[Nathan Gilman, 34785552]

CIS4930 Individual Coding Assignment
Spring 2023

1. Problem Statement

Over the past twenty years, technological progress has surged in almost every field, with many fields having significant advancements in this time period. One relevant example of this is the ability of computers to recognize provided audio samples. This has opened new avenues of extracting data and drawing conclusions. A specific example of the new technological progress in audio recognition is the ability for computers to reliably predict the emotion of a clip. This is an important accomplishment because it will help us understand human emotions more objectively.

This advancement has many potential benefits for society, ranging from impacting customer service at retail stores to completely changing the way that individuals with communication issues or disorders interact with society. As researchers continue to push the boundaries of this field, we can expect even more audio related progress.

2. Data Preparation

Unlike previous assignments, this assignment does not deal with a single csv file for us to load. Instead, we are using a folder that contains a total of 400 .wav audio files. To begin the data preparation/exploration, I first gathered the file names of these files into a list of strings. This allowed me to dynamically iterate over the files and extract features as necessary. I also used sklearn to divide these files by a 70-30 test-train split for later validation of my model.

Once I stored the necessary files within a list, I created additional lists to hold the data frames of extracted features. The features that I decided to extract were Mel-frequency cepstral coefficients (MFCCs), Zero Crossing Rate (ZCR), and Mel Spectrogram. I chose these features because they were directly covered and explained in the course materials. While there were also a few additional features covered in class, I decided to focus on these three features and add more if necessary. These features were extracted with librosa, a popular python module for audio data and files.

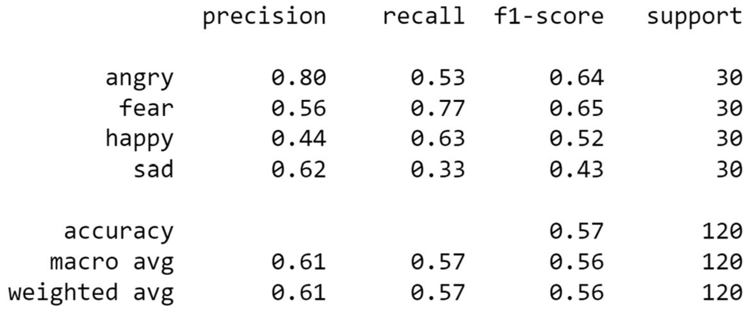
To make it more convenient to work with the extracted features, I decided to create a single feature matrix for each of the .wav files. This was done by iterating the test and train file sets for each of the 4 emotions. As a result, I ended up with a total of 8 lists of feature matrices representing the happy, sad, angry, and fear categories for both the training and test datasets.

Overall, by using these techniques to manipulate and extract features from the audio files, we were able to effectively prepare the data for further analysis and modeling.

3. Model Development

- Model Training
    - *Unlike previous assignments, I needed to take a different approach when deciding on key features for model implementation. This was because the assignment dealt with a multi-class classification problem and, as such, some of the previously used classifiers could not be used in this case, namely logistic regression. I decided to use the K Nearest Neighbors model due to its popularity for multi-class classification problems and its performant properties.*

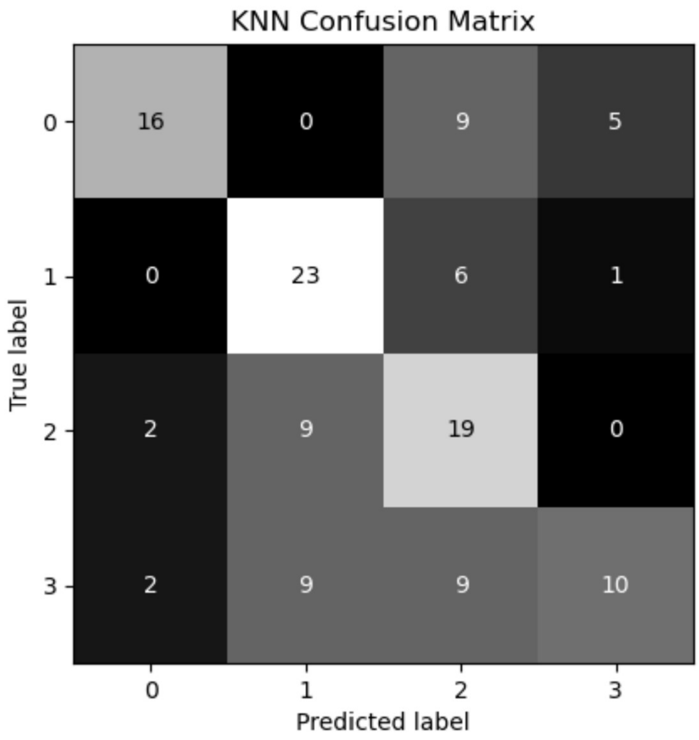
- Model Evaluation
    - *The KNN model that I implemented had the following results:*



```
              precision    recall  f1-score   support

       angry       0.80      0.53      0.64        30
        fear       0.56      0.77      0.65        30
       happy       0.44      0.63      0.52        30
         sad       0.62      0.33      0.43        30

    accuracy                           0.57       120
   macro avg       0.61      0.57      0.56       120
weighted avg       0.61      0.57      0.56       120
```



