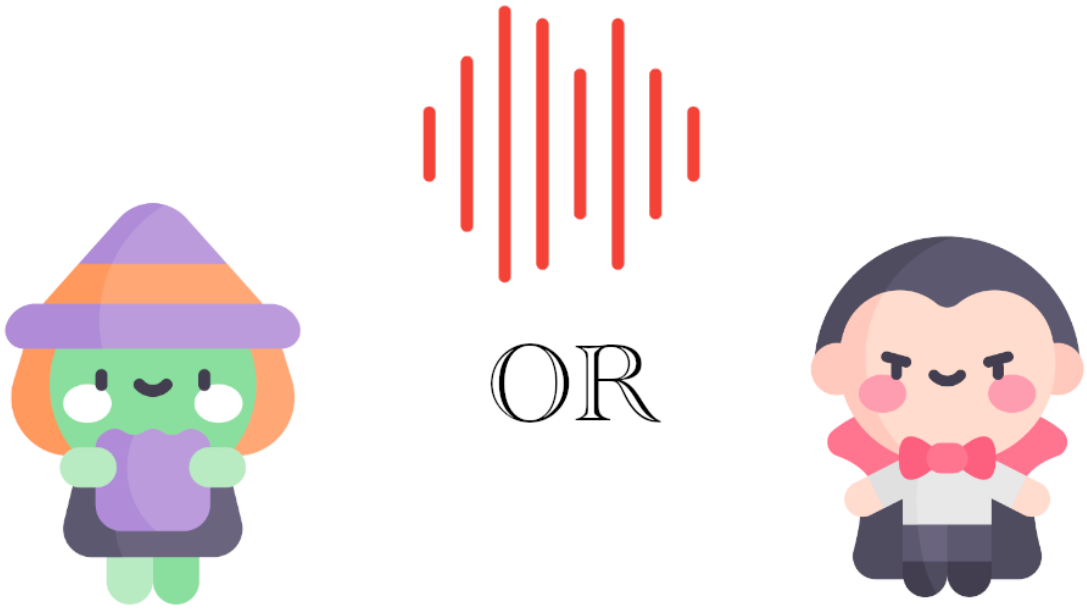
Artificial Intelligence and Machine Learning - CS 337

Gender Recognition from Audio

Shiv Kiran Bagathi - 200050019
Bukke Likith Rajesh - 200050025
Deekonda Venkatesh Prasad - 200050030
Jithendra Landa - 200050068



Introduction

Gender Detection from audio is a supervised learning task. In the speech recognition problem input will be the audio signal and we have to predict the text from the audio signal. We can't take the raw audio signal as input to our model because there will be a lot of noise in the audio signal. Speech signals taken from a recorded speech can be used to acquire acoustic attributes such as duration, intensity, frequency and filtering.

There are several studies for gender recognition and identification by voice using machine learning. However, the development of an accurate prediction model for gender recognition by voice is still considered a rather difficult and challenging task. The difficulties of this classification problem arise since speech signals are highly time-varying and have very high randomness.

We tackled this problem by developing two models

1. Feature-Based Model
2. End-to-end Model

Feature Based Model

This Model is classification based on acoustic properties of the whole fourier transform of the music.wav file.

DataSet:

Voice gender dataset (https://www.kaggle.com/primaryobjects/voicegender/home)
This database was created to identify a voice as male or female, based upon acoustic properties of the voice and speech. The dataset consists of 3,168 recorded voice samples, collected from male and female speakers. The voice samples are collected from :
– The Harvard-Haskins Database of Regularly-Timed Speech.

– Telecommunications & Signal Processing Laboratory Speech Database at McGill University.
– VoxForge Speech Corpus.– Festvox CMU-ARCTIC Speech Database at Carnegie Mellon University.
The voice samples are pre-processed by acoustic analysis in R using the seewave and tuneR packages, with an analyzed frequency range of 0hz-280hz (human vocal range).

At the time of this report the dataset has around 300k views adn 32k downloads.

What does the Dataset Contain ?

