Interpretable Classification and Analysis of Musical Genres

Alex Katopodis

1. Introduction

There is a functionally endless amount of music available today, with Spotify alone boasting well over 70 million songs in its streaming library[4]. With this many options, it's unlikely that you'd like a randomly sampled song. It's thus natural to bucket similar songs, which humans have done with *genres*. Genres are defined to be musical pieces characterized by a particular style, form, or content[1].

Formal definitions of different genres exist, but classification is still not straightforward. Consider rock, defined to be "played on electronically amplified instruments and characterized by a persistent heavily accented beat...", and pop, defined to be"popular music"[1]. "Here Comes the Sun" by the Beatles certainly can fall into either category, yet is considered a rock song. Conversely, Taylor Swift's "Love Story" is deemed pop even though it matches the definition of rock. Incredibly, genres are almost always universally agreed upon despite having overlapping definitions. It suggests that genre is an abstract concept not well conveyed by words. This raises the question of what audio characteristics actually define a musical genre?

This case study aims to shed light on the question through an analysis of waveform audio data. The goal is to see if there are patterns in the audio features of different genres and accurately classify songs. Specifically, we will compare audio features across genres and use interpretable deep learning to find audio clips, called "prototypes" that are most representative of and good at distinguishing their respective genre.

Prior work on this audio dataset usually follows the very different approach of converting waveforms to images and performing blackbox image classification[6]. Other studies have used audio derivatives (such as timbre and tempo) and classic ML to achieve good performance[5], though this particular study seems to lack a validation set and overfits the training data. Our case study is unique in its usage of raw audio data, good model performance on validation and testing data, and good interpretability. Hopefully, this will provide unique insight as to what defines a genre.

The code for this study and can be accessed on my GitHub.

2. Data

The data for this study are from the GTZAN dataset, consisting of songs from 10 different genres. The represented genres are: blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, and rock, all with 100 songs each for a total of 1000. Each song is represented by a 30 second wav file. While GTZAN contains other data like images of the Mel Spectrogram of each song and csv of audio derivatives, these are not directly used in my study. The data were collected between 2000 and 2001 and come from CDs, radio, and microphone recordings.

The wav files are the foundation of this study. Audio features will be computed and analyzed from them, and they will be the inputs for the model. Wav files contain the measured amplitude of the sound wave at the recording device. The songs are roughly the same volume, though some slightly louder than others and reach larger amplitudes. The audio is mono, meaning there is only one channel and recording (as opposed to stereo). The sample rate is 22050Hz, meaning 22050 pressure measurements were taken per second. Given the duration is 30 seconds, the shape of each file is (661500, 1). This is a low sample rate, and thus the recordings cannot catch high frequencies. But, it does save space. The labels were given based on the common consensus for each song.

The audio features I will compute for analysis before the modeling phase are listed here along with their justification.

1. **Zero Crossing Rate:** How often the waveform goes from positive to negative. High values usually imply more percussive sounds and suggest how heavy the beat of a track is. Knowing how percussive a track is might be valuable in distinguishing genre.

2. **Spectral Centroid:** weighted average of the frequencies present in a track. Higher values are usually perceived as "brighter" sounds, and it will be interesting to see if some genres are brighter than others.

3. **Spectral Bandwidth:** width of the band around the spectral centroid that encompasses half of the spectral power. Spectral bandwidth can give insight into how rich vs focused the music sounds. High values suggest

the presence of more complex sounds.

4. **Spectral Flatness:** denotes how noise-like a sound is. High values are more noisy. Certain genres might rely on purer, more tonal sounds, while others more distorted and noisy.

There was not a sufficient amount of data for model training, and thus the training set needed to be artificially increased in size. Cropping audio was not an option since the model prototypes were already short at 1.5 seconds, and shorter inputs would lead to smaller prototypes. Instead, traditional audio processing techniques were used. These included polarity inversion, EQ via high and low pass filters, random noise addition, and pitch shifting. These four transforms were randomly applied 9 times to each of the 700 training examples to artificially create 6300 data points for 7000 total.

