Music Genre Classification with Spotify Data

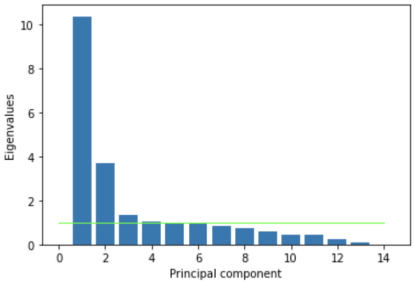
Data Cleaning

After importing the data (musicData.csv), the first thing I dealt with was null values. I found that 5/50005 rows had null values so for simplicity, I decided to drop those rows. I then went on to also drop the instance_id, artist_name, track_name, and obtained_date columns since I decided that these columns would not be useful predictor variables for music genre. Then, since key was in string format, I changed it into numerical values – I changed it based on the number of sharps and flats in a key (I considered keys "positive" and flats "negative" ex. if a key has neither, it equals 0, if a key has 1 sharp, it equals +1, and if a key has 1 flat, it equals -1). Then, since mode is in categorical format (2 possible values – Major or Minor), I dummy coded the column. Finally, before splitting my data into train and test sets, I changed the category labels of the genres into numerical labels (values 0-9) to make it usable for the models later on.

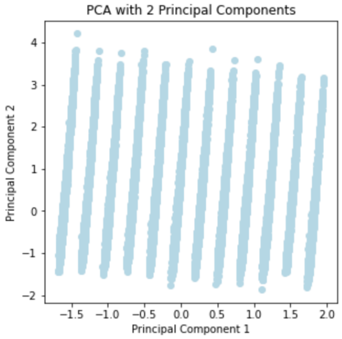
I then split the the data into train and test sets – for each genre, I used 500 randomly picked songs in the test set and used the remaining 4500 songs in the train set – this resulted in 5000 songs in the test set and 45000 songs in the train set. Then, since I saw earlier that when looking at the value counts for duration_ms, there were values for this column that were less than or equal to 0 (which is not possible because a song cannot last negative or zero seconds – it must play for at least some time), I fixed it by imputing the median song duration (227406.5 milliseconds) in the train set without these nonpositive values in place of of these nonpositive values in both the train and test set. Furthermore, since I also saw earlier that when looking at the value counts for tempo, there were 4980 values with no numerical value (had a '?' in place), I fixed it by imputing the median tempo (119.90100000000001 beats) in the train set without '?' values in place of of these '?' values in both the train and test set. Finally, I used StandardScaler to standardize non-categorical predictor variables.

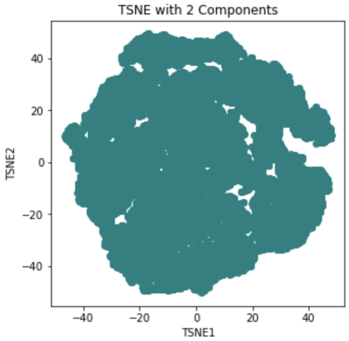
Dimensionality Reduction and Clustering

When running an initial Principal Component Analysis (PCA) on the data and looking at the eigenvalues for each component, I saw that 4 components had eigenvalues greater than 1 (the Kaiser criterion line). Based on this, I calculated that the first component explains 47.601321% of the variance in the data, the second component explains 16.858273% of the variance in the data, the third component explains 6.16436% of the variance in the data, and the fourth component explains 4.837013% of the variance in the data. Since these 4 components collectively explain over 75% of the variance in the data, I ran a PCA with n_components set to 4, fitting and transforming the train set and transforming the test set based on this PCA.

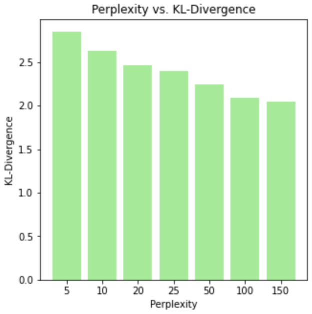


I went on to plot the PCA with 2 principal components but, as seen below, the scatterplot did not show any significant relationship. I also plotted the t-SNE with 2 components but, as seen below, the scatterplot also did not show any significant relationship.