The following acoustic properties of each voice are measured and included within the CSV:

- meanfreq: mean frequency (in kHz)
- sd: standard deviation of frequenc
- median: median frequency (in kHz)
- Q25: first quantile (in kHz)
- Q75: third quantile (in kHz)
- IQR: interquantile range (in kHz)
- skew: skewness (see note in specprop description)
- kurt: kurtosis (see note in specprop description)
- sp.ent: spectral entropy
- sfm: spectral flatness
- mode: mode frequency
- centroid: frequency centroid (see specprop)
- peakf: peak frequency (frequency with highest energy)
- meanfun: average of fundamental frequency measured across acoustic signal
- minfun: minimum fundamental frequency measured across acoustic signal

- maxfun: maximum fundamental frequency measured across acoustic signal
- meandom: average of dominant frequency measured across acoustic signal
- mindom: minimum of dominant frequency measured across acoustic signal
- maxdom: maximum of dominant frequency measured across acoustic signal
- dfrange: range of dominant frequency measured across acoustic signal
- modindx: modulation index. Calculated as the accumulated absolute difference between adjacent measurements of fundamental frequencies divided by the frequency range
- label: male or female

Peakf and duration are set to a default value of 0, 20 seconds for CPU constraints.

Data Analysis:

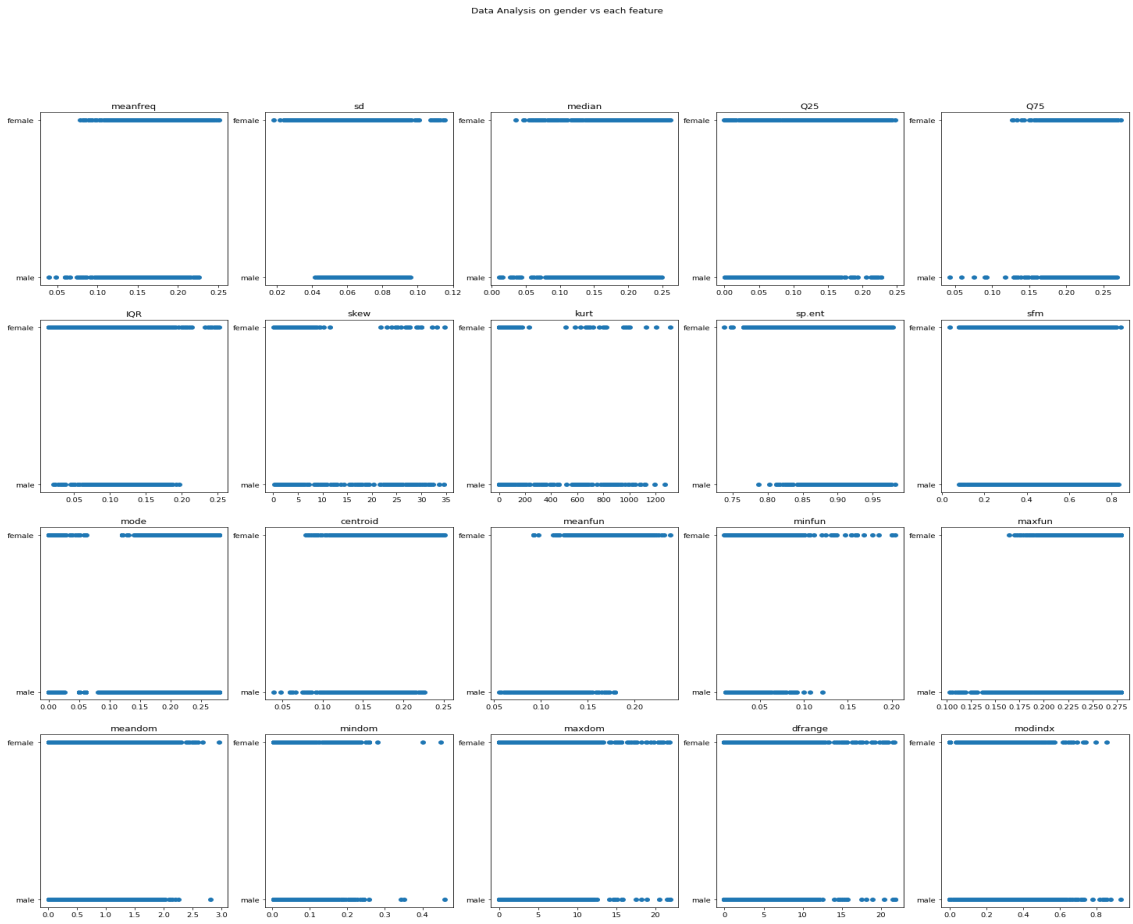


Fig: Variation of label with each of features.

