Processing steps for birds audio classification

We have two options:

* features for deep learning
  + see general\_frontend.py and train\_lstm.py
* knowledge-based features for regular machine learning
  + knowledge\_frontend.py and knowledge\_classifiers.py

# Part I – Using deep learning

DNN steps for mel spectrogram or similar features

1) Start with ogg folder called train\_audio and src in the same folder.

Interface gráfica do usuário, Aplicativo, Email

Descrição gerada automaticamente

2) The train\_audio folder has the files in format ogg and named as:

Interface gráfica do usuário

Descrição gerada automaticamente

3) Choose the name for a general folder in a parent folder. The default is “../outputs”. Create the file with list of ogg files and their labels. Also, note the histograms.

C:\github\birds\_dnn\src>python create\_label\_file.py ../train\_audio ../outputs/

Created output folder ../outputs/

Found 1048 files with extension ogg in folder ../train\_audio

Wrote file ../outputs/wavs\_labels.csv

Wrote file ../outputs/labels\_dictionary.json

The files ../ outputs/ wavs\_labels.csv and ../outputs\labels\_dictionary.json will be used in all simulations, with different frontends and ML models.

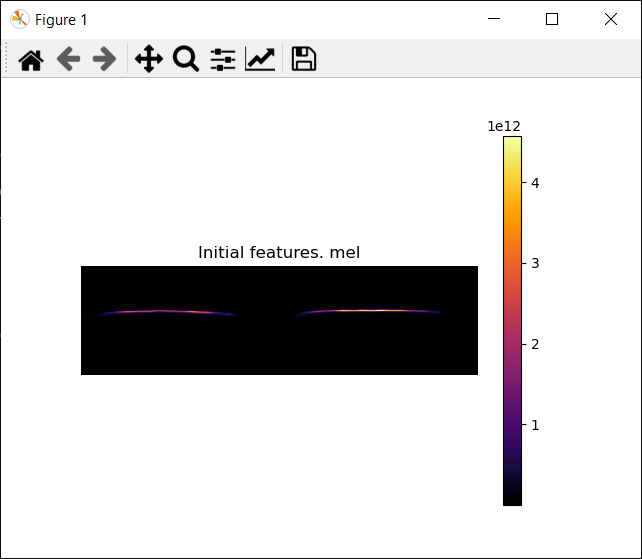
4) Choose the features (magnasco, mel, stft) and the dimensions D (frequency) and T (time), compose the output folder name with them and the name of the file with the features. For instance:

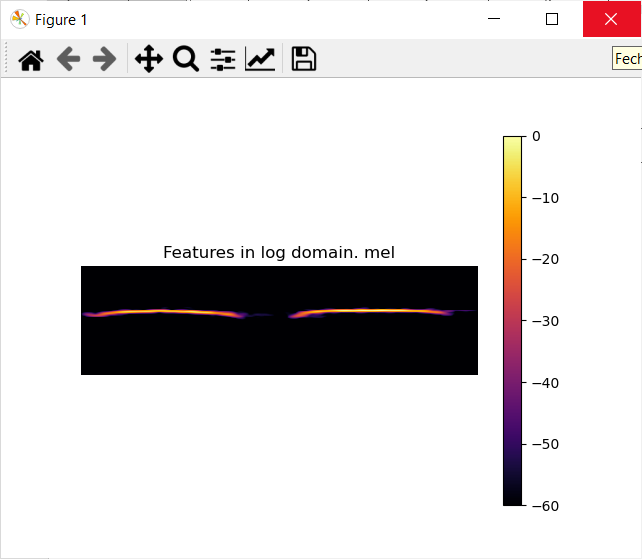
python feature-extraction\general\_frontend.py …

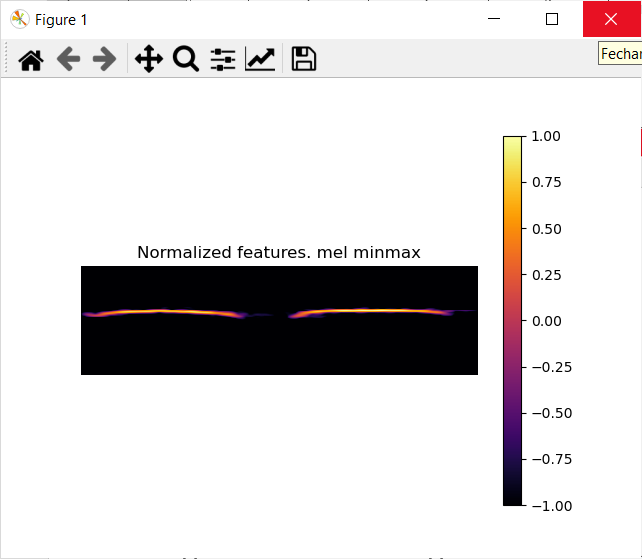
Files are written in the output folder

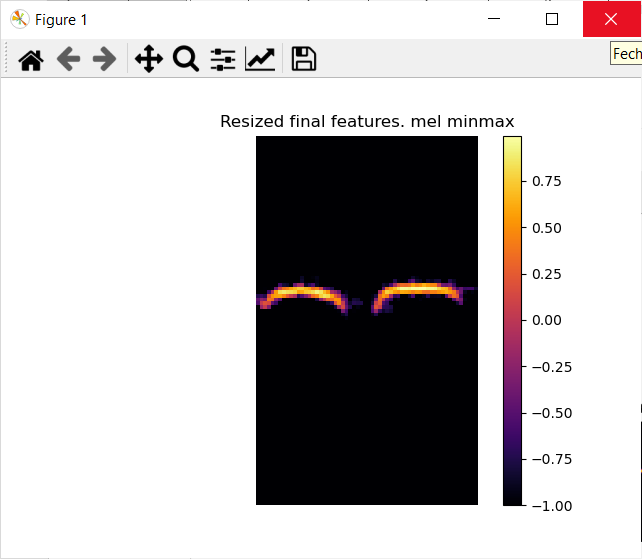
commandline\_args.txt has the command that was used.

You can observe the features being processed in the main 4 stages using the --show\_plot option. For instance:









5) Now train the ML model with mixed speakers:

python train\_lstm.py

# Part II – Using machine learning

Similar to first Part, but now use:

python .\knowledge\_frontend.py --output\_dir ..\knowledge\_features

and then:

python .\knowledge\_classifiers.py