Rock or not? This sure does [Category] Audio and Music

Anand Venkatesan, Arjun Parthipan, Lakshmi Manoharan
anand95 arjun777 mlakshmi

Motivation

- Music Genre Classification continues to be an interesting topic for study, given the ever-changing understanding of music genres and the extensive feature set we could build our machine learning models from. Successful implementation of genre classification can lead to more personalized music recommendations and well-defined music generation systems.
- Our objective is to identify an audio clip as belonging to a particular genre and providing song recommendations (from our data set) based on the identified genre.

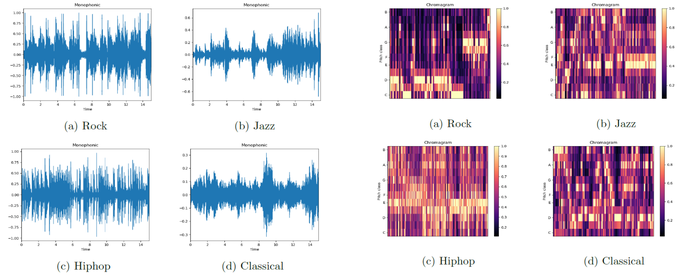

Figure 1. Amplitude Envelope Waveform

Figure 2. Chromagram samples

Dataset and Pre-Processing

- This project utilizes the GTZAN collection from MARSYAS for four genres *viz.* Rock, Jazz, Hiphop and Classical, consisting of 400 tracks in total, 100 for each genre. All tracks are 22050Hz mono 16-bit audio files of 30s duration.
- We identified Mel Frequency Cepstral Coefficients (MFCC), which are traditionally used for speech recognition, to be representative of timbral features in music.
- MFCCs are derived from a cepstral representation of the audio clip.
- With the frequency warping in mel scale, the frequency bands are equally spaced which approximates the human auditory system response more closely than the linearly spaced frequency bands in normal cepstrum.

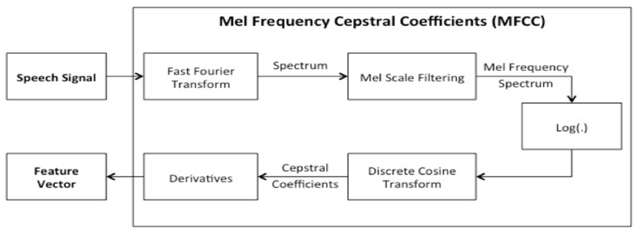

Figure 3. MFCC Flowchart

Distance Metric

- Given the non-scalar nature of attributes obtained from the MFCC processing, we could not use the conventional Euclidean distance metric to measure similarity between song tracks.
- We instead used KL Divergence as distance metric. The KL divergence between two distributions p(x) and q(x) is given by:

$$2KL(p||q) = log\frac{|\Sigma_q|}{|\Sigma_p|} + tr(\Sigma_q^{-1}\Sigma_p) + (\mu_p - \mu_q)^T\Sigma_q^{-1}(\mu_p - \mu_q) - d$$

$$D_{KL}(p,q) = KL(p||q) + KL(q||p) = 2KL(p||q)$$

Classification Methods

We experimented with several supervised and unsupervised techniques for classifying the song tracks:

1) K-Nearest neighbors
2) K-Means clustering
3) Support Vector Machines
4) Neural Network

Three different feature sets:

1) 20 MFCCs for each track
2) 15 MFCCs for each track
3) Mean vector & flattened upper triangular elements of covariance matrix to form 135x1 feature vector*

We also used Principal Component Analysis for Dimensionality Reduction and repeated the experiments.

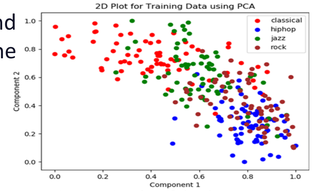

Figure 4. 2D Plot of the data using PCA

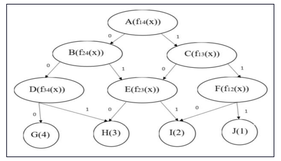

Figure 5. Two-class SVM

Genres	K-NN (MFCC)			Neural Network (Reduced Dimension)		
	Accuracy in %	Recall in %	Standard Deviation	Accuracy in %	Recall in %	Standard Deviation
Rock	78.3	33.33	1.30	73.33	87.5	0.51
Jazz	80	66.67	0.65	80	73.67	0.39
Hiphop	93.33	100	0	93.33	92.3	0.13
Classical	95	93.33	0.13	93.33	91.67	0.13