Further analysis is performed in feature_colab.ipynb

Models Trained:

We trained the features on various models for classifying gender

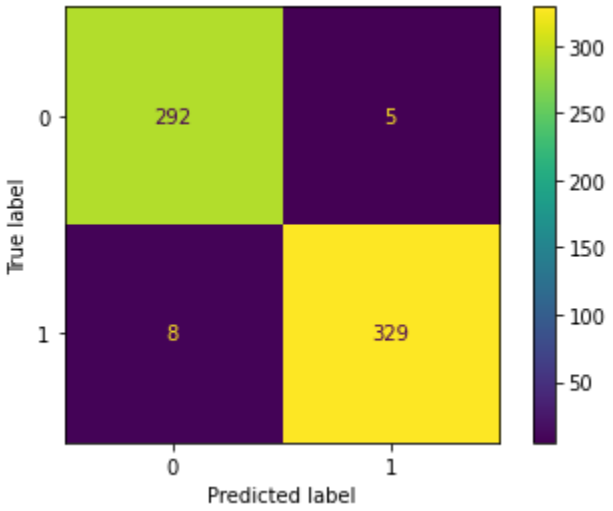
The Models we trained are :

1. Neural Net with 2 hidden layers
2. SVM Classifier
3. KNN Classifier
4. Random Forest Classifier
5. Decision Tree
6. MLP Classifier
7. GMM Model

Neural Net with 2 hidden layers.

```
# we will use a simple neural network with 2 hidden layers
# we will use the cross entropy loss function and the Adam optimizer
# we will use the accuracy as the metric to evaluate the model
```

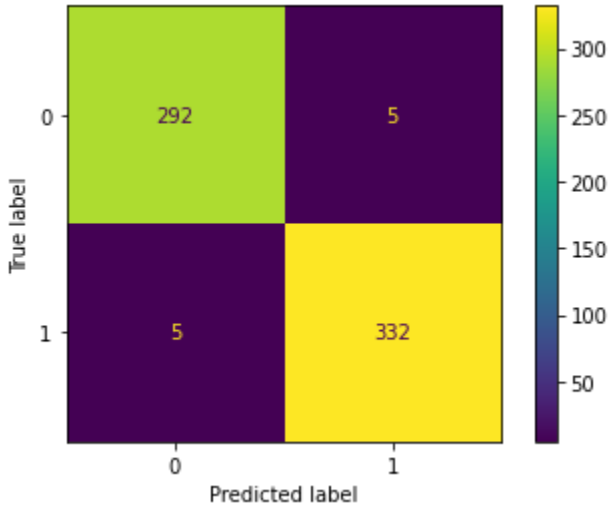
This model is based on simple sigmoid activation and final softmax layer with hidden layer of 6 nodes.



Results: Neural Net accuracy: 0.98, precision: 0.98, recall: 0.979, f1_score: 0.98