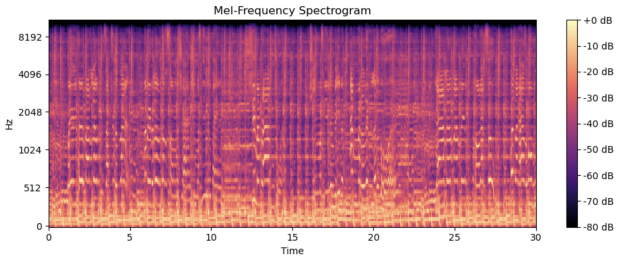

Figure 1. A Mel-Spectrogram of Stevie Ray Vaughan's "Pride and Joy" from the dataset. The power of each frequency at any point in the song is shown.

3. Methodology

As briefly mentioned in the introduction, this study comprises of three parts. First, standard EDA and statistical testing will be used to find patterns in the genre-specific distributions of the audio features mentioned in the data section. Second will be a modeling phase with the sole objective of accurately predicting genre from waveform alone. The model architecture is outlined in the next subsection, but specifically a Prototypical Part Network (ProtoPNet) will be used. This was chosen for its interpretability, which will hopefully allow me to uncover the patterns between genres in the final part of the analysis.

Before modeling, the data were split into a training (70%), validation (15%), and testing set (15%). At no point was the testing set ever observed, including during any EDA, before the final model testing phase. It was randomly set aside before all other coding began.

ProtoPNet[2] uses deep learning to find "prototypes", in our case a 1.5 second trimmed audio clip from one training instance, that are generally representative of a genre. When performing inference, a waveform input will be encoded and compared to all prototypes via cosine similarity. If the similarity is high, we can infer that the audio sounds like the similar prototype. Predictions will be made based on which prototypes the model identifies as being (mostly) present in the input. Intuitively, we can view this as asking what genres the prototypes that sounded similar to our input belonged to.

Each genre will have 10 prototypes that represent a "typical" sound from a certain genre in the training set. I will analyze these prototypes humanly to hopefully uncover some of the more subjective factors leading to genre classification.

3.1. Assessing Distributions of Variables of Interest

Audio processing algorithms will be applied to the wav files to compute the features mentioned in the data section. I do not write the algorithms from scratch, but instead use the librosa python library to calculate them. I will compare the distribution of these variables across genres to see if there are any discernible patterns in the physical wave composition.

To perform this comparison, I will first test for normality (via a Shaprio-Wilk test) and homogeneity of variance (via a Levene's test) for the distributions of the four variables of interest when grouped by genre. If there is evidence to suggest they are normally distributed and exhibit equal variance, I will use an ANOVA test to see if the mean values of the statistics vary between genres. If the tests show non-normal distributions and non-constant variance, I will look at the distributions to see how violated the assumptions appear to be. If the assumptions are strongly violated, I will use a Kruskal-Wallis test, which relaxes the normality and equal variance assumptions.

All hypothesis testing will be carried out at the $\alpha = 0.05$ significance level. If there are differences in means, I will assess the distributions to see what the patterns the data are showing.

3.2. Interpretable Modeling

The ProtoPNet model consists of two components: the backbone network and the prototype fully-connected layer.

The backbone model is a one dimensional convolutional neural network that takes the input from (1, 661500, 1) to (128, 20, 1). This is achieved through three convolutional layers that output 32, 64, and 128 channels, respectively, followed by batch normalization and maxpooling. During backbone training, one fully connected layer was added to the end of the network to simulate a situation similar to the prototype layer it will later be added in front of. The backbone model was trained for 30 epochs on four Nvidia A100

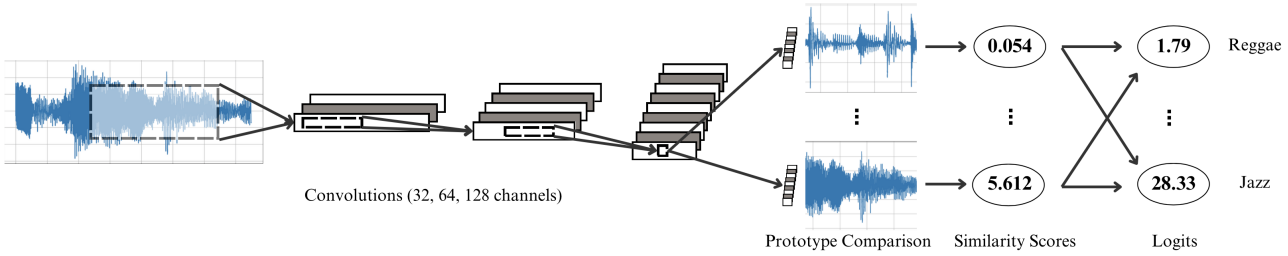

Figure 2. A visual representation of the ProtoPNet used to classify audio.

