

# Advanced Topics in Audio Processing Using Deep Learning

Barr Kirel, Dor Tirosh, Noam Shaked Cohen

January 3, 2026

## 1 Analytic Part

### 1.1 Continuous Signals

#### 1.1.1 Time convolution property

By definitions of convolutions and Fourier transform:

$$\mathcal{F}\{x_1(t) * x_2(t)\} = \int_{-\infty}^{\infty} \left[ \int_{-\infty}^{\infty} x_1(\tau)x_2(t-\tau)d\tau \right] e^{-j\omega t} dt \quad (1)$$

Interchanging the order of integration:

$$= \int_{-\infty}^{\infty} x_1(\tau) \left[ \int_{-\infty}^{\infty} x_2(t-\tau)e^{-j\omega t} dt \right] d\tau \quad (2)$$

Let  $u = t - \tau$ , so  $t = u + \tau$  and  $dt = du$ .

$$= \int_{-\infty}^{\infty} x_1(\tau) \left[ \int_{-\infty}^{\infty} x_2(u)e^{-j\omega(u+\tau)} du \right] d\tau = \int_{-\infty}^{\infty} x_1(\tau)e^{-j\omega\tau} \left[ \int_{-\infty}^{\infty} x_2(u)e^{-j\omega u} du \right] d\tau \quad (3)$$

This is exactly  $X_2(w)$ ! so we got

$$= \int_{-\infty}^{\infty} x_1(\tau)e^{-j\omega\tau} X_2(\omega) d\tau = X_2(\omega) \int_{-\infty}^{\infty} x_1(\tau)e^{-j\omega\tau} d\tau = X_2(\omega)X_1(\omega) \quad (4)$$

So overall we got:

$$\mathcal{F}\{x_1(t) * x_2(t)\} = X_1(\omega)X_2(\omega) \quad (5)$$

#### 1.1.2 Linearity property

$$\mathcal{F}(ax_1(t) + bx_2(t)) = \int_{-\infty}^{\infty} [ax_1(t) + bx_2(t)] e^{-j\omega t} dt \quad (6)$$

From linearity of integrals:

$$= \int_{-\infty}^{\infty} ax_1(t)e^{-j\omega t} dt + \int_{-\infty}^{\infty} bx_2(t)e^{-j\omega t} dt = a \int_{-\infty}^{\infty} x_1(t)e^{-j\omega t} dt + b \int_{-\infty}^{\infty} x_2(t)e^{-j\omega t} dt \quad (7)$$

Which is exactly

$$aX_1(\omega) + bX_2(\omega) \quad (8)$$

#### 1.1.3 Scaling property

$$\mathcal{F}(x(at)) = \int_{-\infty}^{\infty} x(at)e^{-j\omega t} dt \quad (9)$$

Applying substitution of  $u = at$ , so  $dt = \frac{1}{a}du$  and  $t = \frac{1}{a}u$ . the limits of the integration do not change as  $a$  is positive

$$= \int_{-\infty}^{\infty} x(u)e^{-j\omega \frac{1}{a}u} \frac{1}{a}du = \frac{1}{a} \int_{-\infty}^{\infty} x(u)e^{-j\omega \frac{1}{a}u} du = \frac{1}{a} \int_{-\infty}^{\infty} x(u)e^{-j(\frac{\omega}{a})u} du \quad (10)$$

Which is exactly the definition of the Fourier Transform with frequency  $\frac{\omega}{a}$ :

$$\mathcal{F}(x(at)) = \frac{1}{a} X\left(\frac{\omega}{a}\right) \quad (11)$$

#### 1.1.4 Time shifting property

$$\mathcal{F}(x(t - t_0)) = \int_{-\infty}^{\infty} x(t - t_0) e^{-j\omega t} dt \quad (12)$$

We can substitute  $t$  to  $u = t - t_0$  (note that  $du = dt$ ) and get

$$= \int_{-\infty}^{\infty} x(u) e^{-j\omega(u+t_0)} du = \int_{-\infty}^{\infty} x(u) e^{-j\omega u} e^{-j\omega t_0} du = e^{-j\omega t_0} \int_{-\infty}^{\infty} x(u) e^{-j\omega u} du = e^{-j\omega t_0} X(\omega) \quad (13)$$

How does it affect the amplitude:

$$|e^{-j\omega t_0} X(\omega)| = |e^{-j\omega t_0}| \cdot |X(\omega)| \quad (14)$$

Since the magnitude of a complex exponential is always unity ( $|e^{j\theta}| = 1$  for any real  $\theta$ ):

$$= |X(\omega)| \quad (15)$$

So **time shifting does not change the amplitude spectrum of the signal**.