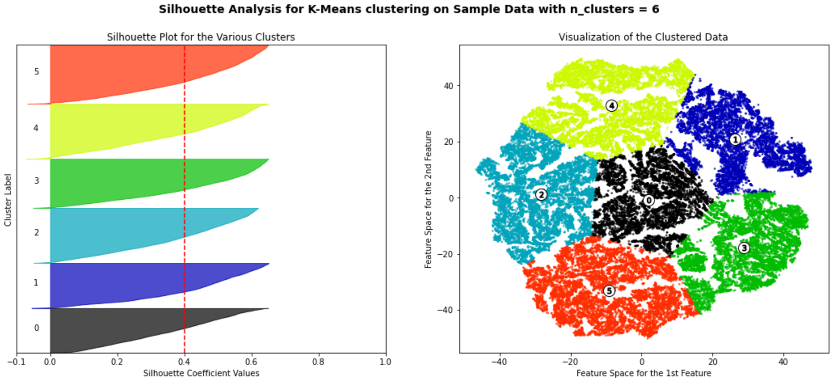


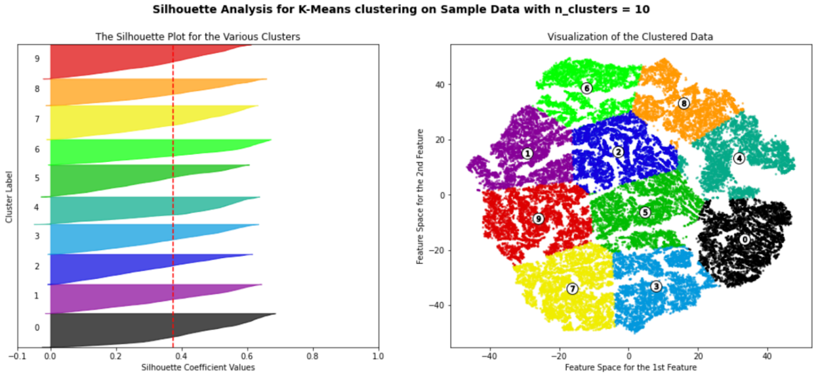


Furthermore, I tried varying the perplexity parameter for t-SNE to see if if it would change the results – as seen by the bar graph below, however, I could see that even when changing the perplexity (from values ranging from 5 to 150), the KL-Divergence doesn't vary much (only slightly decreases as perplexity increases).



For clustering, I looked at various cluster numbers (2-9 clusters) to do K-Means clustering and for each cluster value, calculated the average silhouette score, displayed the silhouette plot, and generated a visualization of the clustered data. I found that the optimal number of clusters is 6 as it has the highest average silhouette score at 0.40001607. Finally, since I know there should be 10 clusters (since there are 10 music genres), I did K-Means clustering with 10 clusters and found the average silhouette score, displayed the silhouette plot, and generated a visualization of the clustered data. I found that 10 clusters had an average silhouette score of 0.3748851, which is not that high and lower that when the number of clusters is 6.

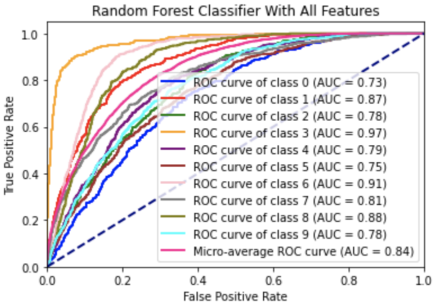




Classification

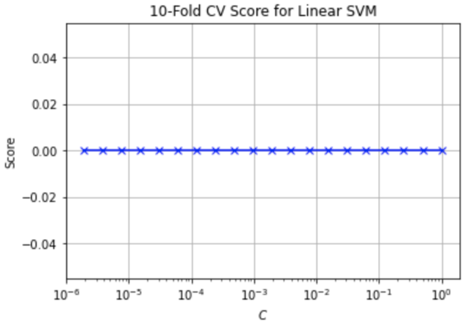
For the classification part, I chose to create Random Forest, a Support Vector Machine, AdaBoost, and a Feedforward Neural Network Classifiers and compare the results of all these models. For all the models, I used the cleaned predictor variables from the dataset that have gone through Principal Component Analysis and clustering as inputs and music genre (as numerical values) as output.

