Wenqing Zhu wz1575 N16257960

Introduction

In this project, our aim is to build a classification model to predict the genre of songs using the provided dataset containing 50,000 songs with various features such as acousticness, danceability, energy, key, and more.

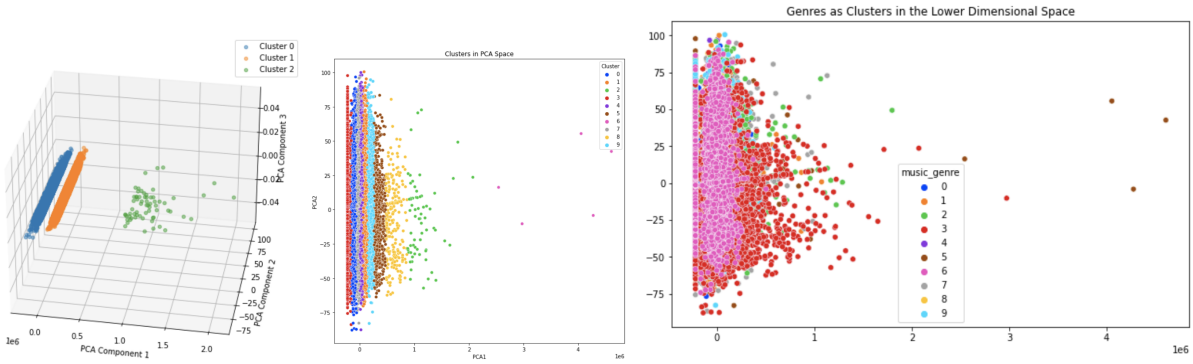
Part 1: Data Cleaning and Preprocessing

I examined the missing values through dropna, given that there are only 5 missing values in each variable, which is a relatively small number compared to the total dataset size, I can drop the missing values without losing much information. In this case, dropping the missing values would be a better choice than imputing the mean, as the impact on the overall data distribution will be minimal.

The challenges I handled included removing missing data, converting column data types, typically from string to numeric values with labelEncoder, and dummying the categorical variables.

Part 2: Dimensionality Reduction and Clustering

I employed dimensionality reduction and clustering techniques to analyze the song dataset. I used PCA to reduce the dimensionality of the audio features and applied the Elbow method to determine the optimal number of clusters. By doing so, I aimed to gain insights into the underlying patterns and similarities within the data, specifically related to the genres of the songs. I visualized the clusters in both 3D and 2D spaces to facilitate a better understanding of how songs with similar audio features were grouped together. This approach allowed for a clearer view of the predictors' clustering and helped in predicting the genre of songs based on their audio characteristics. Besides, I did another dimensionality reduction using PCA after the train test split to reduce the original data with higher dimensions to a lower-dimensional space to have a clearer view of the predictors' clustering.



From the graph, it is very clear that many clusters are overlapped and not well-scattered. Among them, the main clusters are 3 classical and 6 hip-hop.

The challenge I handled was not to normalize the categorical values for dimensional reduction.

Part 3: Train/Test Split

In this code, I performed a train/test split for each genre in the dataset. I created separate train and test data lists, randomly selecting 500 songs for the test set and 4500 songs for the training set for each genre. The train and test data were then combined to create the final training and test datasets.

The purpose of the train/test split is to evaluate the model's performance on unseen data and prevent overfitting. By splitting the data into train and test sets, we can assess how well the model generalizes to new examples. Additionally, performing the split for each genre ensures that the distribution of genres is represented in both the training and test sets, avoiding biases in the model's predictions.

The challenge I handled was to encode the categorical music_genre into numerical.

Part 4: Model Selection and Training

I did a model selection and evaluation using different classification models including SVM, random forest, AdaBoost, and Neural Network, then comparing their performance by evaluating their mean accuracy and standard deviation using 5-fold cross-validation on the training dataset.

I did this because I can then choose the model with better accuracy and consistency. Overall, the Random forest did the best job, so I used it to train the model. Also, since dummy y has 10 variables, I used OneVsRest, which enables multi-class classification.

