# Speaker Identification Using Pitch and MFCC

This project shows a possible strategy to identify persons based on features extracted from voiced speech. Samples are extracted from <https://storage.googleapis.com/download.tensorflow.org/data/speech_commands_v0.01.tar.gz>

The wav files have been converted to flac and organized by imaginary persons. The features used have been the pitch and the mel frequency cepstrum coefficients. At the end of the feature extraction, a neural network is trained with a portion of these features, the model is then compared with a test set.

There are 10 speakers for 194 files. The project has been run on Matlab R2021a.

The project can be divided mainly in two parts: Feature Extraction and Model Training and Assessment.

## Feature Extraction

A particular data structure of Matlab has been used to extract the features: the audioDatastore which is a collection for audio files. So, at first all the audio data have been imported in this data structure and divided in two sets, one used for extracting the features and the other one not considered.

The audioDatastore is shuffled before being split in two sets, in order not to biased to much the training set for the neural network. A frequency sample of 16 KHz has been used for all the files; a windowLength of 30 ms has been used as long as an overlapping window of 25 ms.

Then the features are extracted from the audio store, those are: the cepstrum and the f0; another function is used to detect which part of the audio is a voiced speech and here the features are filtered if they belong to voiced part of the speech. In the function isVoicedSpeech a power threshold of -40 dB is used to capture all minimal sounds. The signal is divided into segments according to the window length and the overlapping rate and then all the parts considered speech (those which energy is more than -40 dB) are temporary stored. After, the zero crossing rate is calculated to detect which part of the speech is voiced, subsequently an array of logicals is returned which indicates which parts of the signal is voiced and speech. The process is made for all audio files. At the end of the cycle, the features are normalized on the same scale, i.e. are shrinked in values, since the network could be biased if the values are too distant from each other’s. Then the last part of the project is conducted: a simple neural network is feed with a good portion the features and with the same indices for the labels.

## Model Training and Assessment

First of all, the indexes for the training set and for the test set are obtained from the indexes of the labels; an hold out scheme is followed, taking 80% for the training set and a 20% for the test set.

A classification neural network model is then run with the indexes of the training set on the features and on the labels. A test accuracy is made after the model training. In average, an accuracy of 85% is achieved on the test set.

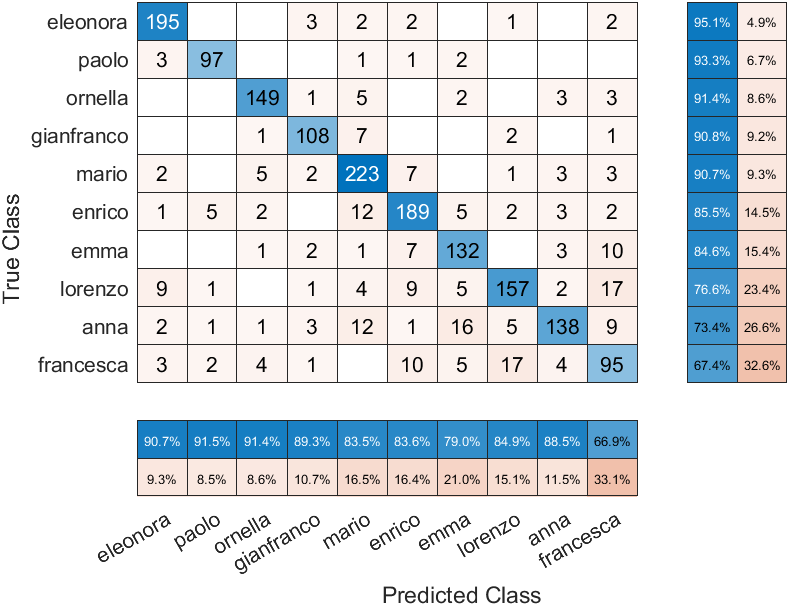


Figure . Percentage of true and predicted class on the test set.

Generally speaking, without fine tuning and optimization, the neural network works quite well. The percentage above are those regarding the features on the test set.