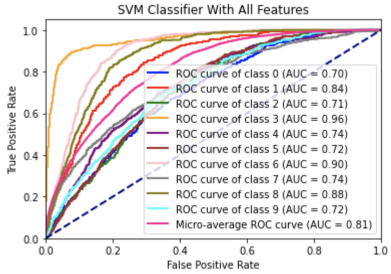
For the Random Forst Classifier, I first tuned the hyperparameters using Grid Search, finding that the ideal parameters were when criterion was entropy, max_depth was 10, max_features was sqrt, min_samples_split was 10, and n_estimators was 200. Using these parameters, I trained a Random Forest Classifier, running the predictions on the test set to calculate my accuracy. I got an accuracy of 36.220% which is quite low. I then plotted a graph to show the ROC curves of all 10 classes as well as the micro-average ROC curve, calculating the AUC for each class as well as the micro-average AUC (the micro-average ROC and AUC calculation summarizes the performance by aggregating the true positive rate and false positive rate across all classes). The micro-average AUC is 0.84427 which means that the model's predictive power is quite good.



```
Random Forest Classifier Accuracy = 36.220%
Random Forest Classifier Micro-average AUC = 0.84427
```

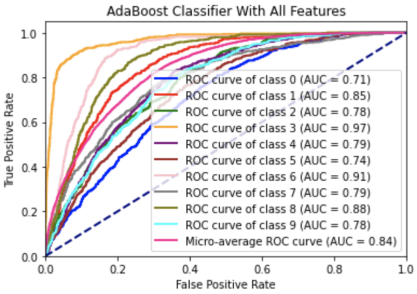
For the Support Vector Machine (SVM), I first found the optimal C value that will be used as a hyperparameter in our SVM model through using a for-loop that looped over different C values ranging from 1 to 1e-6 and identified the C-value that resulted in the highest validation score. I found the optimal C value to be 1 so I used that for my model. Using this parameter, I trained an SVM, running the predictions on the test set to calculate my accuracy. I got an accuracy of 34.040% which is quite low. I the nplotted a graph to show the ROC curves of all 10 classes as well as the micro-average ROC curve, calculating the AUC for each class as well as the micro-average AUC). The micro-average AUC is 0.81071 which means that the model's predictive power is quite good.





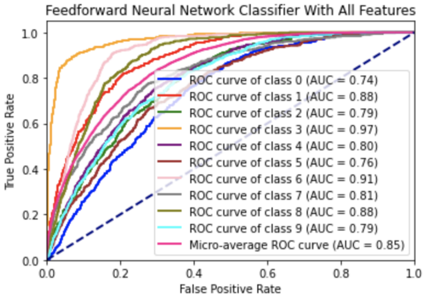
```
SVM Classifier Accuracy = 34.040%
SVM Classifier Micro-average AUC = 0.81071
```

For the AdaBoost Classifier, I first tuned the hyperparameters using Grid Search, finding the ideal parameters were when algorithm was SAMME, base_estimator__max_depth was 6, learning_rate was 0.1, and n_estimators was 200. Using these parameters, I trained an AdaBoost Classifier, running the predictions on the test set to calculate my accuracy. I got an accuracy of 36.620% which is quite low. I then plotted a graph to show the ROC curves of all 10 classes as well as the micro-average ROC curve, calculating the AUC for each class as well as the micro-average AUC. The micro-average AUC is 0.83753 which means that the model's predictive power is quite good.



```
AdaBoost Classifier Accuracy = 36.620%
AdaBoost Classifier Micro-average AUC = 0.83753
```

For the Feedforward Neural Network, I first tuned the hyperparameters using Grid Search, finding the ideal parameters were when activation was logistic, hidden_layer_sizes was (50, 50) (so two hidden layers with 50 neurons each), learning_rate was constant and learning_rate_init was 0.001. Using these parameters, I trained an MLP Classifier, running the predictions on the test set to calculate my accuracy. I got an accuracy of 36.620% which is quite low. I then plotted a graph to show the ROC curves of all 10 classes as well as the micro-average ROC curve, calculating the AUC for each class as well as the micro-average AUC. The micro-average AUC is 0.84776 which means that the model's predictive power is quite good.

