be structured around spacing of regular, strongly pronounced beats, while classical music has more distributed, less pronounced beats, as evidenced in the histograms.[5]

[4] George Tzanetakis, *Tempo Extraction Using Beat Histograms*, 1.

[5] George Tzanetakis et al., *Automatic Musical Genre Classification of Audio Signals*, 3.

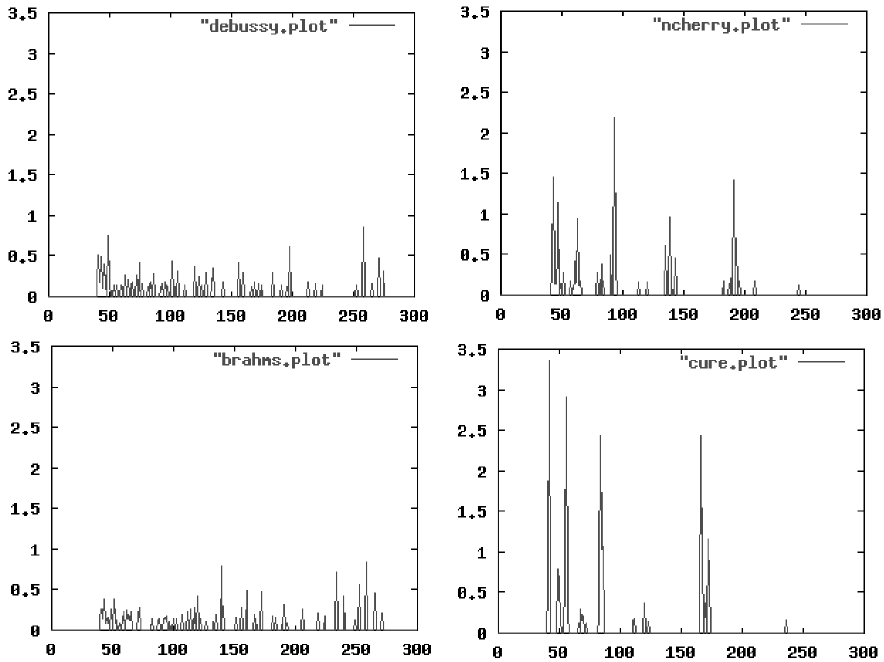

Figure 1. Tzanetakis et al., *Automatic Musical Genre Classification of Audio Signals*, Figure II.

Another incredibly useful descriptor is that of *dynamic complexity*, referring to the rate and character of dynamic changes in the audio.[6] The power spectrum of short frames of audio, usually of a length of a number of milliseconds each, are taken, then processed and converted into energy values and, eventually, loudness values. The fluctuation of these values is found through a dynamic complexity function, giving a value directly proportionate to the dynamic complexity of the music.[7] Figure 2 shows dynamic complexity values of ten different genres. Here it is usually found that "the lowest values correspond to Hip-Hop, Disco and Metal music in which the amplitude of the recorded signal is highly compressed (therefore, more constant). Higher values are obtained for classical and jazz music, two musical genres that maximally

[6] Enric Guaus i Termens, *Audio content processing for automatic music genre classification: descriptors, databases, and classifiers*, 79.

[7] Ibid., 80-81.

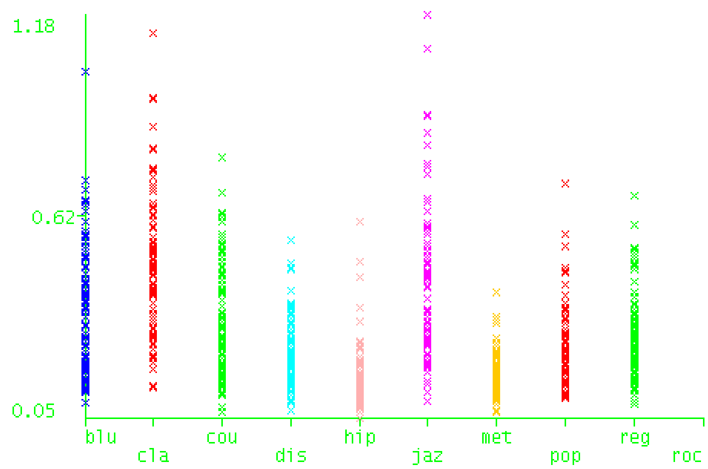

Figure 2. Enric Guaus i Termens, *Audio content processing for automatic music genre classification: descriptors, databases, and classifiers,* Figure 4.8

exploit the dynamics of the musical instruments."[8] Extracting features from music is no trivial task, but the rewards, as seen, are hugely beneficial for classification.