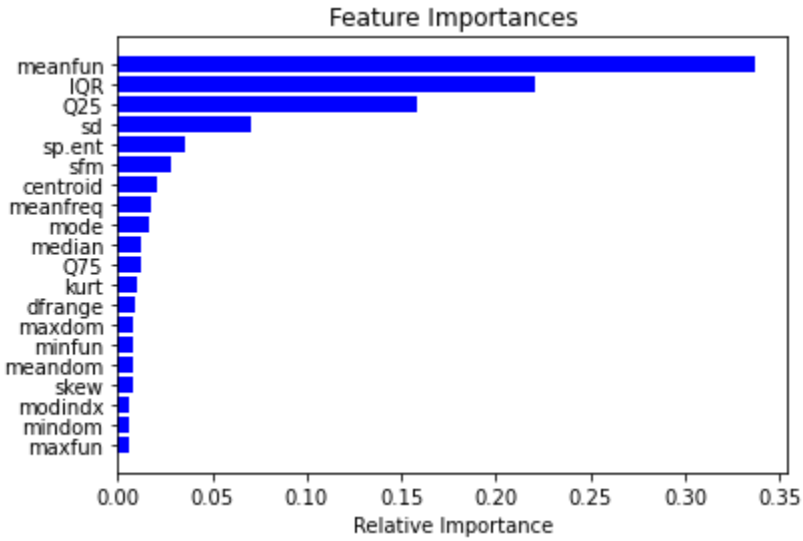
SVM Classifier with rbf kernel

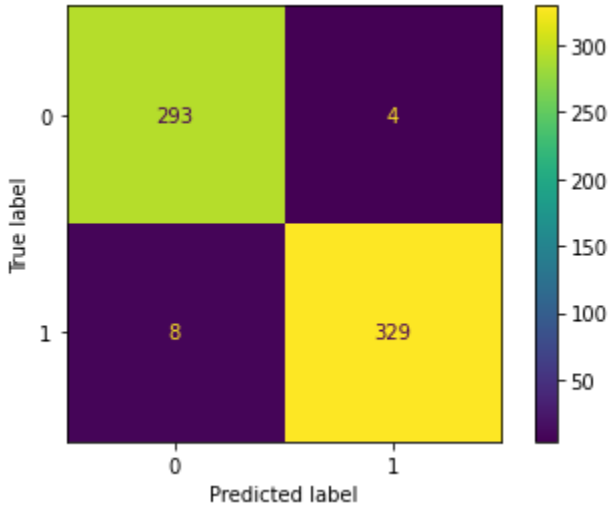
Results: SVM accuracy: 0.984 precision: 0.985 recall: 0.985 f1_score: 0.985



Random Forest Classifier

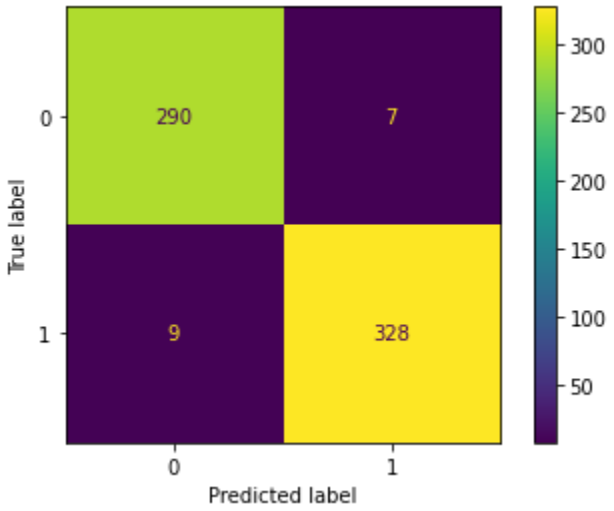
Results: RandomForest accuracy: 0.981 precision: 0.987 recall: 0.976 f1_score: 0.982