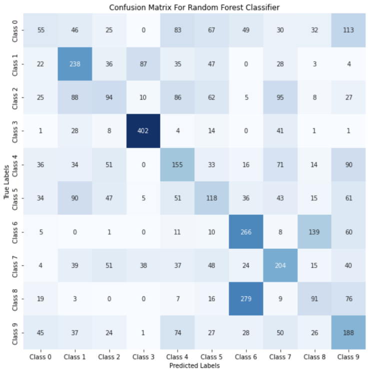


```
Feedforward Neural Network Classifier Accuracy = 36.620%
Feedforward Neural Network Classifier Micro-average AUC = 0.84776
```

Out of these four models, the Feedforward Neural Network model does the best job at predicting music genres of Spotify songs given it having the highest accuracy at 36.620% and the highest micro-average AUC at 0.84776. While my classification wasn't as successful as I had hoped, I think what made it the best it could be was the extensive preprocessing (data cleaning, PCA, clustering) and my extensive hyperparameter tuning for each model.

Model Performance Evaluation (Extra Credit)

When looking at the confusion matrix for the Random Forest Classifier, we can see that the most correctly classified class was Class 3 (which is Classical music) with 402/500 correct classifications. The most incorrectly classified classes were Class 0 with 55/500 correct classifications, Class 2 with 94/500 correct classifications and Class 8 with 91/500 correct classifications – these music genres are Alternative, Blues, and Rap respectively. When looking at the overall predictions, I saw that Class 6 (Hip Hop) was the most predicted class with 703 predictions and Class 0 (Alternative) was the least predicted class with 246 predictions.

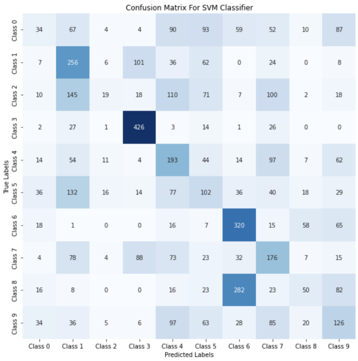


Confusion Matrix For Random Forest Classifier

	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9
Class 0	55	46	25	0	83	67	49	30	32	113
Class 1	22	238	36	87	35	47	0	28	3	4
Class 2	25	88	94	10	86	62	5	95	8	27
Class 3	1	28	8	402	4	14	0	41	1	1
Class 4	36	34	51	0	155	33	16	71	14	90
Class 5	34	90	47	5	51	118	36	43	15	61
Class 6	5	0	1	0	11	10	266	8	139	60
Class 7	4	39	51	38	37	48	24	204	15	40
Class 8	19	3	0	0	7	16	279	9	91	76
Class 9	45	37	24	1	74	27	28	50	26	188

```
Predicted Values
0: 246
1: 603
2: 337
3: 543
4: 543
5: 442
6: 703
7: 579
8: 344
9: 660
```

When looking at the confusion matrix for the SVM Classifier, we can see that the most correctly classified class was Class 3 (which is Classical music) with 426/500 correct classifications. The most incorrectly classified classes were Class 0 with 34/500 correct classifications, Class 2 with 19/500 correct classifications and Class 8 with 50/500 correct classifications – these music genres are Alternative, Blues, and Rap respectively. When looking at the overall predictions, I saw that Class 6 (Hip Hop) was the most predicted class with 779 predictions and Class 2 (Blues) was the least predicted class with 66 predictions.