III. Decision Trees

Now that feature extraction has provided a way to represent the characteristics of the music, classification methods can now be applied to these descriptors. One such powerful technique is the *decision tree*. This consists of a structure of *nodes*, each of which tests a particular attribute. A node has two or more branches, which can be defined as true/false, a specific value, or less than, within, etc. depending on whether the attribute values are boolean, discrete, or numerical. These branches can either lead to further nodes or terminate in *leaves*, which specify a particular classification or a set of probabilities for multiple classifications. New instances are routed down the nodes until they reach a leaf, which determines their

[8] Enric Guaus i Termens, *Audio content processing for automatic music genre classification: descriptors, databases, and classifiers,* 81

classification.[9] Figure 3 shows an example of a simple boolean-based decision tree, along with the corresponding logic statements.

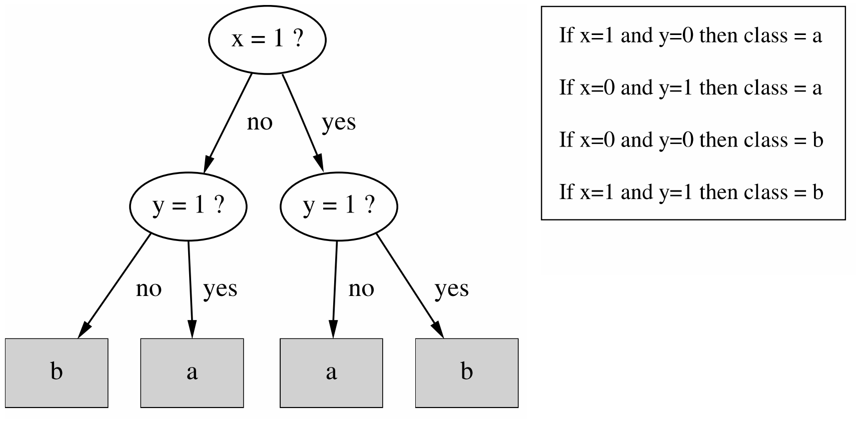

Figure 3. Witten and Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, Figure 3.3

A particular classifier that can be applied to music classification is the well-known CART, or Classification and Regression Tree. CART trees are built by splitting each node in such a way that produces the least combined *impurity* between the two branches. Complete purity means that there is only one class in a leaf, while complete impurity refers to an even distribution of the possible classes.[10] There are various entropy functions to estimate impurity that take into account the a priori probability of a class and the total number of possible classes.[11] Figure 4 presents a generic example of a music classification implementation of CART. The three possible genres of Rock, Classical, and Electronic are classified using unnamed features. Each node presents a feature, which splits into two branches around a particular value. The resulting leaves have excellent purity, leading to confident classification guesses.

---

[9] Witten and Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 82.

[10] Kris West, *Novel techniques for Audio Music Classification and Search*, 58.

