Song genre classification via training on segment chromas and MFCCs

Caleb Jordan
Stanford University
grnstnrd@stanford.edu

Alex Cope
Stanford University
alexcope@stanford.edu

Abstract

Automatic song classification has been an open challenge for a number of years as it has many useful applications including dataset labelling and hit song prediction. There exists research into many different approaches, including all of the standard feature-based classifiers. In this paper, we experiment with classification by multiple Hidden Markov Models trained on chroma segment data from each song. These models are theoretically much better suited to capture characteristics of songs for two reasons. First, they capture the notion that consequetive segments of music are not independent. Second, we derive the chroma and timbal data from the song pitches and aggregate them intelligently over bars of songs, producing time-data that better aligns with chords and real musical features. Ultimately, our model performed much worse than previous models; however, we hope think the results of this model will at least inform our understanding of how machine learning can be applied to musical data.

Introduction

Understanding music is perhaps one of the most open-ended challenges in machine learning. While there are many specific tools to analyze features of music, including pitches, beats, tempo, etc, there are more general problems like song classification and playlist generation that are still being developed. These more general problems are challenging both because of the large number of factors involved, as well as the subjectivity of the results. Classifying songs as ”hit” songs, for example, is an immensely practical and lucrative application, but so far, this classification depends on features completely unrelated to the music.

In this paper, we approach the problem of song classification by genre. Related works have attempted to classify songs by genre using support vector machines and similar models to various degrees of success (Tzanetakis, Essl, and Cook 2001; Anan et al. 2011). To differentiate from previous models, we exploited the time series nature of pitch and timbre information by using Hidden Markov Models to model our data. Instead of using features like tempo and mode to classify music, we used short time features, features derived from short time intervals in the music (Meng and Shawe-Taylor 2005).

Chroma vectors encode pitch information, and when presented in sequence represent melodic and harmonic patterns over time. Different genres of music use different harmonic frameworks as their basis – classical music employs a strict set of contrapuntal and chord progression rules that have largely been deemed irrelevant in modern music, chord sequences in pop music are very simple and center around three or four chords, and jazz music features more color (dissonant) tones and a different standard of chord progressions. Mel Frequency Cepstral Coefficients (MFCCs), unlike chroma vectors, are non-perceptual features and have been used in many audio analysis tasks including speech recognition; they can be thought to encode timbral information; i.e. information about the instrumentation of a song. Different genres of music utilize different instruments (consider a folk song and a heavy metal song) so MFCCs are often used in genre classification (Rump et al. 2010).

Dataset description

All of the musical data came from the Million Song Dataset (Bertin-Mahieux et al. 2011), a dataset of information of one million popular songs which is freely available online via Amazon Web Services. The MSD includes meta data for songs including the estimated key of the song, mode of the song, artist, title, and tempo. In particular, each song includes arrays representing beats, pitches, and timbre throughout the song. These songs are not labelled by genre, however.

To acquire songs labelled by genre, we downloaded song titles from Sharemyplaylists.com, labelled by the genre of the playlists with which they were associated. Of the 3721 title and artist pairs obtained, about one-third were present in the MSD.

For the final dataset, we collected 1033 songs from 18 different genres, with labels and metadata. As a preprocessing step, we sampled the pitch and timbral data (about 1000, 12-dimensional elements originally) into 80 buckets. These buckets were aligned with the onset of measured with the hope that when combined in sequence, they would hold useful chord progression data. Furthermore, we normalized the pitch data by key (using the key information from the MSD), transposing the data so that each song is effectively in the key of C.

Approach

In order to capture the internal structure of each song and the time-dependent nature of the pitch data, we used a Hidden Markov Model to model the data. The observed variables were the normalized, 12-dimensional vectors of pitch data. We assumed that the value of the discrete, latent variable at each timestep determines a normal distribution from which the corresponding pitch data was drawn.

That is, with observed variables $X^{(t)}$ and unobserved variables $Y^{(t)}$ for each segment t, we assume that $X^{(t)} \sim \mathcal{N}(\mu_k, \Sigma_k)$ given $Y^{(t)} = k$. The parameters of one HMM models are as follows:

- ϕ_k, the probability that $Y^{(1)} = k$ for a sequence.
- T, the transition matrix where T_{ij} is $P(Y^{(t+1)} = j | Y^{(t)} = i)$.
- μ_k, the mean corresponding to label k.
- Σ_k, the covariance matrix corresponding to label k.

