**Speech Emotion Detection**

**Introduction:**

As we know that any model in Machine Learning requires huge dataset. So first we look for a large data set so that we can train our model on and then we will test the model according to our requirements for the project.

**Data Used:**

We got audio datasets with around 500 audio files which were in the wav format from the following websites:

<http://kahlan.eps.surrey.ac.uk/savee/Download.html>

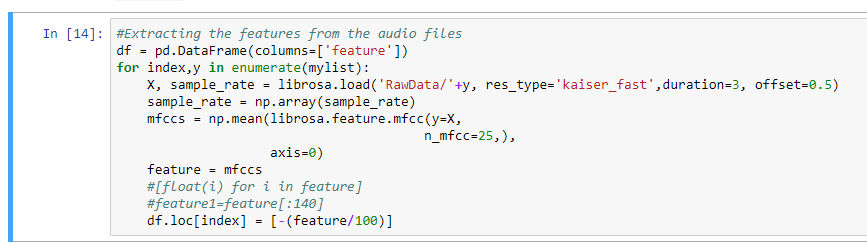
These files consist of recordings from four male actors expressing four different emotions in British English. The sentences were selected from the standard TIMIT corpus and phonetically balanced for each emotion. The recordings were made in a visual media lab with high-quality audio-visual equipment and were processed and labeled for emotion.

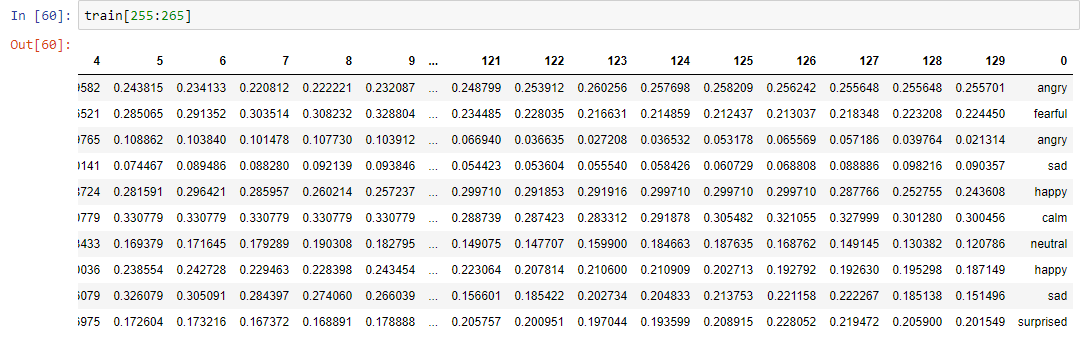
**Extracting Data from Audio files:**

We used Librosa library in Python to process and extract features from the audio files. Librosa is a python package for music and audio analysis. It provides the building blocks necessary to create music information retrieval systems. Using the librosa library we were able to extract features i.e. MFCC (Mel Frequency Cepstral Coefficient). MFCCs are a feature widely used in automatic speech and speaker recognition. We also separated out the females and males voice by the using the identifiers provided in the website. This was because as experiment we found out that separating male and female voices increased by 15%. It could be because of the pitch of the voice was affecting the results.

Each audio file gave us many features which were basically array of many values. These features were then appended by the labels which we created in the previous step.

The next step involved dealing with the missing features for some audio files which were shorter in length. We increased the sampling rate by twice to get the unique features of each emotional speech. We didn’t increase the sampling frequency even more since it might collect noise thus affecting the results





The next steps involve shuffling the data, splitting into train and test and then building a model to train our data.

We built a Multi Perceptron model, LSTM model and CNN models. The MLP and LSTM were not suitable as it gave us low accuracy. As our project is a classification problem where were categorize the different emotions, CNN worked best for us.