As for the phase: Using the property that the angle of a product is the sum of the angles ( $\angle(ab) = \angle a + \angle b$ ):

$$\angle [e^{-j\omega t_0} X(\omega)] = \angle(e^{-j\omega t_0}) + \angle X(\omega) \quad (16)$$

The angle of  $e^{-j\omega t_0}$  is simply  $(-\omega t_0)$ :

$$\angle Y(\omega) = -\omega t_0 + \angle X(\omega) \quad (17)$$

So **time shifting adds a linear phase term  $(-\omega t_0)$  to the original phase spectrum**. This means the phase is shifted by an amount proportional to the frequency  $\omega$ .

#### 1.1.5 Fourier transform of unit gate (rect) function

Applying the Fourier Transform definition on  $\text{rect}(\frac{t}{\tau})$ :

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt = \int_{-\tau/2}^{\tau/2} 1 \cdot e^{-j\omega t} dt \quad (18)$$

Solving the integral:

$$X(\omega) = \left[ \frac{e^{-j\omega t}}{-j\omega} \right]_{-\tau/2}^{\tau/2} \quad (19)$$

$$= \frac{1}{-j\omega} \left( e^{-j\omega\tau/2} - e^{-j\omega(-\tau/2)} \right) \quad (20)$$

$$= \frac{1}{j\omega} \left( e^{j\omega\tau/2} - e^{-j\omega\tau/2} \right) \quad (21)$$

Using  $\sin(\theta) = \frac{e^{j\theta} - e^{-j\theta}}{2j}$ :

$$= \frac{2}{\omega} \left( \frac{e^{j\omega\tau/2} - e^{-j\omega\tau/2}}{2j} \right) \quad (22)$$

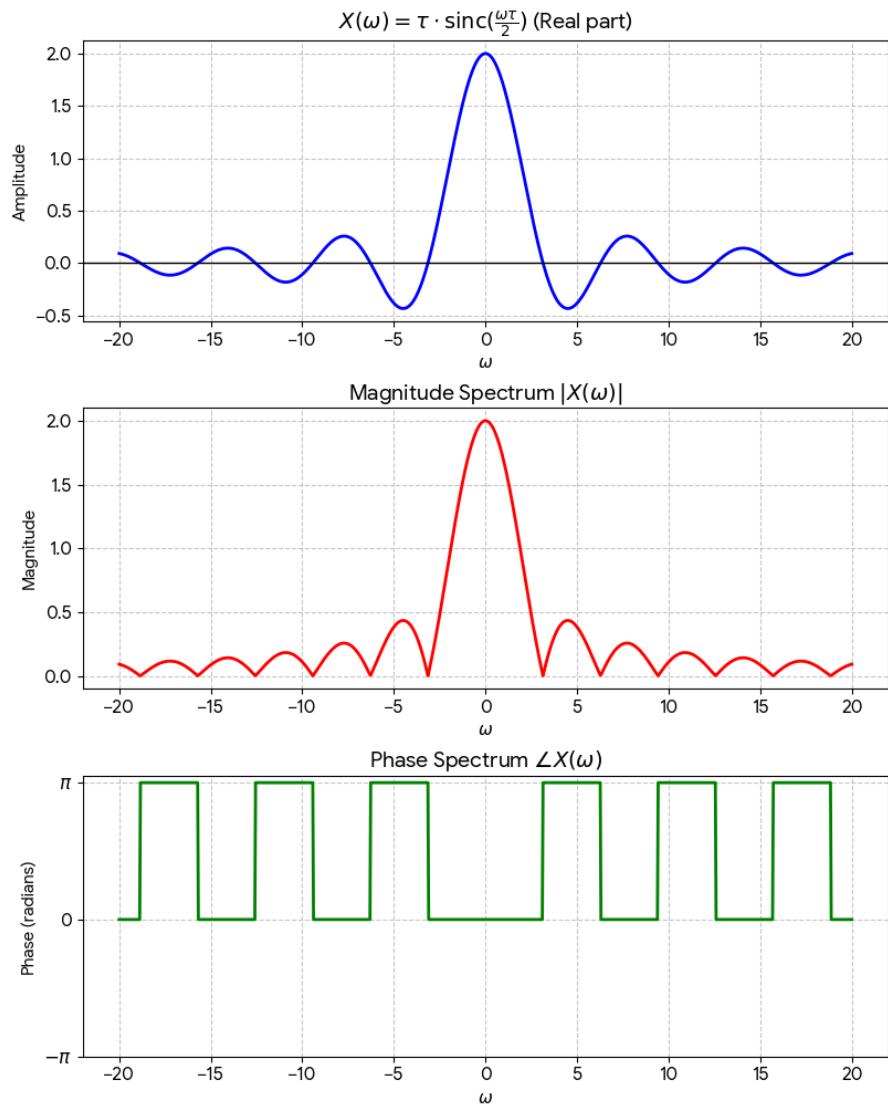
$$= \frac{2}{\omega} \sin\left(\frac{\omega\tau}{2}\right) \quad (23)$$

We can multiply and divide by  $\tau/2$ :

$$= \tau \cdot \frac{\sin\left(\frac{\omega\tau}{2}\right)}{\frac{\omega\tau}{2}} \quad (24)$$

$$X(\omega) = \tau \text{sinc}\left(\frac{\omega\tau}{2}\right) \quad (25)$$

In order to draw it, note that: Power of  $X(\omega)$  is just  $|X(\omega)|$  And that because  $X(\omega)$  is totally real,  $\angle X(\omega)$  is either 0 or  $\pi$ , depends on whether its positive or negative.