To build a classifier, we trained a different model for each genre, using the appropriate labelled sequences. The prediction for a test song is the genre corresponding to the model for which the test song has the highest likelihood:

$$h(x) = argmax_k L(x|\theta_k) \tag{1}$$

is:

$$\begin{aligned} L(x;\theta_m) = \sum_Y (P(Y^{(1)}) P(X^{(1)} = x^{(1)} | Y^{(1)}) \\ \prod_t P(X^{(t)} = x^{(t)} | Y^{(t)}) P(Y^{(t)} | Y^{(t-1)}))) \end{aligned} \tag{2}$$

which is calculated efficiently using clique tree inference (Mengshool 2010).

Experiments

To analyze and preprocess the pitch data, we used a custom Python script to connect to Amazon's EC2 service to access the Million Song Dataset from the cloud and the Python numpy library to normalize and cull the pitch and timbral data vectors. Modeling and learning the HMM was written in matlab, using the expectation-maximization algorithm with clique tree inference. We ran inference on the HMMs within EM to calculate the log probability.

We tested both chroma and MFCC features (Fig. 1). Chroma features are represented as a 12-dimensional vector, encoding the octave-invariant intensity of each pitch on the chromatic scale. A C-major chord, for example, would be represented by a chroma vector with high intensities on the features representing C, E, and G. We transposed each song into the key of C by appropriately shifting the chroma vectors so we could classify songs more accurately by chord patterns. MFCC features are spectral features which encode the timbre of a segment.

Tuning the models involved experimenting with a few different hyperparameters. The most interesting of these was the cardinality K of the latent variables for the HMMs, which we swept from 2 to 24 (Fig. 2). The choice of K had little effect on the results; however, larger K values (starting with 5 to 6 depending on the amount of data) caused fitting gaussian parameters to be an underconstrained problem. This lead to singular covariance matrices, necessitating early termination of the EM algorithm. However, even with early termination, our results demonstrated impressive overfitting on the medium dataset, with about 3% training error. Unless labelled otherwise, all accompanying plots were made using a K value of 4.

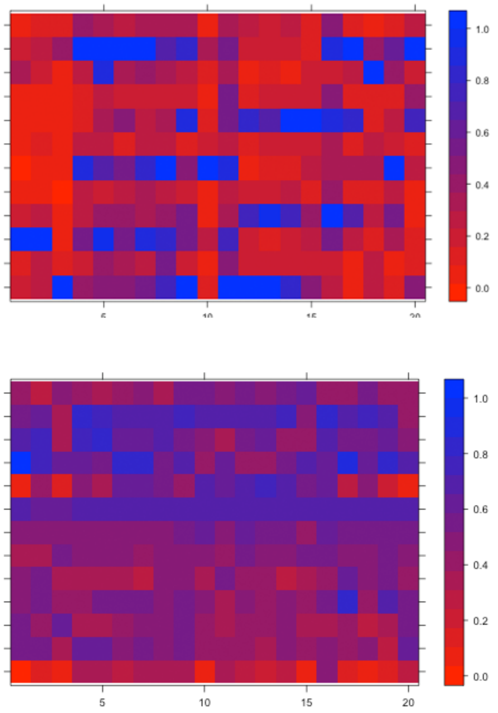

Figure 1: Measure-aligned key-normalized chroma (top) and MFCC (bottom) features for the first 20 bars of Fleet Foxes' *White Winter Hymnal*

Our results show that the HMM is quite simply a poor model with which to solve this problem. The plots presented represent tests on just over 300 songs, with 12 genres having enough data. The learning curve (Fig. 3) suggests a biased model as training error increases with more training data, which was slightly unexpected given the sheer number of parameters the model uses. While the testing error (around 80%) is slightly better than chance (91.7% for 12 genres), it is certainly unacceptable as a real genre classifier.

