Ava McCormack 99121093
CIS4930 Individual Coding Assignment
Spring 2023

1. Problem Statement

In this assignment, I was given 4 folders with an assigned emotion each containing 100 audio files. Given the training data, I must build an audio emotion recognition model that is able to detect sentiment given an audio file. To achieve this goal, I used several different audio feature extraction methods and applied them to several different models.

2. Data Preparation

First, I read the 400 audio files using librosa by retrieving information such as signal and sample rate and kept them separated by emotion. I also created several plots just to be able to visualize each emotion which is included in my ipynb file.

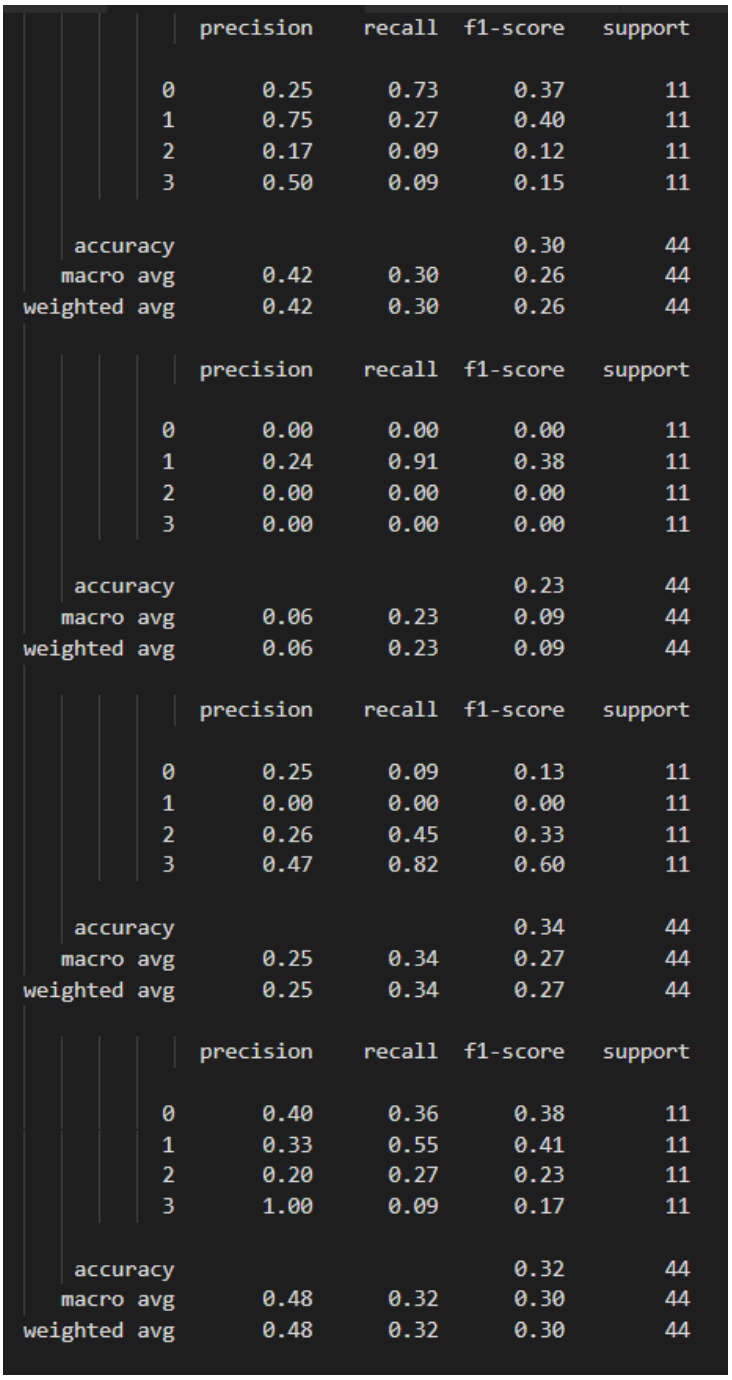
Feature Extraction
I extracted 4 features: Zero Crossing Rate, Loudness, MFCC, and Chroma Vector. Each feature was extracted and saved in a dataframe that contained all 100 files of each respective emotion.
After building those 4 dataframes, I needed to scale the values from -1 to 1.
After scaling, I averaged features every 10 windows for each file.
Following that I needed to average all features in all files by the time windows. I also combined all data into one data frame and assigned a number for each emotion {angry = 0, fear = 1, happy = 2, sad = 3}. This simplified the data and allowed for a much simpler model building stage.
Originally, I had included window in this large array and included it into the model; however, I realized all the models were getting practically the same values. I then took away window and got varying results as expected.
I also realized while model testing that I needed to split the data before averaging the files by window.

