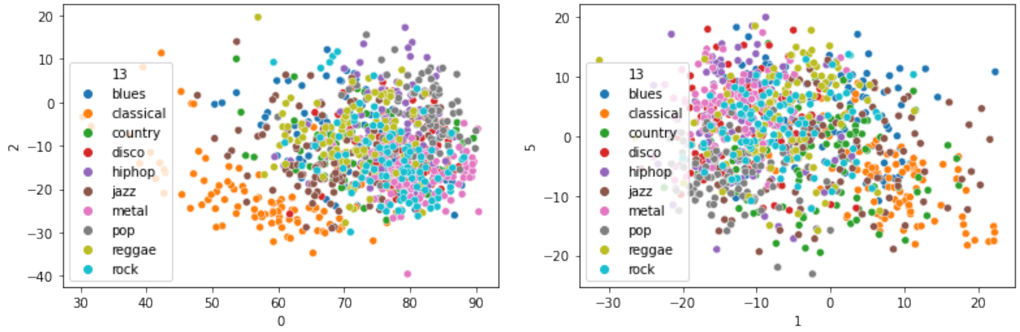
Music Genre Classification

Yitong Shan
Alex Ziyu Jiang

Inspired by the sheer amount of music genres in Spotify, we successfully use Python to build a model for processing audio data and predicting its genre. Our project mainly focuses on using K-Nearest Neighbor, a non-parametric classification method to estimate the genre of our audio data. We have a dataset of 1000 songs, 100 songs for each of the 10 genres that we already know. We first divide each song into several small sections, and extract 13 features from each section using Mel Frequency Cepstral Coefficient (MFCC) method, and calculate the mean and covariance of these features, the results represent the features of the song. Then we use the feature for each song based on the Normal assumptions to estimate the mean and covariance, and modify KL divergence. The resulted formula is used to calculate the distance between two songs (distributions). The following formula is used to calculate KL divergence between two multivariate Gaussian distributions:

$$D_{KL}(N_0||N_1) = \frac{1}{2}\left(\mathrm{tr}(\Sigma_1^{-1}\Sigma_0) + (\mu_1 - \mu_0)^T\Sigma_1^{-1}(\mu_1 - \mu_0) - \mathrm{k} + \ln(\frac{\det\Sigma_1}{\det\Sigma_0})\right)$$

After the step of generating features for each song, we do some data explorations. we plot the distribution of songs with feature pairs as parameters to see if there is any useful information, and we find feature 0 and 1 (features are labeled 0 12) cluster songs with the highest distinctions. Which means they are the key features to distinguish genres. We can see from the graphs below, the orange dots are far apart from other dots, which suggests that the classical genre could be very different from other music genres.

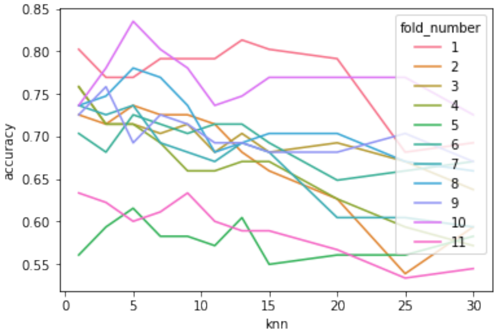


After splitting training and testing sets, we build the prediction model using K-Nearest Neighbors (KNN) method. KNN methid makes predictions based on the K nearest instances in the training data. To find the neighbors, we need to find the distance between the training data and other data points.

By choosing different values of K, we might get different predictions accuracies; Our goal is to find the K value with the highest prediction accuracy. We initially find out when K = 3 and 5, the prediction accuracy is pretty high (∼70%) comparing to other K values. But due to the randomness of splitting training and testing sets every time we run the code, we want to stabilize our result. Therefore, we use K-Fold Cross Validation.

We divide the dataset using K-Fold: hold out 20% of the data as the test set, and the rest 80% are training sets. We choose to generate k = 11 folds. Each time we repeat the splitting process and generate the model, one of the 11 subsets is hold out as the validation set, and the other 10 subsets are put together to form a training set. The KNN classification is repeated 11 times on the training set, so each time the training set is different. We stabilizes our result as it does not matter how we divide the dataset.

After looping through the dataset using K-Fold, we result in a dataframe with K values of KNN and the accuracy. The following graph also visually shows the trend of the accuracy from k=1 t0 k=30 for the 11 folds.



We then average the accuracies group by K values, and sort the accuracies of each k from highest to lowest. We get the following results:

```
     knn  accuracy
2    5.0  0.719880
0    1.0  0.715917
1    3.0  0.710911
3    7.0  0.709901
4    9.0  0.701931
6   13.0  0.690898
5   11.0  0.682917
7   15.0  0.679909
8   20.0  0.660906
9   25.0  0.634898
10  30.0  0.630914
```

We get the result that when k=5, the prediction accuracy is the highest. It indicates when we choose 5 neighbors to classify a specific data, the prediction can be the most accurate and most likely to be its actual class. Since we know k=5 gives the highest accuracy for our model, we want to find out which genre can be the easiest to predict, i.e. the genre that has the highest prediction accuracy. We first split the dataset into testing and training sets, and then divide the training set into 10 training sets according to genre, and then we put each training set to the model to test the accuracy. We get the following results:

```
       genre  knn  accuracy
1  classical  5.0  0.944444
7        pop  5.0  0.900000
4     hiphop  5.0  0.866667
2    country  5.0  0.750000
5       jazz  5.0  0.714286
6      metal  5.0  0.705882
8     reggae  5.0  0.687500
3      disco  5.0  0.631579
0      blues  5.0  0.578947
9       rock  5.0  0.461538
```

We see the 'classical' genre gives the highest accuracy, and the 'rock' genre has the lowest prediction accuracy. It confirms our guesses from the beginning.

In conclusion, by using K-Fold Cross Validation method, under KNN model, k=5 gives us the highest prediction accuracy of over 72%. Also, among the 10 genres, classical genre has the highest accuracy. We are pretty satisfied with out model because we tested some other audios, and we get pretty accurate results. Overall we meet our expectation from the beginning of the quarter. :)

(Thanks to SPA DRP for providing me with this meaningful opportunity, thanks to all mentors for being so supportive. AND I WANT TO THANK ALEX THE MOST, without Alex be patiently helping me and always so encouraging when I had trouble, I would never make it lmao. Thanks for being such a nice friend, a good mentor and a eating buddy.)