Plots and Figures

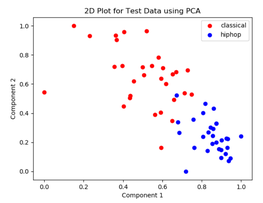

Figure 6a. 2D Plot of for K-Means

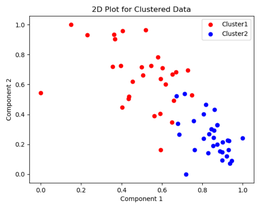

Figure 6b. 2D Plot of for K-Means

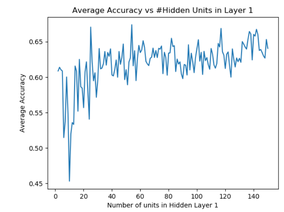

Figure 7. Avg. Accuracy vs #Hidden Units in Layer 1

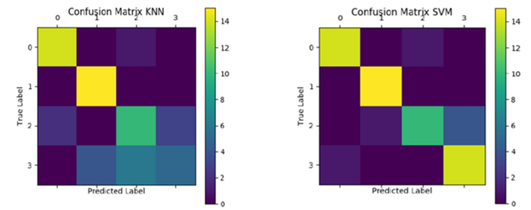

Figure 8. Confusion Matrices for K-NN and SVM (using 15 MFCC feature vector)

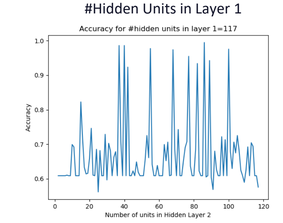

Figure 9. Accuracy vs #Hidden Units in Layer 2

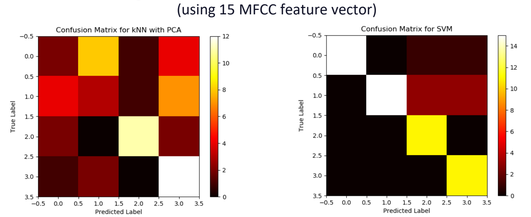

Figure 10. Confusion Matrices for K-NN and SVM (using flattened feature vector)

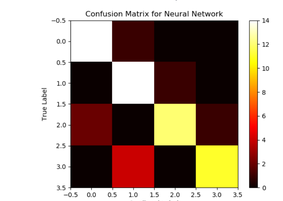

Figure 11. Confusion Matrix for Neural Network

Results and Discussion

We have implemented K-NN, K-Means and poly-kernel SVM using the MFCC feature set with each track represented by a corresponding mean vector and covariance matrix. We also experimented K-NN, SVM and NN Classifiers employing the Euclidian distance, using a flattened feature set described in [*]. Overall, we observed that the 15 MFCC feature vector, coupled with the KL Divergence distance metric, outperforms the other feature sets in all of our experiments. Given the significant overlap among the four genres, the poly-kernel SVM better classified the test set than the classical K-NN and K-means classifier. We have used k-fold cross-validation technique to validate our results. We studied the performance of the K-means classifier using metrics like *Mutual Information Score* (0.89) and *Random Index Score (0.93).* We attempted to build 2-layer deep NN with varying number of neurons in each hidden layer. A (135,124) model yielded consistently good results (~86%) despite the information loss incurred as a result of flattening and further dimensionality reduction using PCA. The recommendation engine, built on top of a K-NN classifier, was able to provide 8 out of 10 similar track recommendations, given a track from a particular genre.

References

[1] Marsyas. "Data Sets" http://marsysas.infodownload/data sets.
[2] De Poli and Prandoni, Sonological Models for Timbre Characterization, http://lcavwww.ep.ch/ prandoni/ documents/timbre2.pdf
[3] Mandel, M., Ellis, D.. Song-Level Features and SVMs for Music Classication http://www.ee.columbia.edu/dpwe/pubs/ismir05svm.pdf.
[4] Chen, P., Liu, S.. An Improved DAGSVM for Multi-class Classication http://ieeexplore.ieee.org/stamp/stamp.jsp? arnumber=0566976.
[5] Pontikakis, Charles Tripp Hochak Hung Manos. "Waveform-Based Musical GenreClassification."
[6] Fu, A., Lu, G., Ting, K.M., Zhang, D.. "A Survey of Audio-Based Music Classification and Annotation" IEEE Transactions on Multimedia. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5664796&tag=1
[7] Bou-Rabee, Ahmed, Keegan Go, and Karanveer Mohan. "Classifying the Subjective: Determining Genre of Music From Lyrics." (2012).