The challenge I handled is the modeling of multi-class classification through OneVsRest.

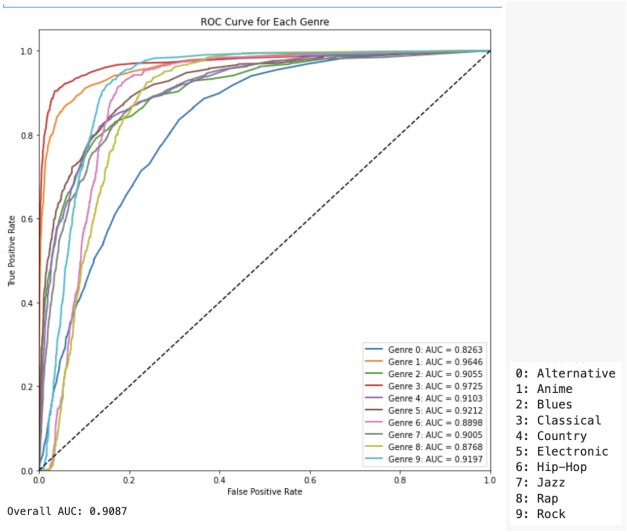
```
SVM: Mean Accuracy = 0.1625, Standard Deviation = 0.0026
Random Forest: Mean Accuracy = 0.5463, Standard Deviation = 0.0007
AdaBoost: Mean Accuracy = 0.4866, Standard Deviation = 0.0062
Neural Network: Mean Accuracy = 0.3311, Standard Deviation = 0.0487
```

Part 5: Model Evaluation

I was evaluating the performance of the model using OneVsRest strategy by calculating the ROC curve and AUC for each genre in the test dataset.

By plotting the ROC curve for each genre and calculating the AUC values, I can visually assess the models performance for each genre.The AUC values provide a measure of how well the model can differentiate between positive and negative instances for each genre.

Finally, I calculated the overall AUC by averaging the AUC values for all genres. This provides an aggregated measure of the model's performance across all genres.



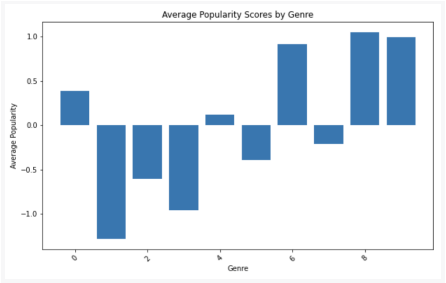
The evaluation of the model using the OneVsRest strategy shows promising performance in predicting the genres of songs. The AUC values range from 0.8263 to 0.9725, indicating good discriminatory power for most genres. Genres 1, 3, and 4 demonstrate particularly strong predictive performance, while genres 0, 6, and 8 show slightly lower AUC values. Overall, the model achieves an average AUC of 0.9087, indicating its effectiveness in distinguishing between different genres.

Part 6: Important Factor

The most important factor that underlies the classification success in this project is the use of dimensionality reduction techniques PCA. By reducing the dimensionality of the feature space while preserving important information, PCA helps to simplify the modeling process, focus on discriminative features, and avoid overfitting. This enables the models to capture the underlying structure and patterns in the data, leading to improved genre classification performance.

Part 7: Extra Credit

By analyzing the dataset, it was observed that certain genres, such as Hip-Hop, tend to have higher popularity scores on average, while genres like Anime have lower popularity scores. This suggests that there is a correlation between genre and the overall popularity of songs within that genre. To visually explore this observation, I drew a bar plot of the average popularity scores for each genre.



It appears that genre 04689 has the highest positive average popularity, indicating that it is generally more popular among listeners. On the other hand, the other genres have negative average popularity, suggesting that they may be less popular or have lower recognition among listeners. Additionally, genre 8 stands out as having the highest positive average popularity among all genres, while genre 1 has the lowest average popularity. This observation provides insights into the varying levels of popularity and listener preferences among different genres in the dataset.