[11] Ibid., 59

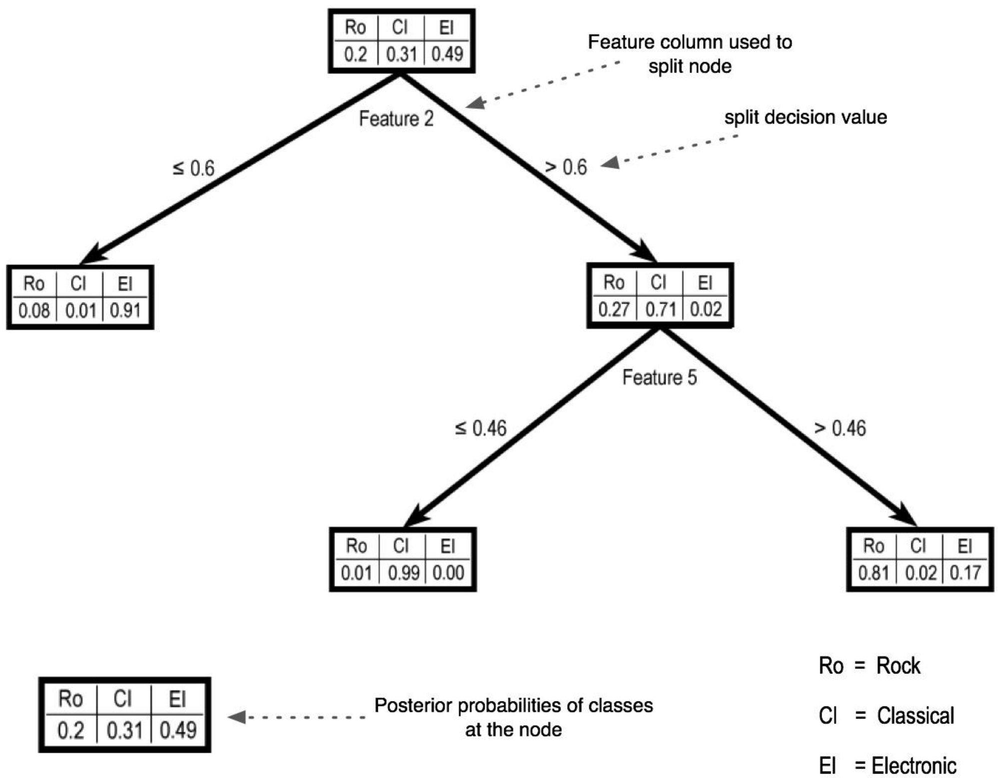

Figure 4. Kris West, *Novel techniques for Audio Music Classification and Search*, Figure 4.1

IV. Naïve-Bayes

Bayes' Theorem is a prominent idea in the field of probability, and also finds incredibly effective application in data mining and classification. The basic rule is as follows:

$$\Pr[H|E] = \frac{\Pr[E|H]\Pr[H]}{\Pr[E]}$$

This is meant to calculate a conditional probability of a hypothesis H, given evidence E.[12] Applying this directly to music classification does present problems, however. When dealing with a large number of features, their possible dependence upon one another must be taken into account. This leads to an over-complex equation that expands all the possible combinations of

[12] Witten and Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 90.

the features, such as follows:

$$p(G, F_1, F_2, \ldots, F_n) = p(G)p(F_1, F_2, \ldots, F_n|G) =$$
$$= p(G)p(F_1|G)p(F_2, \ldots, F_n|G, F_1) =$$
$$= p(G)p(F1|G)p(F_2|G, F_1)p(F_3|G, F_1, F_2) \ldots p(F_n|C, F_1, F_2, \ldots F_n - 1)$$

In addition, such an approach also requires a rather large *training set*, a data set of pre-classified instances used to build the classifier.[13] In order to alleviate these issues, it is possible to use a variation on Bayes theorem known as *Naïve Bayes.* Here, the variables, or features in this case, are treated independently of one another, resulting in a much neater solution:

$$p(G, F_1, F_2, \ldots, F_n) \approx p(G)p(F_1|G)p(F_2|G) \ldots p(F_n|G) = p(G)p(F_i|G)$$

Such a classifier can be built using a much smaller training set, and, in spite of its simplicity, is surprisingly effective in its application.[14]

V. Feature Selection

With any classifier, and particularly so with ones such as the Naïve Bayes, it is vital to use *feature selection* to reduce the set of features used to build classifiers. This process seeks out the most relevant features while pruning out inconsequential ones, greatly reducing the complexity and overhead of classification methods while increasing their effectiveness. One such method, based on the genetic algorithms, approaches features like genes in an evolutionary process. Using a feature vector of fixed size, an individual with random features is created. A classifier is created and tested based on that individual. From this point forward, a continual process of simulated crossover and mutation creates a successive chain of new generations, in which the most effective individuals are retained, akin to the idea of natural selection. The

[13] Anastasia Semyonova, *Music Genre Detection Using the Naïve Bayes Classifier*, 12.

[14] Ibid.

process is stopped when it converges, i.e. the changes mostly stabilize, or at an arbitrary point.[15] Feature selection processes such as these result in high-accuracy classifiers with greatly reduced preprocessing overhead.

VI. Support Vector Machines

While methods such as decision trees and the Naïve Bayes learn from a set of training data to develop abstract classifiers, an alternative is *instance-based learning*, which refers back to stored particular instances in order to classify new cases.[16] *Support vector machines (SVMs)* are algorithms that employ this type of learning. An SVM creates a linear model known as the *maximum margin hyperplane*, illustrated in Figure 5. Here, two classes represented by white and black circles are found to be separable; in other words, their enclosing outlines cannot overlap.

