

Elena Georgieva
Vocal Tuning Project
NYU DS 1008
Spring 2022
GitHub: <https://github.com/elenatheodora/TUNet>

```
In [2]: import os
        from tqdm import tqdm
        from random import randrange
        import gc
        from glob import glob
        import h5py
        import numpy as np
        import librosa
        import librosa.display
        import matplotlib.pyplot as plt
        from scipy import signal
```

```
In [3]: ## Function taken from Saksham Singh. Goal: convert my wave files to h
        # See: https://github.com/f90/Wave-U-Net
        # Groups work like dictionaries, datasets work like arrays
        def create_full_hdf_data(type='train'): ## change train and validati
            fs = 44100 # sampling rate
            A_LEN = 53929 # size of input datapoint
            B_LEN = 44377 # size of aligned output

            path = "/Users/elenageorgieva/Desktop/vtd/train/" # %% change trai

            raw_path = path + "raw" # input data
            tuned_path = path + "output" # output data
            save_path = path + 'temp/' # will not change

            raw_filenames = os.listdir(raw_path)
            all_filenames = [f for f in raw_filenames]

            with h5py.File(f'{save_path}{type}.hdf', "w") as f: # file called "
                for audio_file in all_filenames:
                    if (audio_file == ".DS_Store"): # weird error, ignore .DS_
                        continue
                    grp = f.create_group(audio_file)

                    x1, _ = librosa.load(path + 'raw/' + audio_file, sr=fs, mo
                    x2, _ = librosa.load(path + 'output/' + audio_file, sr=fs,

                    grp.create_dataset("RAW", shape=x1.shape, dtype=x1.dtype,
                    grp.create_dataset("TUNED", shape=x2.shape, dtype=x2.dtype

                    len_a = x1.shape[0]
                    len_b = x2.shape[0]

                    # Fix lengths by padding shorter signal
                    if (len_a > len_b):
                        librosa.util.fix_length(x2, size=len_a)
```

```
elif (len_b > len_a):
    librosa.util.fix_length(x1, size=len_b)

count = 0
a_list = []
b_list = []

for i in range(1024):
    start = randrange(len_a - A_LEN)
    pad = (A_LEN - B_LEN)//2 #4776
    if (start+A_LEN >= len_a) or (start + pad + B_LEN >= len_a):
        continue
    a_list.append([start, start+A_LEN])
    b_list.append([start+pad, start+pad+B_LEN])
    count += 1

grp.attrs["length"] = count # 1024
grp.attrs["fs"] = fs # 44100
a_arr = np.array(a_list) # a_arr = [[5502643 5556572]][6042
b_arr = np.array(b_list)

grp.create_dataset("raw_list", shape=a_arr.shape, dtype=a_arr.dtype)
grp.create_dataset("tuned_list", shape=b_arr.shape, dtype=b_arr.dtype)
```

In [4]: `create_full_hdf_data()` # run above function

In [6]: `## Sanity check, confirming hdf worked`
`hdf = h5py.File("/Users/elenageorgieva/Desktop/vtd/train/temp/train.hdf")`
`# print(list(hdf.keys())) # print file list`

In []: