Music Genre Classification Using Decision Tree Algorithms

Nathan K……
Data Mining Research Paper
Libraries: Pandas, Numpy, Sci-Kit Learn

Abstract--Recommendation systems are becoming increasingly popular within the streaming world, and part of their success is predicated on algorithms that can accurately determine which genres of music that a user is favorable to. Music has nominal and categorical characteristics that can allow one to deduce the genre if the correct training methods are applied. The goal of this research is to apply various decision tree algorithms to a large dataset of songs to determine which algorithms have the best accuracy and which features are most important for predicting music genres.

Keywords—Music Genre, Decision Trees, Gradient Boosting, Classification, Feature Importance.

I. INTRODUCTION

To facilitate the growth of streaming platforms such as Apple Music and Spotify, music recommendation systems have had to become incredibly accurate at determining what sounds a user may want to hear. In a research paper titled *The Use of Deep Learning-Based Intelligent Music Signal Identification and Generation Technology in National Music Teaching*, the authors note that;

"Music genres have gradually formed based on the emergence of musical instruments and the diversification of music storage methods. Now, the traditional music arrangement and music information retrieval have been gradually replaced by computer technology. Digital audio processing, speech recognition, speech compression coding, and text speech conversion have become increasingly diversified and accurate under the revolution of information technology." [1]

However, deducing the genre of a song from listening to it is not always a trivial task. Songs within a single genre can sound totally different from each other, which could cause confusion when trying to understand what genres of music someone may like. This is why various attributes for sound have been made to assist people with understanding the auditory landscape of specific genres. Some of these attributes include loudness, tempo, and speechiness. These are three features, along with other attributes, that every song possesses and by which the genre of music can be discovered.

Songs within genres can have recognizable characteristics. For example, heavy metal songs may have a higher loudness rating than songs from other genres. On average, dance music may have a higher tempo than music from various genres. After analyzing thousands of songs, patterns can be discovered based on the aforementioned attributes. These patterns can highlight which features are essential for categorizing a song into a specific genre. They can also highlight which features are not important.

The data set used was the Music Genre dataset on Kaggle, which was provided as a CSV file. The data set had 50006 examples, 17 features, and 10 different class values. The algorithms were run and the figures/graphs were plotted on a jupyter-notebook using Pandas, NumPy, and Sci-Kit Learn. I originally planned to complete this project on Weka but was unable to convert

the .csv file to a suitable .arff file and found the plotting and graphs provided by Pandas to be a better fit.

II. METHODOLOGY

A. Data Set

The data set had 50006 examples, 17 features, and 10 class values. It was composed of 50000 songs from various genres. The song titles and artists' names were listed. There was no class imbalance since each value was evenly distributed with 5000 instances. The class values were Electronic, Anime, Jazz, Alternative, Country, Rap, Blues, Rock, Classical, and Hip-Hop. Some of the key features were popularity(measured from 0-99) (tempo (measured in beats per minute), loudness (measured in dB), key (the key in which the song is played), and acousticness (measured on a scale between 0 and 1).

B. Data Preprocessing

There were four duplicated rows that were filled with Nan that were dropped from the dataset. Features were then checked for uniqueness to make sure that the nan value was not present so that certain algorithms could run without error.

The features: index, instance_id, track_name, and obtained date were dropped because they did not correlate with the music genre. Previous runs with decision tree algorithms were ran with those included, and instance_id emerged as the most important feature. The rest of the categorical features were encoded so that the algorithms could run without error. They were correlated to numbers. The mode was coded using one-hot encoding since its values are binary, and the rest were label encoded. The class value, music_genre, were encoded for the gradient boosting algorithm.

Table 1 plots the features against the target (music_genre) by mean. Tempo had the highest values correlated to the target. Table 2 plots the distribution of features

