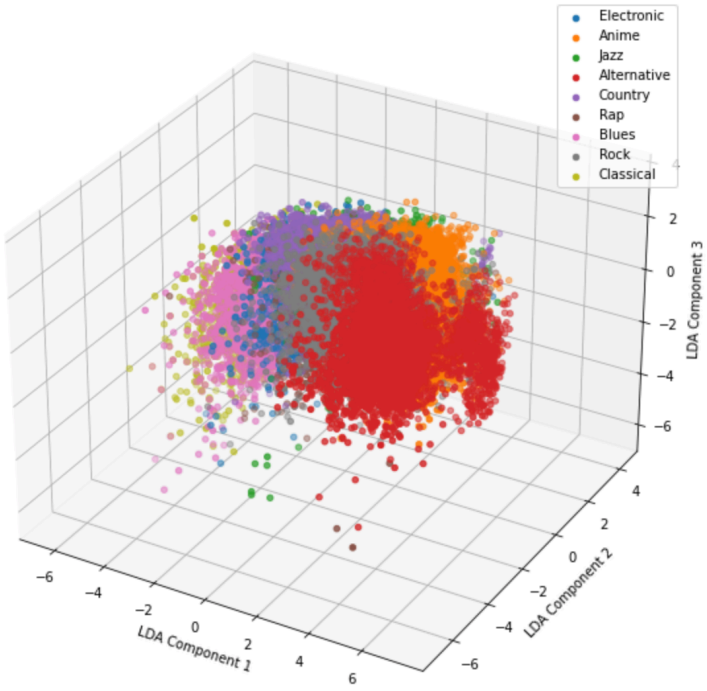
Data Preprocessing & Data Cleaning

The data has 50005 rows of observations. However, after inspecting if the data exist null values, I determined that the data only have 50000 rows of observation that are not null. Therefore, I use dropna() to drop the 5 rows, which is the 10000 to 10004 rows. This results in a distribution 5000 observations per genre. Since there are some durations (-1.0) and tempo (?) of the songs that are missing, I impute the missing value by replacing it with the median of the column as duration and tempo variables are not normally distributed and skewed using median imputation. Tempo variable is initially a column of string making that I have to transform the column to of float to make it useful for computation. For the data to be more useful, some data like categorical data are transformed into numerical data. The encoding is performed by using LabelEncoder(), encoding the music_genre to genre_encoded ranging from 0 to 9, key to key_encoded ranging from 0 to 11 and mode to mode_encoded containing 0 for Major and 1 for Minor. I also drop the "artist_name", "track_name" and "obtained_date" since linguistic properties of artist and song are not so essential on classification. Then, I perform a train_test_split to obtain a test set of 500 and train set of 4500 per genre then appending it into a whole test set containing 5000 observations and train set containing 45000 observations while dropping the genre_encoded of the train set and test set as y_train and y_test and the rest of the test set and train set as X_train and X_test.

Dimensionality Reduction & Clustering

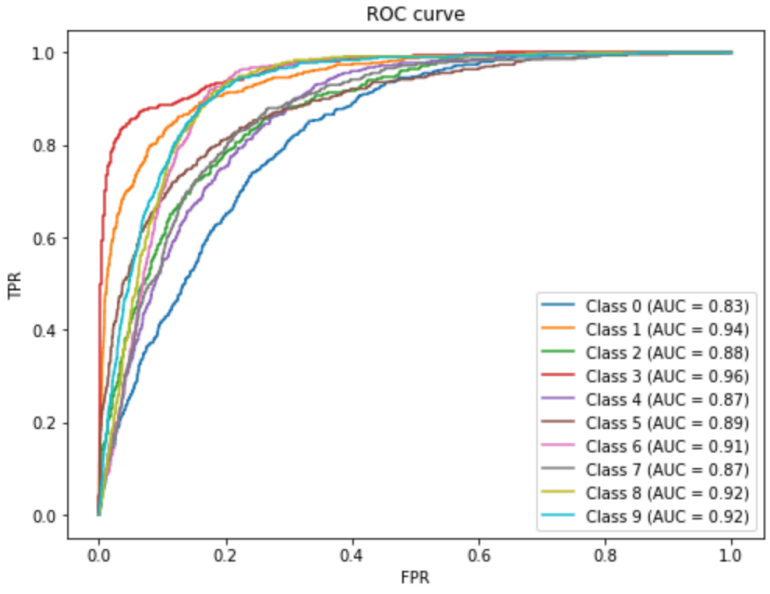
I performed linear discriminant analysis (LDA) for dimensionality reduction with parameter of n_component = 3. Before this, I normalize the non-categorical predictors with StandardScaler to ensure similarity in scale and magnitude. Then, I transform the X_train and X_test data with the fitted lda resulting in X_lda_train and X_lda_test. I plot the 3D representation of LDA as below:



LDA is chosen among the dimensionality reduction method as LDA is a supervised dimensionality reduction technique that maximize separation between classes by projecting data onto a lower-dimensional space, in this case a 3D subspace. It also takes into account of class labels to determine the best projection that separates different clusters. From the 3D representation, it is suggested that there are no clear separation between clusters, there exist some overlapping. By plotting the lda coefficients into heatmap, I determined that the most important feature that underlies the success of classification is the popularity of the tracks. This is possibly the best reflection on the preferences of the general population on music and popular tracks tend to be listened to more frequently making more representative of the general characteristics of a genre.

Classification Model

I build a classification model choosing between AdaBoost model and Random Forest model. I used the transformed in dimensionality reduction, X_lda_train and X_lda_test to replace the original X_train and X_test that we usually use in classification. This reduce the effect of the "curse of dimensionality" which improve model's ability to generalize to new, unseen data. I also did a comparison between training the models with original X_train and training with X_lda_train. The final decision of classification model is Random Forest model with max_depth of 10, criterion of entropy and n_job of 5. This combination of hyperparameters yield the best result under the comparison between the models that uses X_lda_train, which gives an AUC score of 0.89876. The calculation of AUC score is determined with parameter of average = "weighted" and multi_class = "ovo", where it calculates the weighted average of the AUC score which gives a more balanced measure, using a one-vs-one approach as the one-vs-one approach has the advantage of producing a more accurate predictions. Below is the plot displaying the ROC curves:



AUC score: 0.89876

Extra Credit

It is observed that the AUC score of the models that use the original X_train is higher than the AUC score of the models that use the X_lda_train. This is due to the reason that X_train contains more information than X_lda_train, there are informational loss in the process of dimensionality reduction.