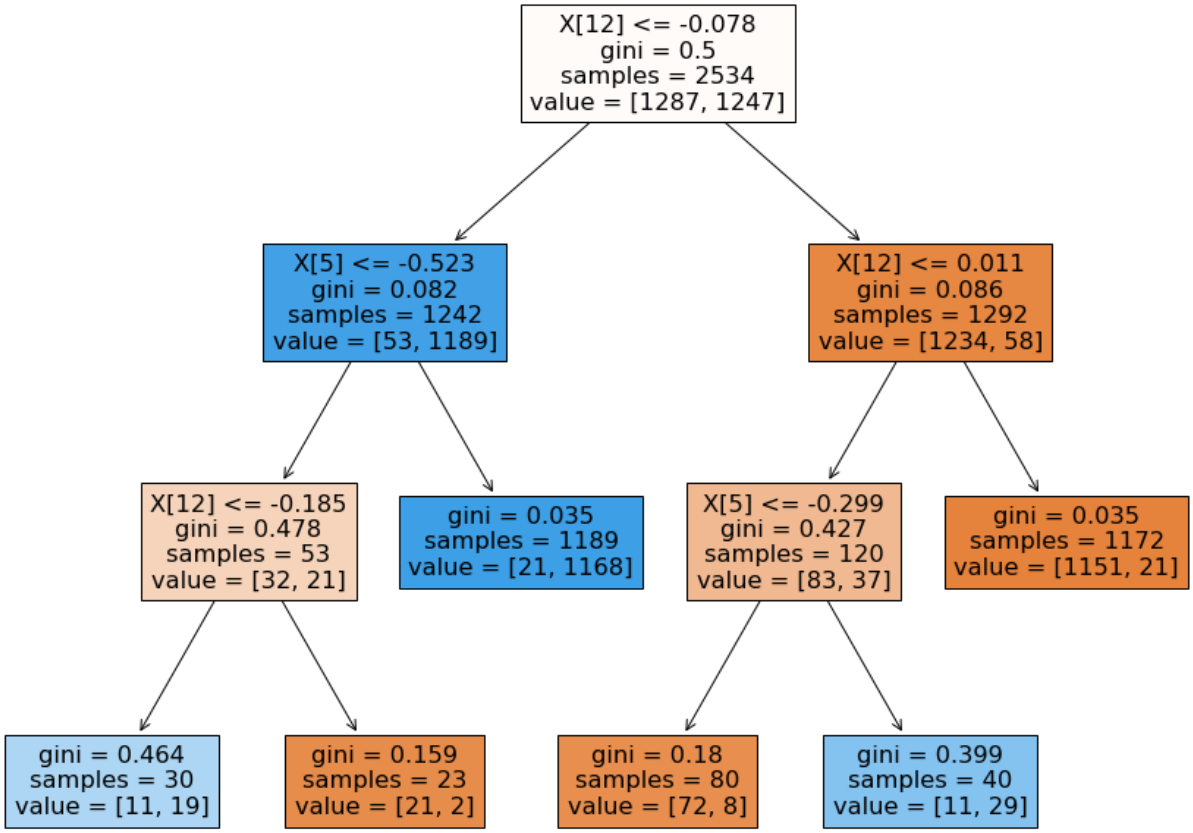




Decision Tree Classifier

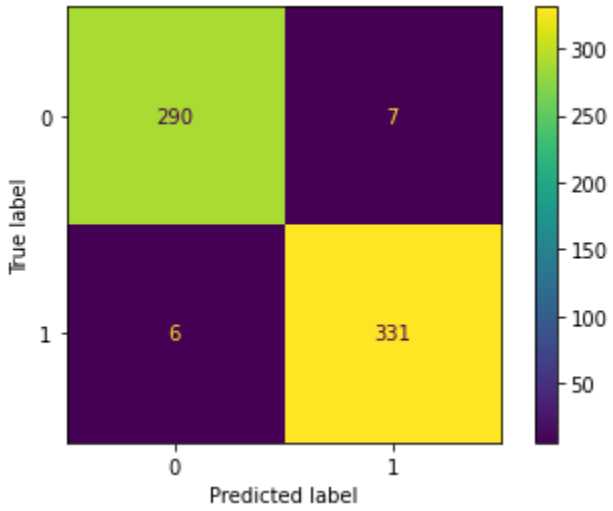
Decision Tree accuracy: 0.974 precision: 0.979 recall: 0.973, f1_score: 0.976


290
7
9
328
0
1
True label
0
1
Predicted label
300
250
200
150
100
50


X[12] <= -0.078
gini = 0.5
samples = 2534
value = [1287, 1247]
X[5] <= -0.523
gini = 0.082
samples = 1242
value = [53, 1189]
X[12] <= 0.011
gini = 0.086
samples = 1292
value = [1234, 58]
X[12] <= -0.185
gini = 0.478
samples = 53
value = [32, 21]
gini = 0.035
samples = 1189
value = [21, 1168]
X[5] <= -0.299
gini = 0.427
samples = 120
value = [83, 37]
gini = 0.035
samples = 1172
value = [1151, 21]
gini = 0.464
samples = 30
value = [11, 19]
gini = 0.159
samples = 23
value = [21, 2]
gini = 0.18
samples = 80
value = [72, 8]
gini = 0.399
samples = 40
value = [11, 29]