Table 1: Features against Target

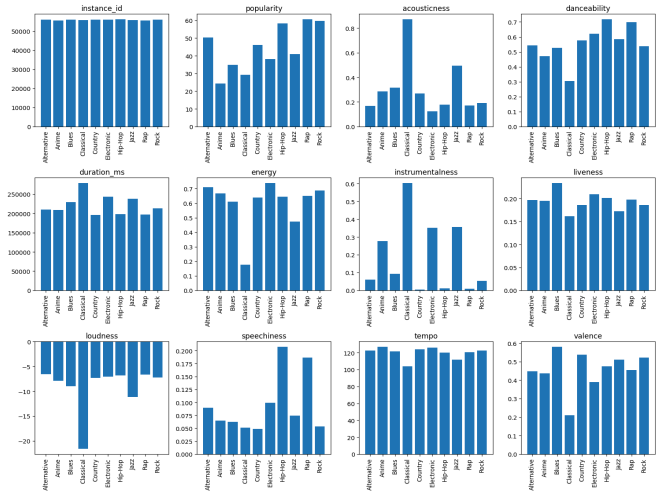
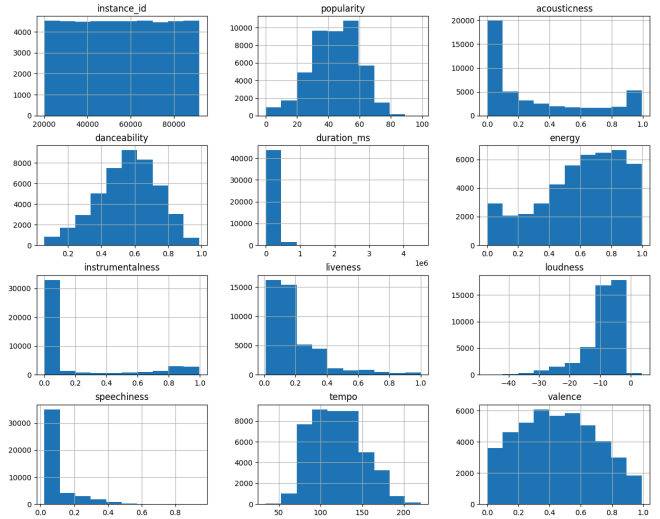


Table 2: Distribution of Features



C. Decision Tree Algorithms

The decision tree classifier, decision stump, random forest, and gradient boosting algorithms were used in this project. The decision tree classifier default for Sci-Kit learn is CART. A useful description for CART mentioned in another paper states that "CART and C4.5 algorithms construct the tree in two phases: growing and pruning, while other ones, as ID3, just execute the growing process…CART algorithm is characterized by the fact that it constructs binary trees, namely each internal node has exactly two outgoing edges. The splitting criteria are the Twoing

criteria, which search for two classes that will make up together more than 50% of the data; the Twoing splitting rule allows us to build more balanced trees, but this algorithm works slower than the Gini rule"[2].

Decision Stump is practically the same as the decision tree classifier except that the max_depth has to be set to 1 to create the stump. The stump consists of the root node and a single level of child nodes. Random Forest is an ensemble method for creating decision trees. Each tree is trained on a random subset of the data, and the final prediction is a combination of the predictions made by individual trees. Gradient Boosting is another ensemble learning technique that combines the predictions of multiple weak learners, which are often times stumps, to create a more predictive model. A weak learner is defined as one whose performance is at least slightly better than random chance. Machine Learning researcher Leslie Valiant notes that "The idea is to use the weak learning method several times to get a succession of hypotheses, each one refocused on the examples that the previous ones found difficult and misclassified. … Note, however, it is not obvious at all how this can be done" [3].

D. Application

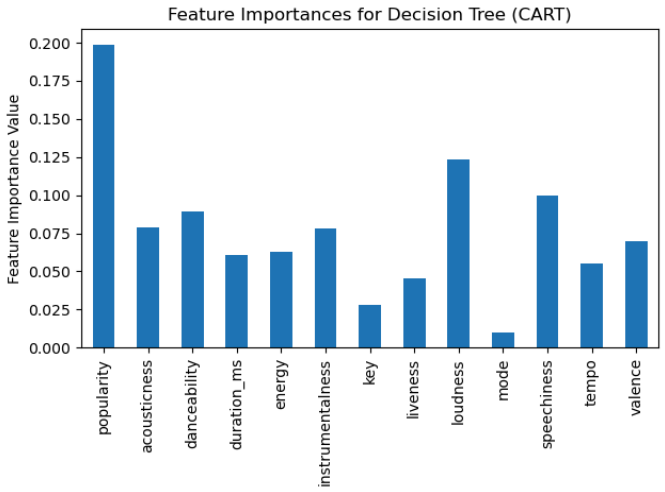
All models were partitioned into a 70 percent training to 30 percent test split. The random state value chosen was 42.
The accuracy, precision, recall, and feature importance were calculated for all of the algorithms. The class value was music_genre whose values were Electronic, Anime, Jazz, Alternative, Country, Rap, Blues, Rock, Classical, and Hip-Hop.

III. RESULTS

Table 1: Evaluation of Decision Tree Classifier

	Evaluation	Score
0	Accuracy	0.43191
1	Precision	0.43131
2	Recall	0.43191

Table 1.1: Feature Importance for Decision Tree Classifier

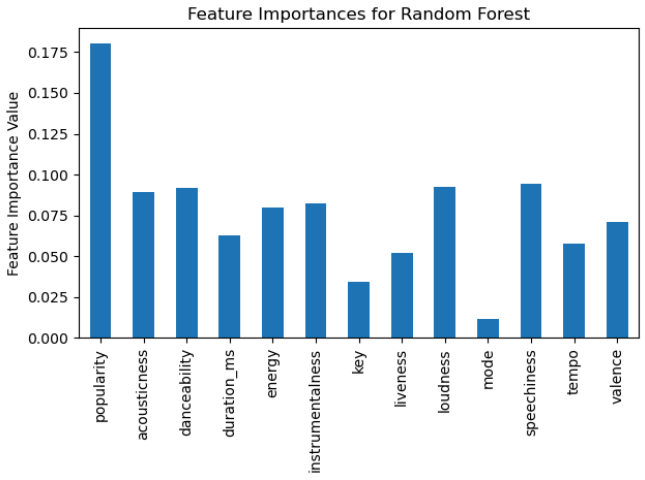


The Decision Tree Classifier was not very accurate with a score of 43 percent. The recall and precision also had scores of 43 percent. Popularity ended up being the most important feature with mode being the lowest.

Table 2: Evaluation of Random Forest Classifier

	Evaluation	Score
0	Accuracy	0.552431
1	Precision	0.554811
2	Recall	0.552431

Table 2.2: Feature Importance of Random Forest Classifier

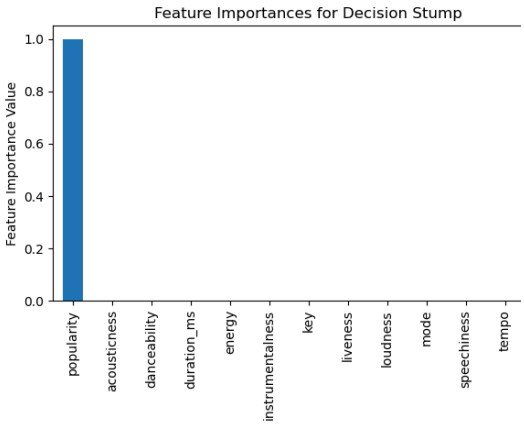


The Random Forest Classifier was not very accurate with a score of 55 percent. The recall and precision also had scores of 55 percent. I was surprised by Random Forest not being the highest or having a much higher score, as in other papers, like *Large-Scale Music Genre Analysis and Classification Using Machine Learning with Apache Spark* by Mousumi Chaudhuty, it is stated that "The random forest classifier manages to achieve 90% accuracy for music genre classification compared to other work in the same domain" [4]. However, it was a significant improvement over the Decision Tree Classifier which was expected. Popularity ended up being the most critical feature with mode being the lowest.

Table 3: Evaluation of Decision Stump Classifier

	Evaluation	Score
0	Accuracy	0.200452
1	Precision	0.041326
2	Recall	0.200452

Table 3.3: Feature Importance of Decision Stump Classifier

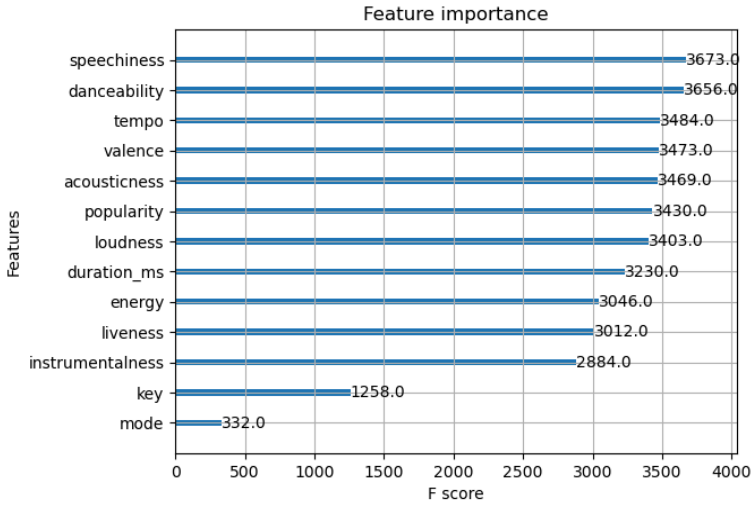


The Decision Stump Classifier was the worst performing classifier very accurate with a score of 20 percent. The recall also had a score of 20 percent but the precision only had a score of four percent. I expected the Decision Stump to be the wordst performing classifier because it is the only classifier which is considered a weak learner. Popularity ended up being the most important feature and all the other features were set to 0. Since the depth of the decision stump was so low, the Precision was ill-defined and being set to 0.0 in labels with no predicted samples.