GPUs. A CNN was chosen for the ease of integration with ProtoPNet and since its embeddings can be easily mapped to regions of influence in the original input. This is necessary for associating embeddings with real prototypes.

The backbone encoder was added onto a ProtoPNet (figure 2) and subsequently trained for 22 epochs on four Nvidia A100 GPUs. The prototype layer learns to find training examples that are representative of a class and use the cosine similarity between the prototypes and input to determine what genre it belongs to. The model is constrained during training to ensure all inputs are similar to at least one prototype in its true class, and dissimilar to all out of class prototypes. 10 prototypes were learned per genre.

3.3. Prototype Analysis

Prototypes learned by the ProtoPNet were saved for further analysis. They were examined "humanly". I personally listened to each prototype to assess if they seemed correct for the genre they were assigned to, and if they sounded similar. Ideally, the prototypes will all appear to be a good fit and encompass a variety of sounds within the genre, but there might be other interesting patterns to pick up on even if this is not the case. Genre is a subjective human made way of categorizing songs, so adding a human element to the analysis will hopefully provide more general insight that isn't immediately clear from the original distribution analysis.

4. Results

4.1. Distributions of Audio Features

Performing a Shapiro-Wilk and Levene test on all distributions suggested that a Kruskal-Wallis test be used for all features. Individual results of those tests are available in appendix 2. However, the normality and equal variance assumptions of ANOVA are somewhat relaxed, and the Shapiro and Levene tests alone do not necessarily gauge the efficacy of using ANOVA. Visually, the distributions actually look fine for performing an ANOVA with the exception of spectral flatness. Thus, I opt use an ANOVA test for spectral centroid, spectral bandwidth, and zero crossing rate, while using a Kruskal-Wallis for the spectral flatness. Their distributions are available in appendix 6.

The ANOVA tests all yielded p-values less than the preset $\alpha = 0.05$. There is sufficient evidence to suggest that the mean differs in at least one genre for spectral centroid ($p < 0.001$), spectral bandwidth ($p < 0.001$), and zero crossing rate ($p < 0.001$). The KW test yielded a p-value also less than 0.05. There is sufficient evidence to suggest that at least one average rank differs in spectral flatness ($p < 0.001$). The distribution of all variables of interest is available in figure 3.

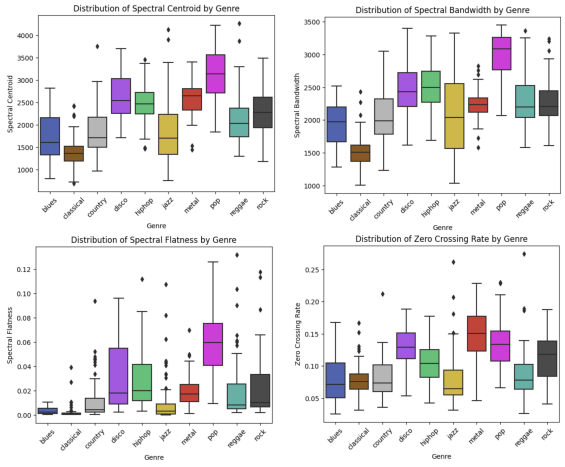

Figure 3. The distributions of all four variables of interest.

4.2. ProtoPNet Performance

After 22 epochs of training, the ProtoPNet achieved 69.54% classification accuracy on the testing data, a near 20% improvement over the backbone, which peaked at 51% accuracy on its own. There is little prior work on this dataset that uses exclusively waveforms and convolutional networks to make predictions, so it's tough to make an accuracy comparison to state of the art models. Other LSTM based models[3] have been trained and achieve better performance at approx. 80% validation accuracy, on average.

