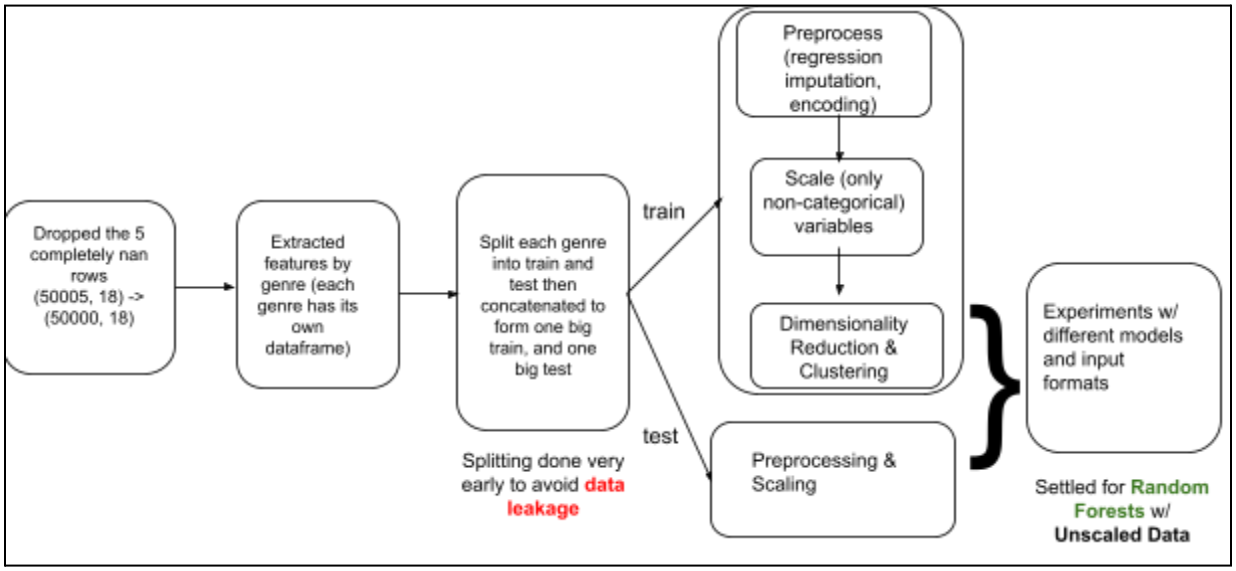
Tomisin Adeyemi
FML Capstone Project



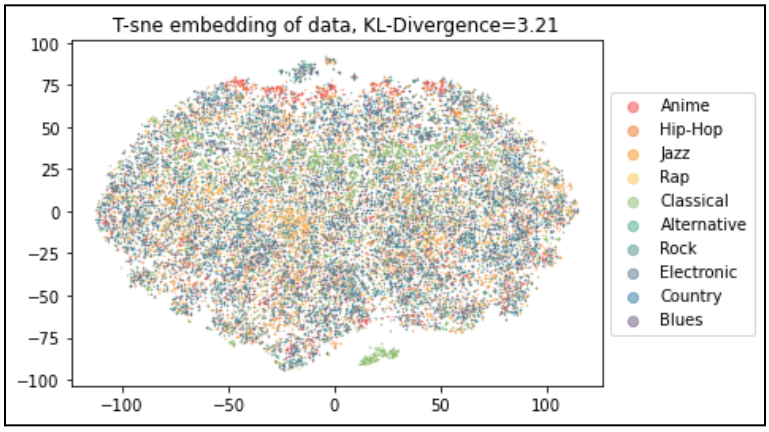
Preprocessing

The diagram above shows the workflow I used for this project. The data was split into training and tested before any imputation or scaling to avoid data leakage. Any form of imputation (mean, regression, mode, etc) would have taken the test data into account before computing the mean, mode, etc. I decided to impute data for both sets separately. I also decided to do the scaling separately for similar reasons.

To impute the missing values in the duration, tempo & instrumentalness columns, I tried a bunch of imputation methods and decided to perform regression imputation. When I imputed the mean, median or mode, the distribution of these columns either became more skewed or exhibited large spikes, thus I chose to build a regression model for each of these columns (based on the non-missing data) that predicts the missing values.

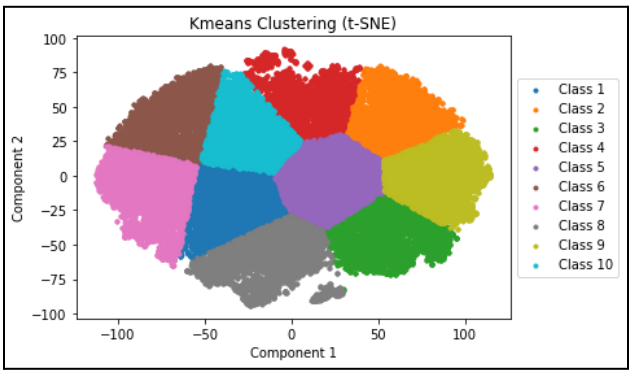
Other preprocessing techniques that were incorporated included dropping the id, artist name, song name and date obtained columns: I decided these columns would either not be useful for any analysis, or make my analysis too complicated. I also encoded the mode column, representing 'Major' as 1, 'Minor' as 0. I also decided to Label Encode the Key column, as opposed to dummy encoding, as the number of unique keys (12) may have incorporated unnecessarily high dimensionality to the dataset. For scaling, I scaled all but the categorical columns (mode, key, genre) to make sure the information in these columns is not lost.