After making those changes, I built and tested my models. The scores, however, were quite poor so I reconstructed my data again. (See here for more)

3. Model Development

- Model Training
  I used 4 models: k-Nearest Neighbors, Decision Trees, Naïve Bayes, and Random Forest.
  I applied each of my 4 extraction features to each model in order to compare and contrast the differences between different models with the same feature extraction methods and the same models with different feature extraction methods. I also created models with all four features.
- Model Evaluation :

Comparing Different Models with all Features:



	precision	recall	f1-score	support
0	0.25	0.73	0.37	11
1	0.75	0.27	0.40	11
2	0.17	0.09	0.12	11
3	0.50	0.09	0.15	11
accuracy			0.30	44
macro avg	0.42	0.30	0.26	44
weighted avg	0.42	0.30	0.26	44

Decision Tree
Accuracy is 0.30
F1-Score for predicting 'Angry' is 0.37
F1-Score for predicting 'Fear is 0.40
F1-Score for predicting 'Happy is 0.12
F1-Score for predicting 'Sad is 0.15

	precision	recall	f1-score	support
0	0.00	0.00	0.00	11
1	0.24	0.91	0.38	11
2	0.00	0.00	0.00	11
3	0.00	0.00	0.00	11
accuracy			0.23	44
macro avg	0.06	0.23	0.09	44
weighted avg	0.06	0.23	0.09	44

KNN
Accuracy is 0.23
F1-Score for predicting 'Angry' is 0
F1-Score for predicting 'Fear is 0.38
F1-Score for predicting 'Happy is 0
F1-Score for predicting 'Sad is 0

	precision	recall	f1-score	support
0	0.25	0.09	0.13	11
1	0.00	0.00	0.00	11
2	0.26	0.45	0.33	11
3	0.47	0.82	0.60	11
accuracy			0.34	44
macro avg	0.25	0.34	0.27	44
weighted avg	0.25	0.34	0.27	44

Naïve Bayes
Accuracy is 0.34
F1-Score for predicting 'Angry' is 0.13
F1-Score for predicting 'Fear is 0
F1-Score for predicting 'Happy is 0.33
F1-Score for predicting 'Sad is 0.60

	precision	recall	f1-score	support
0	0.40	0.36	0.38	11
1	0.33	0.55	0.41	11
2	0.20	0.27	0.23	11
3	1.00	0.09	0.17	11
accuracy			0.32	44
macro avg	0.48	0.32	0.30	44
weighted avg	0.48	0.32	0.30	44

Random Forest
Accuracy is 0.32
F1-Score for predicting 'Angry' is 0.38
F1-Score for predicting 'Fear is 0.41
F1-Score for predicting 'Happy is 0.23
F1-Score for predicting 'Sad is 0.17

The best accuracy (of 0.45) was actually found in the models with only the zero crossing rate feature. The rest of the models performed quite poorly including the one above so I decided to go back and re-evaluate how I was getting data. I decided that instead of averaging files by window size, I would just average all window sizes for each file. This resulted in more values and better model scores overall.

I will forego placing the rest of the scores; however, they are still in the ipynb.

NEW DATASETS MODEL EVALUATION

Comparing Different Models with all Features:

	precision	recall	f1-score	support
0	0.61	0.77	0.68	30
1	0.52	0.43	0.47	30
2	0.84	0.70	0.76	30
3	0.91	0.97	0.94	30
accuracy			0.72	120
macro avg	0.72	0.72	0.71	120
weighted avg	0.72	0.72	0.71	120

Decision Tree

Accuracy is 0.72

F1-Score for predicting 'Angry' is 0.68

F1-Score for predicting 'Fear is 0.47

F1-Score for predicting 'Happy is 0.76

F1-Score for predicting 'Sad is 0.94

	precision	recall	f1-score	support
0	0.57	0.87	0.68	30
1	0.41	0.30	0.35	30
2	0.47	0.47	0.47	30
3	0.64	0.47	0.54	30
accuracy			0.53	120
macro avg	0.52	0.53	0.51	120
weighted avg	0.52	0.53	0.51	120