The LSTM model in this article was significantly larger and trained much longer than our convolutional backbone. Given the time constraints, it was not possible to explore if a stronger backbone and more training would lead to comparable validation and test performance.

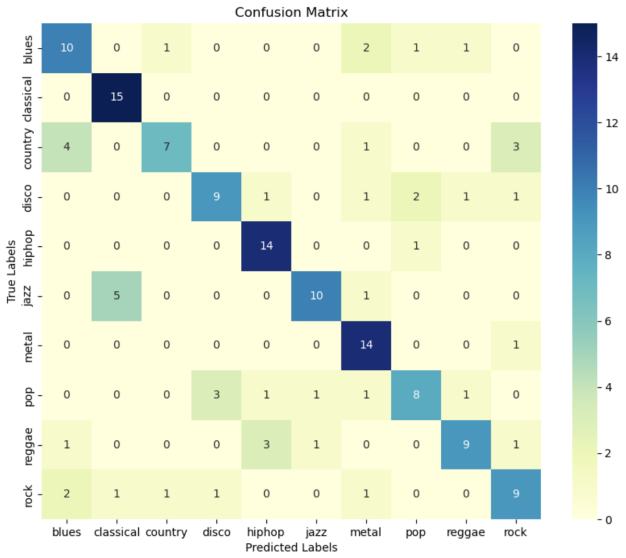

Figure 4. Confusion matrix of genre classifications.

Genre	Sensitivity	Specificity
Blues	0.67	0.59
Classical	1.00	0.71
Country	0.47	0.78
Disco	0.60	0.69
Hip-Hop	0.93	0.74
Jazz	0.63	0.83
Metal	0.93	0.67
Pop	0.53	0.67
Reggae	0.60	0.75
Rock	0.60	0.60

Table 1. ProtoPNet Sensitivity and Specificity by Musical Genre

The model sensitivities and specificities are listed in table 1 and visualized in figure 4. The model did well at picking up on classical, hiphop, and metal genres (all with > 0.9 sensitivity) and was decently specific with country, reggae, and jazz (all with > 0.75 specificity). It was very conservative when predicting country, picking up on less than half of country songs. From the confusion matrix, we can see that the model particularly struggled to discern between classical and jazz as well as blues and country.

4.3. Genre Prototype Analysis

The human analysis performed by listening to the prototypes led to the conclusion that the model was picking up heavily on instrumentation and timbre. Music with similar instruments, like blues/rock and jazz/classical, particularly struggled to be discerned. An in depth look at the classical prototypes supported this theory. Classical music features a variety of instruments, and this was well captured in the learned prototypes. Prototypes 3, 7, and 8 were particularly interesting for this genre, since they appeared to learn different kinds of instruments. Specifically, 3 seems to have learned the timbre of an oboe, 7 the piano, and 8 a low pitched brass or woodwind instrument.

Other genres also alluded to this kind of classification. For example, within the rock genre, the featured instruments varied greatly between all prototypes. Prototype 0 comes from David Bowie's "Space Oddity", which uses unique instruments like a stylophone and mellotron. 1 was dominated by vocals, 3 a brass instrument, 4 a particularly loud bass, 5 a prevalent rhythm guitar, and 9 a unique synthesizer. Even in tracks that featured a blend of the traditional guitar, bass, and percussion instruments, there were still noticeable differences in guitar tone through applied effects.

Visualizing the prototypes with tSNE (figure 5) shows that there is probably good separation of prototypes in the higher dimensional space. Some genres, like metal, classical, reggae, blues, disco, and hip-hop, are tightly clustered together. A few, mainly jazz and country, are a bit more scattered. This implies that the prototypes are doing their job well and finding representative audio that is unique to a specific genre. The tSNE visualization lines up with the results in the confusion matrix.