Table 4: Evaluation of Gradient Boost Classifier

	Evaluation	Score
0	Accuracy	0.571985
1	Precision	0.576737
2	Recall	0.571985

Table 4.4: Feature Importance of Gradient Boost Classifier

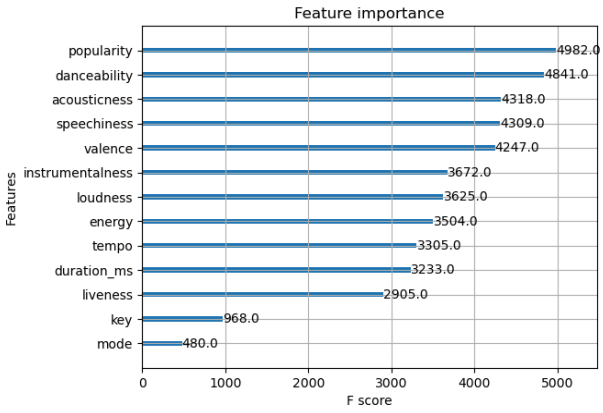


The Gradient Boost Classifier was the best performing classifier at this stage with an accuracy of 57 percent. The recall and precision also had scores of 57 percent. Popularity ended up being the most important feature and with mode being the least important. Due to this classifier having the highest score, I decided to see if I could improve the accuracy and I was successful in doing so.

Table 5: Evaluation of Gradient Boost Classifier

	Evaluation	Score
0	Accuracy	0.769788
1	Precision	0.774093
2	Recall	0.769788

Table 5.5: Feature Importance of Gradient Boost Classifier



The Gradient Boost Classifier was the best performing classifier at this final stage with an accuracy of 76 percent which is a good accuracy score. The recall had a scores of 76 percent and the precision had a score of 77 percent. Popularity ended up being the most important feature and with mode being the least important. I was able to improve the performance of the classifier by using stratified sampling which ensured that the class distribution was protected for both the test and training sets.

IV. CONCLUSION

The best-performing decision tree classifier was the gradient boost, with an accuracy of 76 percent, a recall of 76 percent, and a precision score of 77 percent. All of the decision tree classifiers had popularity as the feature with the highest importance. This was very surprising to me. I thought that the feature with the highest importance would have something to do with how the music sounded.

However, I think that popularity being the most important feature can make sense in a vacuum since it may be the most important feature for current recommendation systems. When you use Spotify, you are usually recommended the most popular songs at the time, regardless of what genre that you listen to most. While the classes were evenly balanced, the popularity measures may not have been balanced across them which means that certain scores may have been highly correlated to certIn genres. 0.5, for example, might have been the most likely score for country music and so if that was the number guessed, then the algorithm could accurately predict that best.

REFERENCES

[1] Tang, H., Zhang, Y., & Zhang, Q. (2022, April 29). *The use of deep learning-based intelligent music signal identification and generation technology in National Music teaching*. Frontiers. https://www.frontiersin.org/articles/10.3389/fpsyg.2022.762402/full

[2] Chaudhury, M., Karami, A., & Ghazanfar, M. A. (2022, August 17). *Large-scale music genre analysis and classification using Machine Learning with apache spark*. MDPI. https://www.mdpi.com/2079-9292/11/16/2567

[3] Khan, F., Tarimer, I., Alwageed, H. S., Karadağ, B. C., Fayaz, M., Abdusalomov, A. B., & Cho, Y.-I. (2022, October 29). *Effect of feature selection on the accuracy of music popularity classification using machine learning algorithms*. MDPI. https://www.mdpi.com/2079-9292/11/21/3518

[4] Valient, L. (2013). *Probably approximately correct: Nature's algorithms for...* Goodreads. https://www.goodreads.com/book/show/16043523-probably-approximately-correct