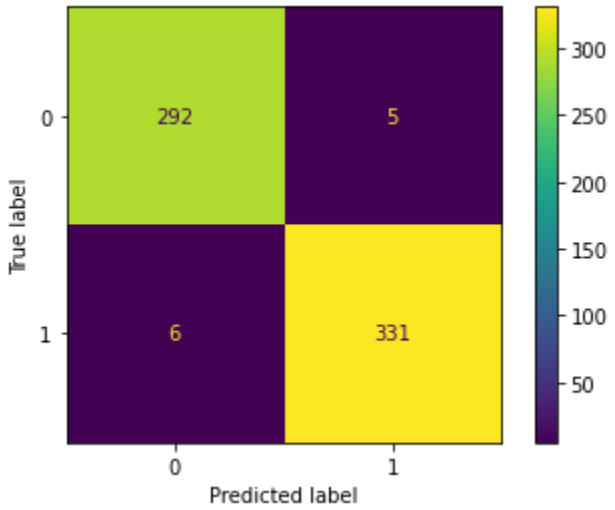
KNN Classifier

KNN accuracy: 0.979 precision: 0.979 recall: 0.982 f1: 0.98



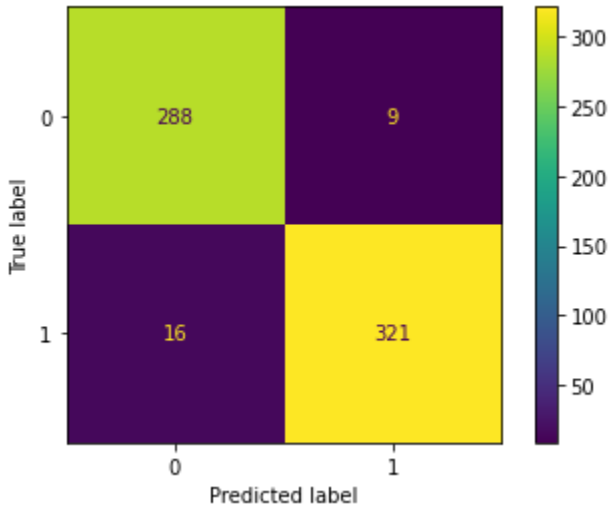
MLP Classifier

MLP accuracy: 0.982, precision: 0.985 recall: 0.982 f1_score: 0.983



GMM Classification

GMM accuracy: 0.96 precision: 0.972 recall: 0.952 f1_score: 0.962



Observations:

All models give relatively high accuracy on the model as the selected features are sufficient to properly differentiate between genders.

Neural Networks give relatively better performance but take more time.

Here we trained model on extracted features from the frequency distribution of the fourier transform of audio clip.

Next we will train a model to extract out the required features when we will give entire distribution as input.

End-to-End Model:

We applied Neural Network on features extracted from audio clips using famous extraction techniques

Dataset:

Common Voice is a corpus of speech data read by users on the Common Voice website (http://voice.mozilla.org/), and based upon text from a number of public domain sources like user submitted blog posts, old books, movies, and other public speech corpora. Its primary purpose is to enable the training and testing of automatic speech recognition (ASR) systems.

What does the Dataset Contain?

The corpus is labelled with 3 labels age, gender and accent.

It consists of 380k recorded voice samples, collected from male and female speakers.

Each row of a csv file represents a single audio clip, and contains the following information:
filename - relative path of the audio file
text - supposed transcription of the audio
age - age of the speaker, if the speaker reported it

- teens: '< 19'
- twenties: '19 - 29'
- thirties: '30 - 39'
- fourties: '40 - 49’
- fifties: '50 - 59’
- sixties: '60 - 69'

seventies: '70 - 79'

eighties: '80 - 89’

nineties: '> 89'

gender - gender of the speaker, if the speaker reported

Male

female

accent - accent of the speaker, if the speaker reported it

There are 30 accents.

The audio clips for each subset are stored as mp3 files in folders with the same naming conventions as it’s corresponding csv file. So, for instance, all audio data from the valid train set will be kept in the folder “cv-valid-train” alongside the “cv-valid-train.csv” metadata file.

Feature Extraction :

We extracted out features using preparation.py file which extracts all the features from the .mp3 and files and stores in .npy which are further used by the model.

We have used librosa library in python to extract the features.