	0	1	2	3
0	16	0	9	5
1	0	23	6	1
2	2	9	19	0
3	2	9	9	10

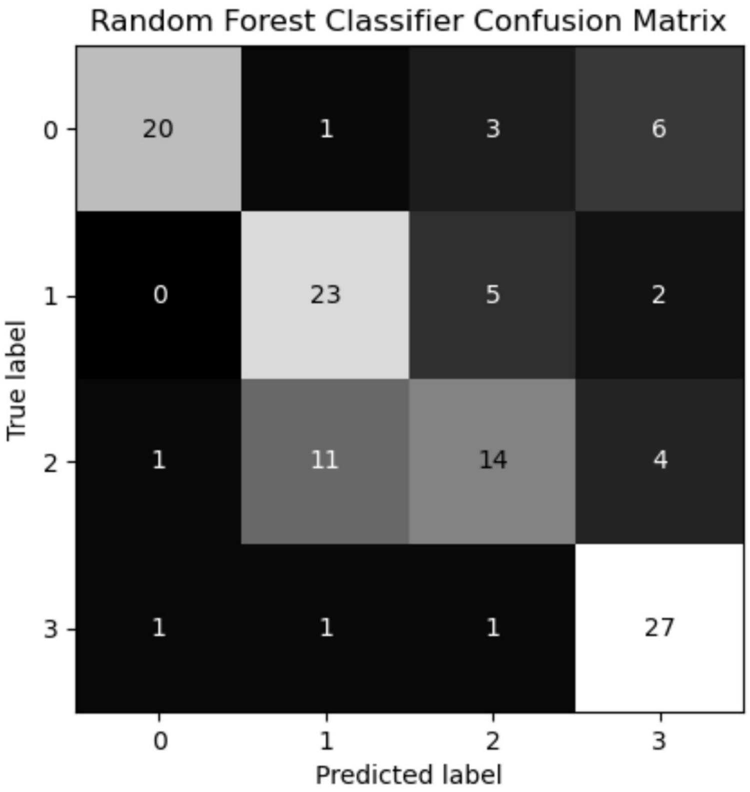
```
<Figure size 3000x1500 with 0 Axes>
```

- *In addition to the KNN model, I also tested the Naive Bayes Classifier and Random Forest Classifier on the extracted features. The Naïve Bayes classifier did slightly better than the KNN model with an accuracy of 0.62. The Random Forest Classifier does significantly better than the other 2 with an accuracy of 0.74. The results for the random forest classifier are below:*

```
              precision    recall  f1-score   support

       angry       0.86      0.63      0.73        30
        fear       0.78      0.70      0.74        30
       happy       0.56      0.77      0.65        30
         sad       0.87      0.87      0.87        30

    accuracy                           0.74       120
   macro avg       0.77      0.74      0.75       120
weighted avg       0.77      0.74      0.75       120
```



```
<Figure size 3000x1500 with 0 Axes>
```

4. Discussion

- *While my KNN model had an accuracy of about 60%, I feel that I could have greatly improved the performance in a variety of ways. Firstly, I chose the three features that I believed to be primary in terms of importance for classification. I omitted other features such as loudness and chroma. Although I am unsure of the effect that including these features would have on my model's performance, I am inclined to believe that it would result in an suitable increase. Additionally, my model's post processing was modest as I believed that the librosa library took care of various parts of processing for me. I chose a MinMaxScaler from Python's sklearn module to scale the extracted features in the data frame. I don't* *believe that additional post processing would have yielded me results comparable to additional feature extraction, but it may have.*

- *There were two primary challenges faced in this assignment. The first challenge faced was the structure of the data. Whereas in past assignments the data was contained to a single csv, this project forced me to look towards the internet and search for resources regarding the extraction of feature from folders of files. Eventually, I gathered enough information and was able to begin my implementation. A second challenge I faced was choosing an appropriate model for this assignment. As my knowledge of the classifiers is rough at best, I was hesitant on choosing a model. Eventually, I went with the KNN model because it was a generally popular model that I read to be performant.*

- *While I found this assignment challenging due to its different nature compared to previous assignments, it was nonetheless rewarding. It forced me to learn how to deal with more complex and varying datasets. Additionally, the assignment was useful for me due to my final project involving an audio identification component.*

5. Appendix

- *Repository Link: https://github.com/npgilman/MachineLearningAudioModalities*
- *Librosa landing page: https://librosa.org/doc/main/index.html*