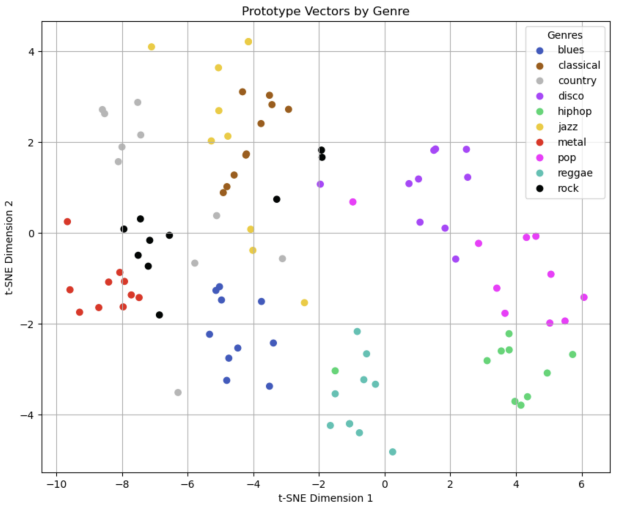

Figure 5. Latent representation of the learned prototypes. The prototype vectors were reduced from 128 dimensions to 2 dimensions via tSNE to visually represent possible separation and clustering.

5. Discussion

The distribution analysis gives several insights into how we might define a genre. The data suggest that there are patterns between spectral features and zero crossing rate across genres. Consider pop. Across the board, pop exhibited high spectral feature values, on average, compared to other genres. These high spectral feature values imply that pop music is likely made of more complex sounds that cover a larger set of frequencies at higher power. The high zero crossing rate is likely indicative of a strong background beat. This is substantially more informative than the dictionary definition of music that is "popular". Given that pop songs consistently exhibited these patterns and were accurately predicted by the model, there is strong evidence to suggest that definition can be improved to encompass these spectral elements.

Classical music, on average, exhibits low spectral features and zero crossing rate values compared to other genres. This suggests it is comprised of simpler and more pure sounds, as well as less percussive. Given that classical instruments are all acoustic and often don't include percussion, this makes sense. While the theory of the music may be rich and complex, it appears that classical music might have simpler timbres and sounds compared to genres that are produced more digitally.

The final ProtoPNet performed well on its test set. Between the prototypes it discovered and the misclassifications it made, it seems to really have honed in on instrumentation. Like the results section pointed out, prototypes seemed to all focus on one instrument in particular. This idea of instrumentation persists when looking at the confusion matrix and sensitivities and specificities. Struggling to discern classical from jazz definitely could be instrument related. From the spectral and zero crossing rate features, we see that jazz and classical are sounding, on average, fundamentally different. There is, however, lots of overlap in instruments. Plenty of jazz music contains piano and brass instruments, as does classical. Blues, country, and rock also all tend to share a similar drum, bass, guitar, and vocals composition, which might explain these misclassifications.

The instruments present in a song are definitely a great first indication of genre. For example, hearing violin in reggae or flute in a metal song would definitely throw a listener off balance. There is some evidence that the model goes deeper than this though. Take metal versus rock. Metal traditionally has the same instruments as rock, but they're played with a different style and effects. The model could possibly be understanding other concepts like bpm and basic scales. It is, however, unlikely that 1.5 second prototypes are able to reliably capture ideas like a chord progression or rhythm. These ideas would need to be further tested on more data.

Ultimately, genre is human can be subjective. There are some songs that even people can't agree on, so it's unlikely we'll ever find perfect definitions. The analysis of spectral features, zero crossing rate, and learned prototypes do paint a clearer picture of what might lead to their classifications, though. Instruments and effects alone are great at discerning most songs. When looking at more fine grained scenarios with instrument overlap, as is the case for jazz and classical, the inclusion of spectral features and zero crossing rate can help further distinguish.

There is still more work that could be done to expand on the findings in this analysis. Plenty of music theory was left out, and often times genre can be discerned from specific chord progressions (as is the case with something like a blues scale). More data and complex modeling would likely be able to pinpoint these ideas better and further add to the ongoing discussion of how we define a genre.

References

[1] Merriam-webster's collegiate dictionary, 2003.

[2] Chaofan Chen, Oscar Li, Chaofan Tao, Alina Jade Barnet, Jonathan Su, and Cynthia Rudin. This looks like that: Deep learning for interpretable image recognition, 2019.

[3] Ramoliya Fenil. Proper eda lstm genre classifier for audio data, 2024.

[4] Tim Ingham. Spotify now hosts 70 million songs. but it can't keep that up forever, 2021.