Upon close inspection, some interesting results appear. First is the observation that, when limiting the covariance matrix to be diagonal, the models performed incredibly poorly with around 90% training and testing error; i.e. the assumption that pitches are independent increases the error to chance. This suggests that these models do somewhat capture the dependencies between the different notes in segments. Second, studying differences between predicted labels and real labels of test data revealed interesting rela-

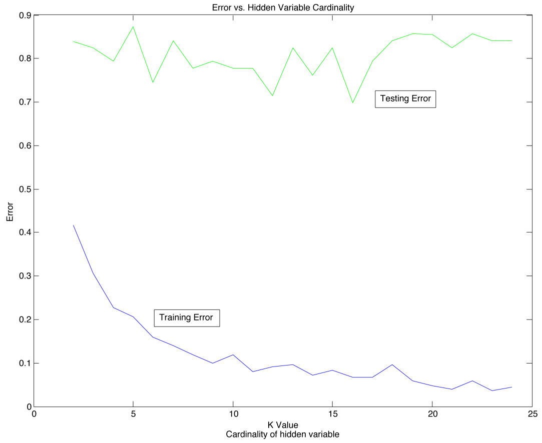

Figure 2: Learning curves over K values of the HMM

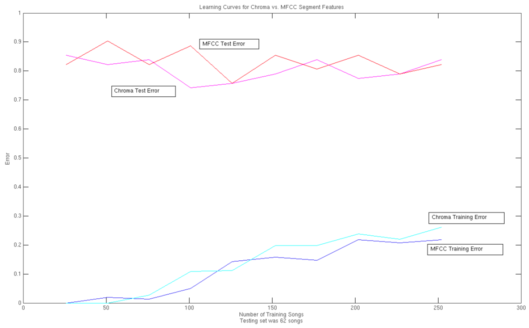

Figure 3: Learning curves over both chroma and MFCC segment types for 300 songs.

tionships. In particular, the largest number of misclassified songs were all blues songs mistakenly identified as pop songs. Ultimately, we believe these models may be capturing important characteristics of the music that have nothing whatsoever to do with the genre, but further investigation would be required.

Conclusion

We have explored a previously untried method of automatic song classification: using Hidden Markov Models to model time-segment pitch data within songs. These models were entirely unsuccessful at classifying songs by genre. However, there are several things to be learned from the expedition and its poor performance compared to previous attempts. First, the genre of the songs seems more or less independent of the per-segment data within a song. Second, by comparison of different distributions within the models, it is clear that ours does capture relationships between pitches within chords. Finally, while these models do not classify genre well, they may yet have success with other problems, as commonly occurring misclassifications suggest that the models learn similarities between certain genres. Further experiments with Hidden Markov Models and song data should focus on other problems; while genre classification is entirely impervious, it is possible that these models would perform quite well separating songs with different musical structures or common chord progressions. In a different domain, these models may be better suited to automatically detect chord progressions and other musical patterns within songs. For now, the hypothesis that time-series models can automatically classify songs is successfully disproven.

References

Anan, Y.; Hatano, K.; Bannai, H.; and Takeda, M. 2011. Music genre classification using similarity functions. In *Proceedings of the 12th international society for music information retrieval conference*, ISMIR '11.

Bertin-Mahieux, T.; Ellies, D. P.; Whitman, B.; and Lamere, P. 2011. The million song dataset. In *Proceedings of the 12th international society for music information retrieval conference*, ISMIR '11.

Meng, A., and Shawe-Taylor, J. 2005. An investigation of feature models for music genre classification using the support vector classifier. In *Proceedings of the 6th international society for music information retrieval conference*, ISMIR '05.

Mengshool, O. 2010. Understanding the scalability of bayesian network inference using clique tree growth curves. *Artificial Intelligence* 174:984–1006.

Rump, H.; Miyabe, S.; Tsunoo, E.; Ono, N.; and Sagama, S. 2010. Autoregressive mfcc models for genre classification improved by harmonic-percussion separation. In *Proceedings of the 11th international society for music information retrieval conference*, ISMIR '10.

Tzanetakis, G.; Essl, G.; and Cook, P. 2001. Automatic musical genre classification of audio signals. In *Proceedings of the 2nd international society for music information retrieval conference*, ISMIR '01.