KNN

Accuracy is 0.53

F1-Score for predicting 'Angry' is 0.68

F1-Score for predicting 'Fear is 0.35

F1-Score for predicting 'Happy is 0.47

F1-Score for predicting 'Sad is 0.54

	precision	recall	f1-score	support
0	0.86	0.60	0.71	30
1	0.50	0.67	0.57	30
2	0.00	0.00	0.00	30
3	0.51	1.00	0.67	30
accuracy			0.57	120
macro avg	0.47	0.57	0.49	120
weighted avg	0.47	0.57	0.49	120

Naïve Bayes

Accuracy is 0.57

F1-Score for predicting 'Angry' is 0.71

F1-Score for predicting 'Fear is 0.57

F1-Score for predicting 'Happy is 0

F1-Score for predicting 'Sad is 0.67

	precision	recall	f1-score	support
0	0.74	0.93	0.82	30
1	0.79	0.73	0.76	30
2	0.96	0.80	0.87	30
3	1.00	0.97	0.98	30
accuracy			0.86	120
macro avg	0.87	0.86	0.86	120
weighted avg	0.87	0.86	0.86	120

Random Forest

Accuracy is 0.86

F1-Score for predicting 'Angry' is 0.82

F1-Score for predicting 'Fear is 0.76

F1-Score for predicting 'Happy is 0.87

F1-Score for predicting 'Sad is 0.98

Already, we can see much better scores.

The model with the highest accuracy when fitted with all features was Random Forest 0.86

The model with the lowest accuracy when fitted with all features was KNN with 0.53

Comparing Different Models with Loudness:

	precision	recall	f1-score	support
0	0.25	0.30	0.27	30
1	0.24	0.17	0.20	30
2	0.70	0.47	0.56	30
3	0.49	0.70	0.58	30
accuracy			0.41	120
macro avg	0.42	0.41	0.40	120
weighted avg	0.42	0.41	0.40	120

Decision Tree

Accuracy is 0.41

F1-Score for predicting 'Angry' is 0.27

F1-Score for predicting 'Fear is 0.20

F1-Score for predicting 'Happy is 0.56

F1-Score for predicting 'Sad is 0.58

	precision	recall	f1-score	support
0	0.34	0.40	0.37	30
1	0.29	0.33	0.31	30
2	0.75	0.40	0.52	30
3	0.60	0.70	0.65	30
accuracy			0.46	120
macro avg	0.50	0.46	0.46	120
weighted avg	0.50	0.46	0.46	120

KNN

Accuracy is 0.46

F1-Score for predicting 'Angry' is 0.37

F1-Score for predicting 'Fear is 0.31

F1-Score for predicting 'Happy is 0.52

F1-Score for predicting 'Sad is 0.65

	precision	recall	f1-score	support
0	0.00	0.00	0.00	30
1	0.00	0.00	0.00	30
2	0.00	0.00	0.00	30
3	0.25	1.00	0.40	30
accuracy			0.25	120
macro avg	0.06	0.25	0.10	120
weighted avg	0.06	0.25	0.10	120

Naïve Bayes

Accuracy is 0.25

F1-Score for predicting 'Angry' is 0

F1-Score for predicting 'Fear is 0

F1-Score for predicting 'Happy is 0

F1-Score for predicting 'Sad is 0.40

	precision	recall	f1-score	support
0	0.25	0.30	0.27	30
1	0.24	0.17	0.20	30
2	0.70	0.47	0.56	30
3	0.49	0.70	0.58	30
accuracy			0.41	120
macro avg	0.42	0.41	0.40	120
weighted avg	0.42	0.41	0.40	120