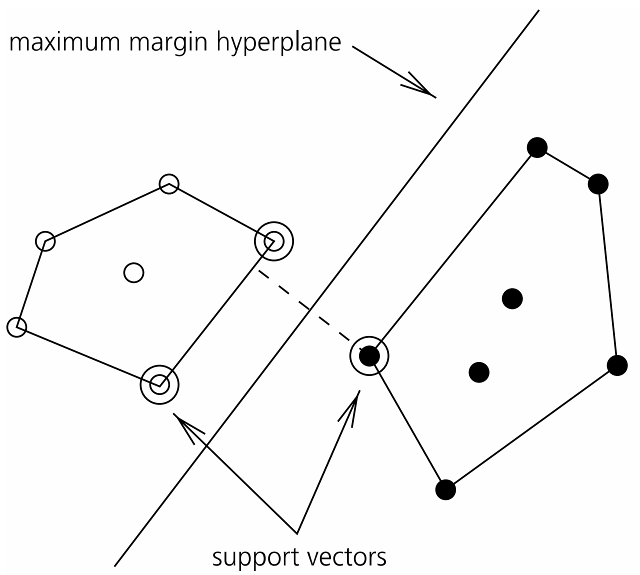

Figure 5. Witten and Frank, *Data Mining: Practical Machine Learning Tools and Techniques,* Figure 6.8

---

[15] Carlos N. Silla Jr. et al, *A Machine Learning Approach to Automatic Music Genre Classification,* 14.

[16] Witten and Frank, *Data Mining: Practical Machine Learning Tools and Techniques*,128.

Thus, these instances are classifiable using a hyperplane, i.e. a line in this case. “The maximum margin hyperplane is the one that gives the greatest separation between the classes—it comes no closer to either than it has to… The instances that are closest to the maximum margin hyperplane—the ones with minimum distance to it—are called *support vectors*.” [17] All other instances besides the support vectors can be discarded.

While SVMs are a powerful classification tool in general, they become immensely useful in music classification with the involvement of *active learning.* Rather than relying on a pre-processed classifier built upon a previously established set of data, active learning proceeds on the basis of user feedback.[18] For example, when building a classifier, the user can be provided with set of instances with optimal spread, from which positive examples can be chosen to form a basis for initial classification. As the user is presented with new content, every instance of negative feedback can provide the classifier with new support vectors.[19] The first advantage of this approach is the reduced overhead every time the classifier needs to be modified, since the classifier is based upon the support vectors alone, rather that built from comprehensive set of training data. This also means that this classification is more dynamic and adaptive to new errors, making up for any deficiencies in the initial training set. Finally, and most notably, this method of classification takes into account the individual user, including personal taste and interpretation, which are absolutely vital considering the many subjective and perceptually based aspects of music.[20]

[17] Witten and Frank, *Data Mining: Practical Machine Learning Tools and Techniques*,128.

[18] Michael I. Mandel, *Support Vector Machine Active Learning for Music Retrieval,* 6.

[19] Ibid., 7.

[20] Ibid., 6.

VII. Conclusion

Automatic music classification is certainly a daunting task, spanning across many fields such as musicology, music theory, signal processing, and data mining. One of the greatest hurdles faced is the task of representing the characteristics of perceptual music in a way that is usable by machines, which feature extraction seeks to solve. Once those features are obtained, core methods such as decision trees and Naïve Bayes provide quite effective solutions to music classification. In an age and culture where users and consumers are more than willing to voice their feedback, methods such as support vector machines add an unprecedented level of effectiveness, providing robust and adaptive solutions that can be tailored to the individual. At first glance, machine learning and music sometimes appear to be polar opposites in separate worlds. However, there is no denying how much automatic music classification has helped millions of people around the world to access and enjoy a timeless and transcendent art.

Bibliography

Mandel, Michael I. *Support Vector Machine Active Learning for Music Retrieval.* http://mr-pc.org/work/mmsj05.pdf (accessed May 12, 2013).

Semyonova, Anastasia. *Music Genre Detection Using the Naïve Bayes Classifier.* (Bachelors Thesis, University of Tartu) 2009.

Silla Jr., Carlos N. et al. *A Machine Learning Approach to Automatic Music Genre Classification.* www.lbd.dcc.ufmg.br/colecoes/jbcs/14/3/002.pdf (accessed January 21, 2013).

Termens, Enric Guaus i. *Audio content processing for automatic music genre classification: descriptors, databases, and classifiers.* (PhD Dissertation, Universitat Pompeu Fabra, Barcelona) 2009.

Tzanetakis, George. *Tempo Extraction Using Beat Histograms.* http://www.music-ir.org/evaluation/mirex-results/articles/tempo/tzanetakis.pdf (accessed May 11, 2013).

Tzanetakis, George, et al. *Automatic Musical Genre Classification of Audio Signals*. http://ismir2001.ismir.net/pdf/tzanetakis.pdf (accessed January 22, 2013).

West, Kris. *Novel techniques for Audio Music Classification and Search.* (PhD Dissertation, University of East Anglia) 2008.

Witten, Ian H and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques,* Second Edition. San Francisco: Elsevier Inc., 2005.