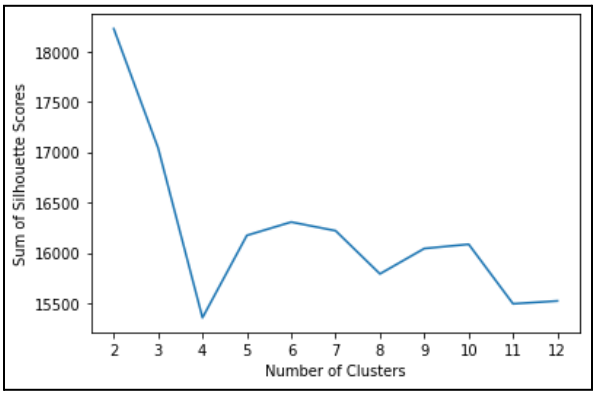
Dimensionality Reduction & Clustering



I experimented with different dimension reduction techniques, but none seemed to be able to clearly separate the data in the lower dimension. I decided to visualize my t-SNE embedding because it looks the prettiest. The lack of separation in the genres is emphasized by the KL Divergence score of 3.21: the distributions in the higher dimensional space and lower dimensional space are very different. Increasing the

perplexity did not help much either. I also attempted clustering the t-SNE embeddings, as suggested by the instructions, but the clusters are not representative of the original dataset. In addition, plotting a silhouette sum graph showed that 10 was not the optimal number of clusters.

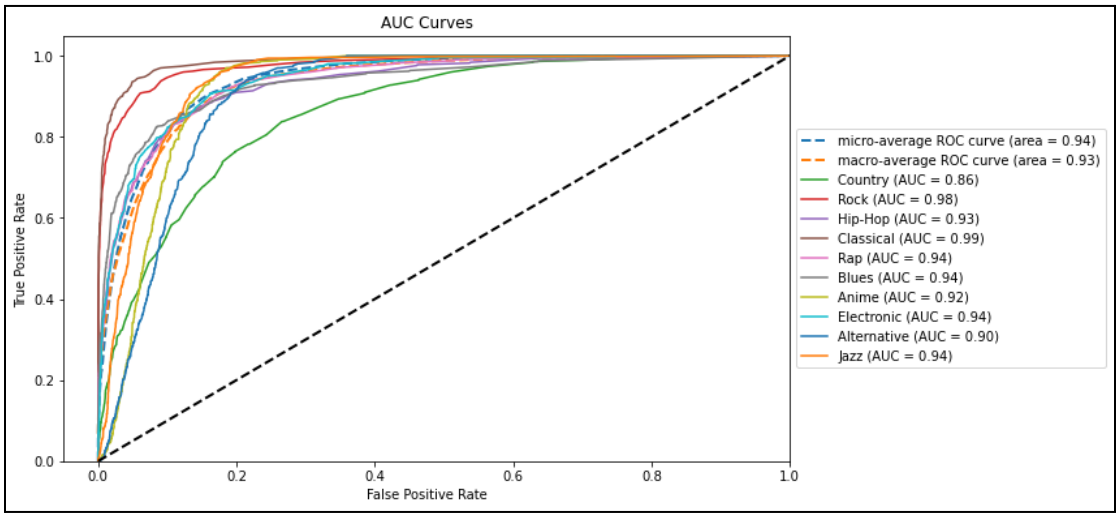




These results are not very surprising, as an avid lover of music and user of Spotify, I can attest to the fact that separating music into genres is not a very straightforward task, especially only using 2 dimensions. This goes to show that a lower dimensional embedding may not be helpful for this classification task, as too much information would be lost and classifiers may not work optimally.

Classification

I decided to use a Random Forest model, as the lower dimensional embedding showed that this is a relatively complex dataset. Thus the strong learner nature of Random forests is good for this task. I tried 2 things: using the scaled data, and using the raw, unscaled data. I achieved a higher AUC with the latter, so I decided to use these raw values in my classification. I label encoded the genres, and used the one vs rest instead of the one vs one multiclass classifier for easier classification. The plot of the AUC is shown below:



Highest AUC: Micro-Auc = 0.94

This is scarily high, but given that the dataset is balanced and any form of data leakage was avoided, it shows that the Random Forests was able to pick up on complex nuances that lower dimensional embeddings cannot. Trying to improve the AUC at this point would have probably led to overfitting/leakage, which I did my best to avoid.

Extra credit

For Extra credit I decided to build a Logistic Regression model to predict songs that are above/below the median duration (in minutes), then examine the model coefficients to see if any non-trivial variables really stuck out for this classification.

For more context, given a coefficient for a continuous variable b, $e^b - 1$gives how much of the odds of the outcome (this case popularity) change for each 1 unit change in the predictor. The largest *seeming* coefficient is speechiness, e^(-4.11) = 0.016, and 0.016 - 1 = -0.984, suggesting that each unit increase in speechiness reduces the odds of having a longer duration than the median by **98.4%!!** This seems a bit counterintuitive: one would expect that songs with more words would be longer, but this is the opposite.