Random Forest

Accuracy is 0.41

F1-Score for predicting 'Angry' is 0.27

F1-Score for predicting 'Fear is 0.20

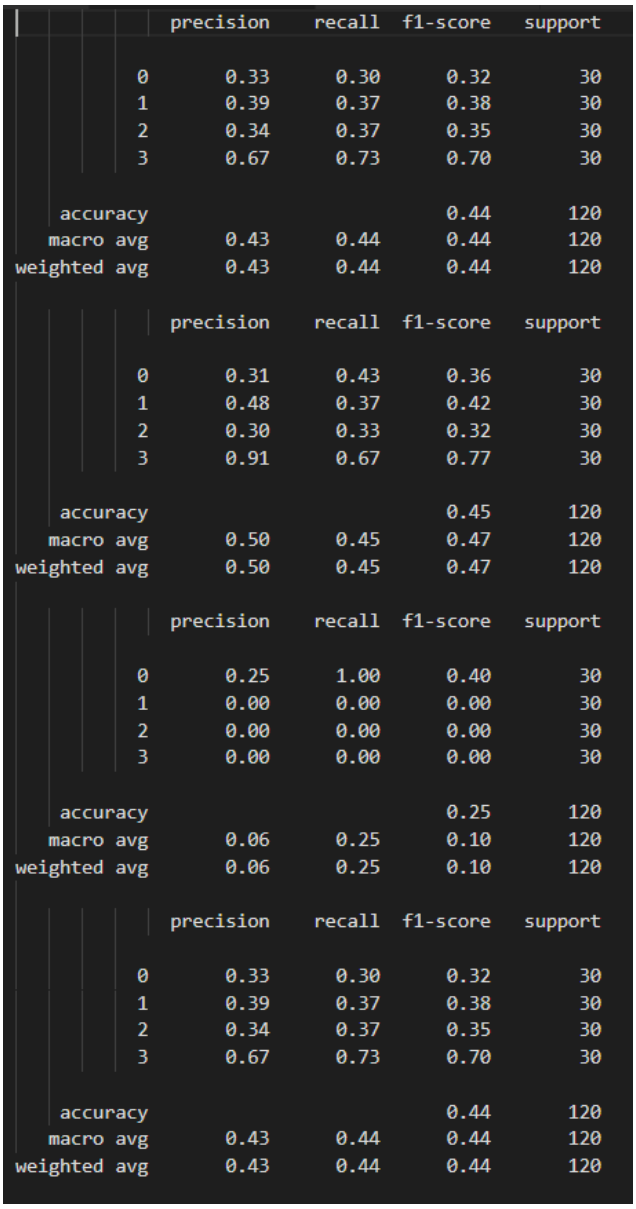
F1-Score for predicting 'Happy is 0.56

F1-Score for predicting 'Sad is 0.58

The model with the highest accuracy when fitted with Loudness was KNN 0.46

The model with the lowest accuracy when fitted with Loudness was Naïve Bayes with 0.25

Comparing Different Models with Zero-Crossing Rate:



	precision	recall	f1-score	support
0	0.33	0.30	0.32	30
1	0.39	0.37	0.38	30
2	0.34	0.37	0.35	30
3	0.67	0.73	0.70	30
accuracy			0.44	120
macro avg	0.43	0.44	0.44	120
weighted avg	0.43	0.44	0.44	120

	precision	recall	f1-score	support
0	0.31	0.43	0.36	30
1	0.48	0.37	0.42	30
2	0.30	0.33	0.32	30
3	0.91	0.67	0.77	30
accuracy			0.45	120
macro avg	0.50	0.45	0.47	120
weighted avg	0.50	0.45	0.47	120

	precision	recall	f1-score	support
0	0.25	1.00	0.40	30
1	0.00	0.00	0.00	30
2	0.00	0.00	0.00	30
3	0.00	0.00	0.00	30
accuracy			0.25	120
macro avg	0.06	0.25	0.10	120
weighted avg	0.06	0.25	0.10	120

	precision	recall	f1-score	support
0	0.33	0.30	0.32	30
1	0.39	0.37	0.38	30
2	0.34	0.37	0.35	30
3	0.67	0.73	0.70	30
accuracy			0.44	120
macro avg	0.43	0.44	0.44	120
weighted avg	0.43	0.44	0.44	120

Decision Tree

Accuracy is 0.44
F1-Score for predicting ‘Angry’ is 0.32
F1-Score for predicting ‘Fear is 0.38
F1-Score for predicting ‘Happy is 0.35
F1-Score for predicting ‘Sad is 0.70

KNN

Accuracy is 0.45
F1-Score for predicting ‘Angry’ is 0.36
F1-Score for predicting ‘Fear is 0.42
F1-Score for predicting ‘Happy is 0.32
F1-Score for predicting ‘Sad is 0.77

Naïve Bayes

Accuracy is 0.25
F1-Score for predicting ‘Angry’ is 0.40
F1-Score for predicting ‘Fear is 0
F1-Score for predicting ‘Happy is 0
F1-Score for predicting ‘Sad is 0

Random Forest

Accuracy is 0.44
F1-Score for predicting ‘Angry’ is 0.32
F1-Score for predicting ‘Fear is 0.38
F1-Score for predicting ‘Happy is 0.35
F1-Score for predicting ‘Sad is 0.70

The model with the highest accuracy when fitted with zero-crossing was KNN 0.45
The model with the lowest accuracy when fitted with zero-crossing was Naïve Bayes with 0.25