1. Mel Spectogram Frequency
2. Mel Frequency cepstrum coefficients(mfcc)
3. Tonnetz
4. Chroma

Mel Spectogram Frequency

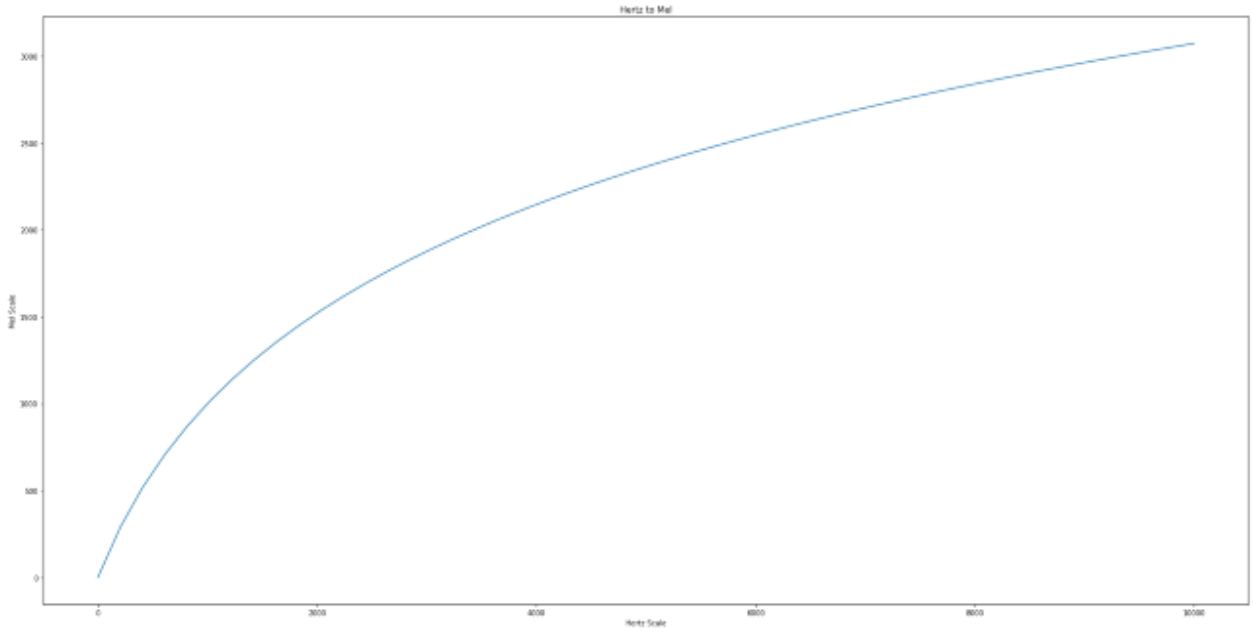
We use Mel Scale for conversion into features. The Mel Scale is a logarithmic transformation of a signal’s frequency. The core idea of this transformation is that sounds of equal distance on the Mel Scale are perceived to be of equal distance to

humans. I.e we can easily tell the difference between 100 hz and 200 hz but that is not as easy difference between 1000 hz and 1100 hz.

The transformation from the Hertz scale to the Mel Scale is the following:

$$m = 1127 \cdot \log\left(1 + \frac{f}{700}\right)$$

Visualizing the transformation



Mel Spectrograms are spectrograms that visualize sounds on the Mel scale as opposed to the usual frequency domain.

Mel Frequency cepstrum coefficients(mfcc)

The mel-frequency cepstrum (MFC) is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency.

Mel-frequency cepstral coefficients (MFCCs) are coefficients that collectively make up an MFC.

The basic procedure to develop MFCCs is the following:

1. Convert from Hertz to Mel Scale
2. Take logarithm of Mel representation of audio
3. Take logarithmic magnitude and use Discrete Cosine Transformation
4. This result creates a spectrum over Mel frequencies as opposed to time, thus creating MFCCs

If the ML problem warrants MFCCs to be used, such as automatic speech recognition or denoising audio, the number of coefficients used is a hyperparameter of the model. Because of this, the number of MFCCs will vary based on the problem. However, for this example, we will use librosa's default 20 MFCCs.

Tonnetz

Networks or lattices of tones. The tonnetz tonal centroids — the "central" tones. These are features that help in Detecting Harmonic Change in Musical Audio or variances due to tones in audio.

Chroma

The underlying observation is that humans perceive two musical pitches as similar in color if they differ by an octave. Based on this observation, a pitch can be

separated into two components, which are referred to as tone height and chroma. Assuming the equal-tempered scale, one considers twelve chroma values represented by the set

{C, C♯, D, D♯, E , F, F♯, G, G♯, A, A♯, B}

that consists of the twelve pitch spelling attributes as used in Western music notation.

These 12 pitch variations are taken as the 12 features.