[5] Ndiatenda Ndou, Ritesh Ajoodha, and Ashwini Jadhav. Music genre classification: A review of deep-learning and traditional machine-learning approaches, 2021.

[6] Nandi Papia. Cnns for audio classification, 2021.

6. Appendix

GitHub link: https://github.com/alexkato29/Music-Genre-Classification

Table 2. Shapiro-Wilk Normality Test P-values across Various Audio Features and Genres

Genre	ZCR	Spectral Centroid	Spectral Bandwidth	Spectral Flatness
Blues	< 0.001	0.0128	0.0343	< 0.001
Classical	0.0019	0.0011	0.0011	< 0.001
Country	< 0.001	< 0.001	0.0799	< 0.001
Disco	0.7966	0.1898	0.0297	< 0.001
Hip-hop	0.3148	0.3760	0.8525	< 0.001
Jazz	< 0.001	< 0.001	0.0907	< 0.001
Metal	0.1998	0.2621	0.0598	< 0.001
Pop	0.2067	0.2858	0.0013	0.5067
Reggae	< 0.001	< 0.001	0.0075	< 0.001
Rock	0.1204	0.9272	0.0119	< 0.001

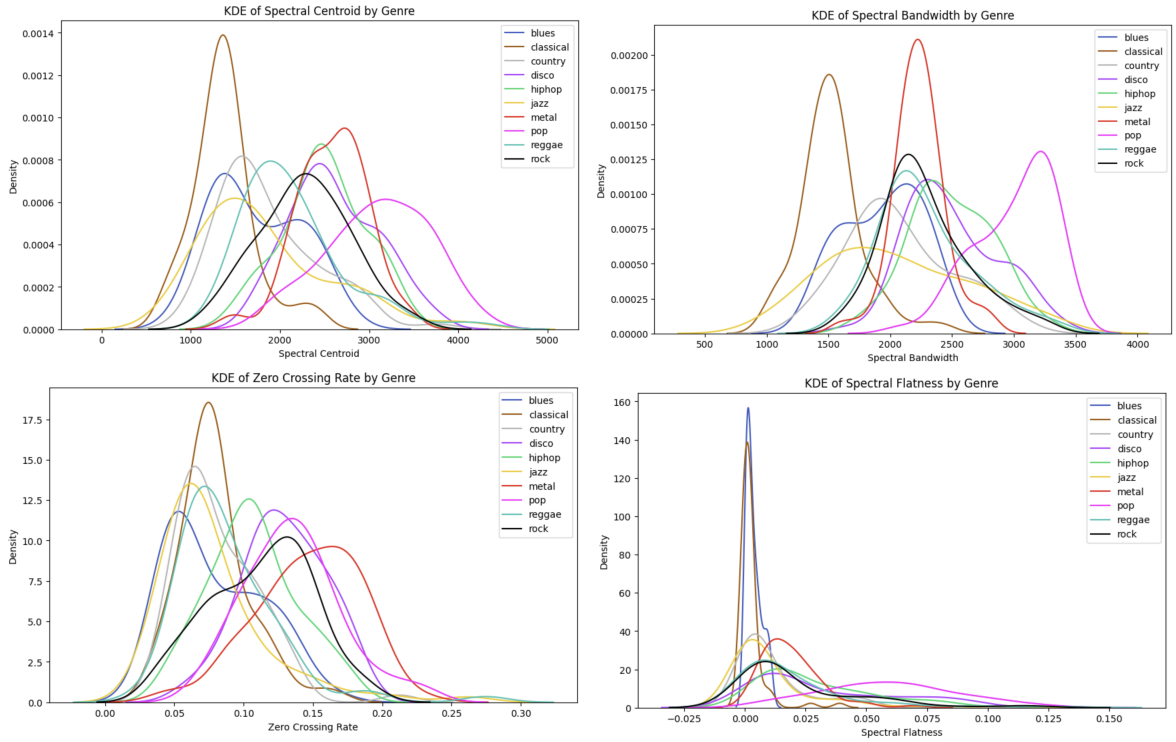

Figure 6. The distributions of all four variables of interest by genre. Spectral centroid, spectral bandwidth, and ZCR look to satisfy the assumptions of ANOVA, while spectral flatness clearly violates equal variance.