Comparing Different Models with MFCC:

	precision	recall	f1-score	support
0	0.53	0.77	0.63	30
1	0.32	0.27	0.29	30
2	0.30	0.27	0.28	30
3	0.60	0.50	0.55	30
accuracy			0.45	120
macro avg	0.44	0.45	0.44	120
weighted avg	0.44	0.45	0.44	120

Decision Tree

Accuracy is 0.45
F1-Score for predicting 'Angry' is 0.63
F1-Score for predicting 'Fear is 0.29
F1-Score for predicting 'Happy is 0.28
F1-Score for predicting 'Sad is 0.55

	precision	recall	f1-score	support
0	0.55	0.87	0.68	30
1	0.26	0.23	0.25	30
2	0.36	0.40	0.38	30
3	0.46	0.20	0.28	30
accuracy			0.42	120
macro avg	0.41	0.42	0.40	120
weighted avg	0.41	0.42	0.40	120

KNN

Accuracy is 0.42
F1-Score for predicting 'Angry' is 0.68
F1-Score for predicting 'Fear is 0.25
F1-Score for predicting 'Happy is 0.38
F1-Score for predicting 'Sad is 0.28

	precision	recall	f1-score	support
0	0.54	0.93	0.68	30
1	0.38	0.17	0.23	30
2	0.41	0.40	0.41	30
3	0.62	0.53	0.57	30
accuracy			0.51	120
macro avg	0.49	0.51	0.47	120
weighted avg	0.49	0.51	0.47	120

Naïve Bayes

Accuracy is 0.51
F1-Score for predicting 'Angry' is 0.68
F1-Score for predicting 'Fear is 0.23
F1-Score for predicting 'Happy is 0.41
F1-Score for predicting 'Sad is 0.57

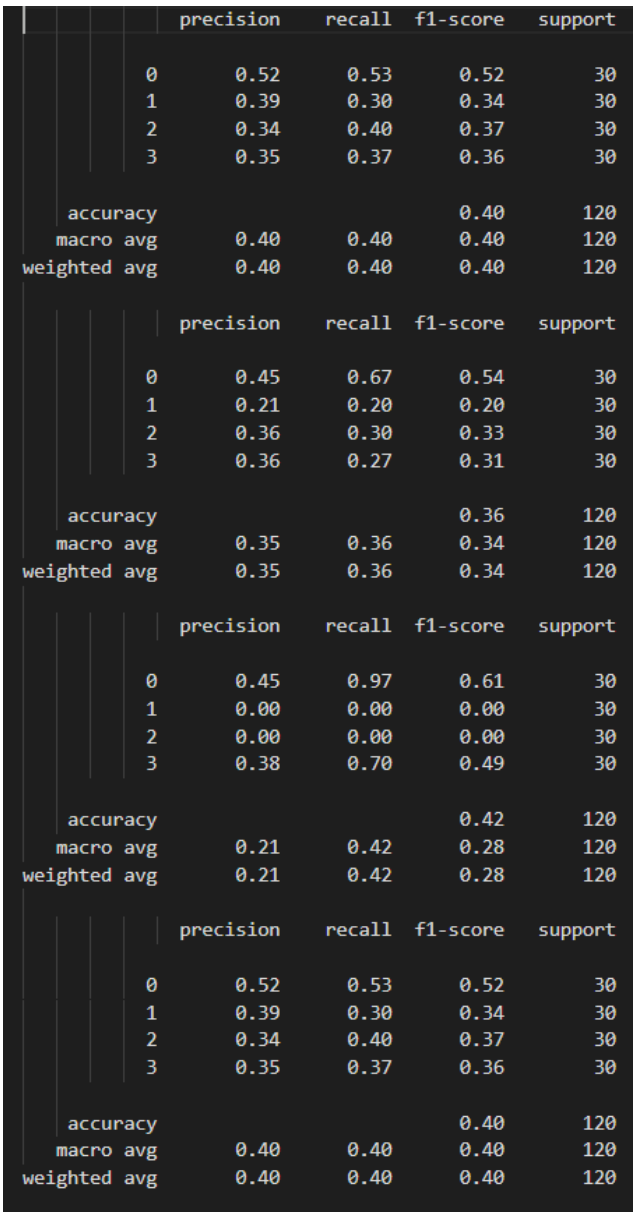
	precision	recall	f1-score	support
0	0.53	0.77	0.63	30
1	0.32	0.27	0.29	30
2	0.30	0.27	0.28	30
3	0.60	0.50	0.55	30
accuracy			0.45	120
macro avg	0.44	0.45	0.44	120
weighted avg	0.44	0.45	0.44	120