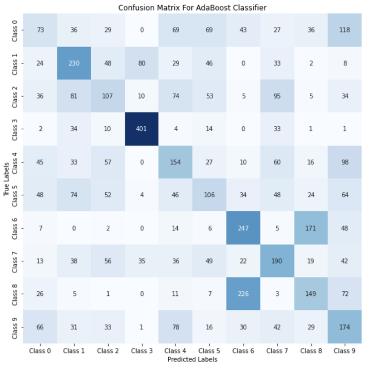


Confusion Matrix For SVM Classifier

	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9
Class 0	34	67	4	4	90	93	59	52	10	87
Class 1	7	256	6	101	36	62	0	24	0	8
Class 2	10	145	19	18	110	71	7	100	2	18
Class 3	2	27	1	426	3	14	1	26	0	0
Class 4	14	54	11	4	193	44	14	97	7	62
Class 5	36	132	16	14	77	102	36	40	18	29
Class 6	18	1	0	0	16	7	320	15	58	65
Class 7	4	78	4	88	73	23	32	176	7	15
Class 8	16	8	0	0	16	23	282	23	50	82
Class 9	34	36	5	6	97	63	28	85	20	126

```
Predicted Values
0: 175
1: 804
2: 66
3: 661
4: 711
5: 502
6: 779
7: 638
8: 172
9: 492
```

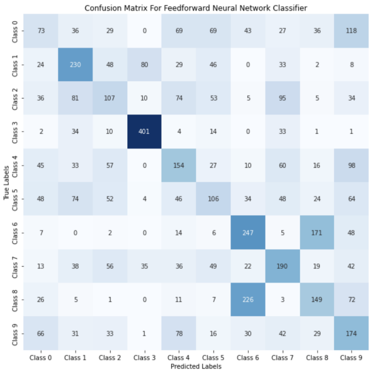
When looking at the confusion matrix for the AdaBoost Classifier, we can see that the most correctly classified class was Class 3 (which is Classical music) with 401/500 correct classifications. The most incorrectly classified classes were Class 0 with 73/500 correct classifications, Class 2 with 107/500 correct classifications and Class 6 with 106/500 correct classifications – these music genres are Alternative, Blues, and Hip Hop respectively. When looking at the overall predictions, I saw that Class 9 (Rock) was the most predicted class with 659 predictions and Class 0 (Alternative) was the least predicted class with 340 predictions.



	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9
Class 0	73	36	29	0	69	69	43	27	36	118
Class 1	24	230	48	80	29	46	0	33	2	8
Class 2	36	81	107	10	74	53	5	95	5	34
Class 3	2	34	10	401	4	14	0	33	1	1
Class 4	45	33	57	0	154	27	10	60	16	98
Class 5	48	74	52	4	46	106	34	48	24	64
Class 6	7	0	2	0	14	6	247	5	171	48
Class 7	13	38	56	35	36	49	22	190	19	42
Class 8	26	5	1	0	11	7	226	3	149	72
Class 9	66	31	33	1	78	16	30	42	29	174

```
Predicted Values
0: 340
1: 562
2: 395
3: 531
4: 515
5: 393
6: 617
7: 536
8: 452
9: 659
```

When looking at the confusion matrix for the Feedforward Neural Network Classifier, we can see that the most correctly classified class was Class 3 (which is Classical music) with 401/500 correct classifications. The most incorrectly classified classes were Class 0 with 73/500 correct classifications, Class 2 with 107/500 correct classifications and Class 6 with 106/500 correct classifications – these music genres are Alternative, Blues, and Hip Hop respectively. When looking at the overall predictions, I saw that Class 9 (Rock) was the most predicted class with 659 predictions and Class 0 (Alternative) was the least predicted class with 340 predictions.



	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9
Class 0	73	36	29	0	69	69	43	27	36	118
Class 1	24	230	48	80	29	46	0	33	2	8
Class 2	36	81	107	10	74	53	5	95	5	34
Class 3	2	34	10	401	4	14	0	33	1	1
Class 4	45	33	57	0	154	27	10	60	16	98
Class 5	48	74	52	4	46	106	34	48	24	64
Class 6	7	0	2	0	14	6	247	5	171	48
Class 7	13	38	56	35	36	49	22	190	19	42
Class 8	26	5	1	0	11	7	226	3	149	72
Class 9	66	31	33	1	78	16	30	42	29	174

```
Predicted Values
0: 340
1: 562
2: 395
3: 531
4: 515
5: 393
6: 617
7: 536
8: 452
9: 659
```

Based on this, we can reasonably conclude that Classical music has various distinct qualities in comparison to other music genres that make it more easily correctly classify while Alternative and Blues music have qualities that are less distinguishable in comparison to other music genres.