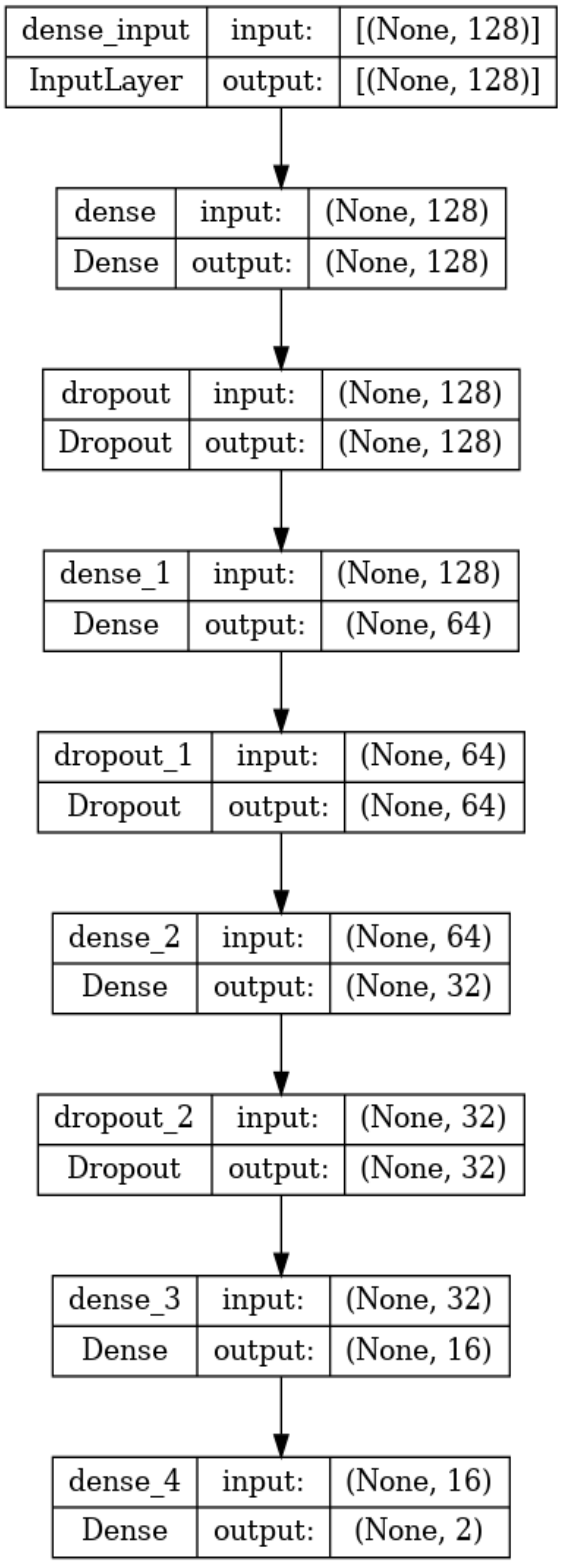
Note: We've only converted .mp3 to .npy based on MSF scale.
For conversion of other feature types you need to re-run preparation function containing entirety of the 13.4 GB dataset.

Training :

We have used the tensorflow.keras to train the model here is the architecture of the model. The architecture is defined in utils.py

Architecture :

- We designed our Model with 4 hidden with reLu as activation function.
- We kept reducing number of nodes in each layer by a factor of 2.
- Each layer has drop out value of 0.2
- Total number of trainable parameters used = 27,410
- Model is optimized with adam optimizer and trained with sparse categorical cross entropy with accuracy as metric.


dense_input | input: | [(None, 128)]
InputLayer | output: | [(None, 128)]
dense | input: | (None, 128)
Dense | output: | (None, 128)
dropout | input: | (None, 128)
Dropout | output: | (None, 128)
dense_1 | input: | (None, 128)
Dense | output: | (None, 64)
dropout_1 | input: | (None, 64)
Dropout | output: | (None, 64)
dense_2 | input: | (None, 64)
Dense | output: | (None, 32)
dropout_2 | input: | (None, 32)
Dropout | output: | (None, 32)
dense_3 | input: | (None, 32)
Dense | output: | (None, 16)
dense_4 | input: | (None, 16)
Dense | output: | (None, 2)

Models is trained and saved as model.pkl in ./results

Testing :

To run a test with a given audio file you can simply run python3 test.py {path_to_audio_file}

Or to record your mic you can simply run python3 test.py

Results :

We trained our model with 100 epochs and of batch size 64.

The value of loss of validation reduced from 0.39 to 0.16 when implemented with Early Stopping and reach accuracy of 94%.

	precision	recall	f1-score	support
0.0	0.93	0.94	0.94	3450
1.0	0.93	0.93	0.93	3244
accuracy			0.93	6694
macro avg	0.93	0.93	0.93	6694
weighted avg	0.93	0.93	0.93	6694

Observations :

The value of loss of validation reduced from 0.39 to 0.16 when implemented with

The value of accuracy is comparatively lower than previous data. This may be due to selective analysis of particular values which are more favorable for detection of gender such as lower mean fundamental frequency corresponds to man.

This is model is better for extended to voice recognition and sound recognition as the features constitute of entire distribution.

First Model is specially tailored for this problem statement. Where data is separable based on particular features.