Random Forest

Accuracy is 0.45
F1-Score for predicting 'Angry' is 0.63
F1-Score for predicting 'Fear is 0.29
F1-Score for predicting 'Happy is 0.28
F1-Score for predicting 'Sad is 0.55

The model with the highest accuracy when fitted with MFCC was Naïve Bayes 0.51
The model with the lowest accuracy when fitted with MFCC was KNN with 0.42

Comparing Different Models with Chroma:



	precision	recall	f1-score	support
0	0.52	0.53	0.52	30
1	0.39	0.30	0.34	30
2	0.34	0.40	0.37	30
3	0.35	0.37	0.36	30
accuracy			0.40	120
macro avg	0.40	0.40	0.40	120
weighted avg	0.40	0.40	0.40	120

	precision	recall	f1-score	support
0	0.45	0.67	0.54	30
1	0.21	0.20	0.20	30
2	0.36	0.30	0.33	30
3	0.36	0.27	0.31	30
accuracy			0.36	120
macro avg	0.35	0.36	0.34	120
weighted avg	0.35	0.36	0.34	120

	precision	recall	f1-score	support
0	0.45	0.97	0.61	30
1	0.00	0.00	0.00	30
2	0.00	0.00	0.00	30
3	0.38	0.70	0.49	30
accuracy			0.42	120
macro avg	0.21	0.42	0.28	120
weighted avg	0.21	0.42	0.28	120

	precision	recall	f1-score	support
0	0.52	0.53	0.52	30
1	0.39	0.30	0.34	30
2	0.34	0.40	0.37	30
3	0.35	0.37	0.36	30
accuracy			0.40	120
macro avg	0.40	0.40	0.40	120
weighted avg	0.40	0.40	0.40	120

Decision Tree
Accuracy is 0.40
F1-Score for predicting ‘Angry’ is 0.52
F1-Score for predicting ‘Fear is 0.34
F1-Score for predicting ‘Happy is 0.37
F1-Score for predicting ‘Sad is 0.36

KNN
Accuracy is 0.36
F1-Score for predicting ‘Angry’ is 0.54
F1-Score for predicting ‘Fear is 0.20
F1-Score for predicting ‘Happy is 0.33
F1-Score for predicting ‘Sad is 0.31

Naïve Bayes
Accuracy is 0.42
F1-Score for predicting ‘Angry’ is 0.61
F1-Score for predicting ‘Fear is 0
F1-Score for predicting ‘Happy is 0
F1-Score for predicting ‘Sad is 0.49

Random Forest
Accuracy is 0.40
F1-Score for predicting ‘Angry’ is 0.52
F1-Score for predicting ‘Fear is 0.34
F1-Score for predicting ‘Happy is 0.37
F1-Score for predicting ‘Sad is 0.36

The model with the highest accuracy when fitted with Chroma was Naïve Bayes 0.42
The model with the lowest accuracy when fitted with Chroma was KNN 0.36

If we compare the modified data to the original, we will see an increase of up to 54% in accuracy.
The only models with accuracy scores that decreased were three of the Zero-Crossing Feature models; however, the scores were only decreased by 0.01, 0.01, 0.9 and the increase to one was .20.

Now, we will summarize the results of the modified data models.

The **overall best model**, Random Forest fitted with all features, had an accuracy score of 0.86
The **best single feature model**, Naïve Bayes fitted with MFCC, had an accuracy score of 0.51

Comparing Same Model with Different Features:
ALL models with all features performed best with accuracy scores of 0.72, 0.53, 0.57, and 0.86

If we look at just individual features:
Decision Tree with MFCC - 0.45
KNN with Loudness - 0.46
Naïve Bayes with MFCC - 0.51
Random Forest with MFCC - 0.45

4. Discussion

- My best model performs with an accuracy of 86%.
- Some problems I ran into during this assignment, was the vast amount of data.
- I also had trouble trying to transform the data into the appropriate format.
- I would realize I had gone wrong somewhere while observing the results of model testing and fix my data
- I also had issues with figuring out how to apply the extracted features to the classifiers I have used previously. This ended up being quite simple, but I was unsure how to approach it at first.

5. Appendix

Colab: https://colab.research.google.com/drive/1GpuQHtSQifkt0KZ8ESmneOlS-bZ120SX?usp=sharing