## 1.2 Fourier Series

### 1.2.1 Fourier series of delta function

Given a unit impulse train  $x(t) = \delta_{T_0}(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT_0)$ .

The exponential Fourier series coefficients are defined as:

$$D_n = \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} x(t) e^{-jn\omega_0 t} dt \quad (26)$$

Considering the interval  $[-T_0/2, T_0/2]$ , the function  $x(t)$  contains only the single impulse at  $t = 0$ , so  $x(t) = \delta(t)$ .

$$D_n = \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} \delta(t) e^{-jn\omega_0 t} dt \quad (27)$$

Using the sifting property of the delta function  $\int \delta(t)f(t)dt = f(0)$ :

$$D_n = \frac{1}{T_0} \cdot e^{-jn\omega_0(0)} = \frac{1}{T_0} \cdot 1 \quad (28)$$

So:

$$D_n = \frac{1}{T_0} \quad (29)$$

Since  $x(t)$  is periodic, we can represent it as a Complex Exponential Fourier Series:

$$x(t) = \sum_{n=-\infty}^{\infty} D_n e^{jn\omega_0 t} \quad (30)$$

We previously calculated the Fourier coefficients  $D_n$ :

$$D_n = \frac{1}{T_0} \quad (31)$$

To find  $X(\omega)$ , we will be substituting this right back into the series:

$$x(t) = \sum_{n=-\infty}^{\infty} \frac{1}{T_0} e^{jn\omega_0 t} \quad (32)$$

We apply the Fourier Transform to both sides:

$$X(\omega) = \mathcal{F} \left\{ \sum_{n=-\infty}^{\infty} e^{jn\omega_0 t} \right\} \quad (33)$$

$$X(\omega) = \frac{1}{T_0} \sum_{n=-\infty}^{\infty} \mathcal{F} \{ e^{jn\omega_0 t} \} \quad (34)$$

We use the standard Fourier Transform for a complex exponential:

$$\mathcal{F} \{ e^{j\omega_c t} \} = 2\pi \delta(\omega - \omega_c) \quad (35)$$

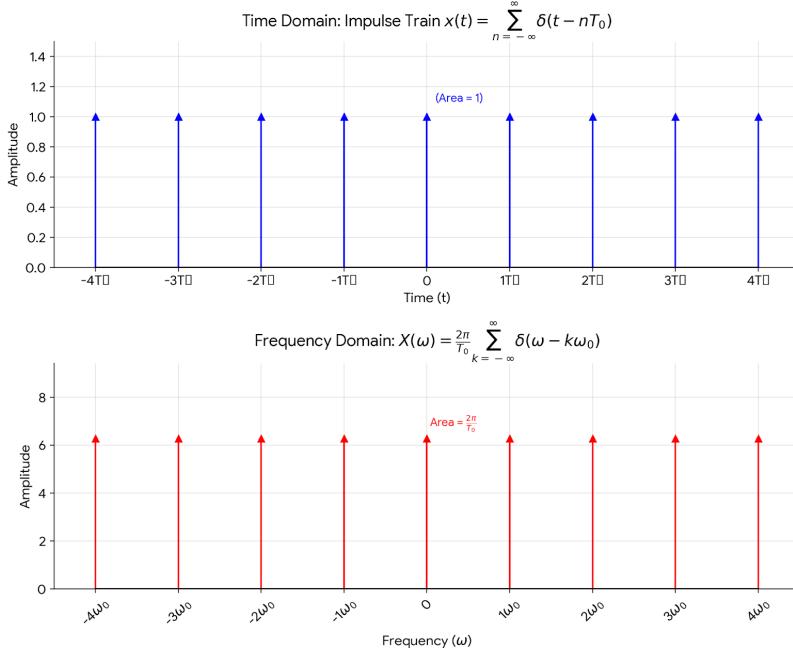
Therefore:

$$\mathcal{F} \{ e^{jn\omega_0 t} \} = 2\pi \delta(\omega - n\omega_0) \quad (36)$$

So:

$$X(\omega) = \frac{1}{T_0} \sum_{k=-\infty}^{\infty} 2\pi \delta(\omega - k\omega_0) = X(\omega) = \frac{2\pi}{T_0} \sum_{k=-\infty}^{\infty} \delta(\omega - k\omega_0) \quad (37)$$

$T_0$  is the time duration between each two consecutive deltas in the time domain, and the interval between each  $D_i$  and  $D_{i+1}$  in the frequency domain is  $\omega_0 = \frac{2\pi}{T_0}$



### 1.3 spectrum of a specific continuous, periodic function

The function is a sum of time shifted rects:

$$x(t) = \sum_{k=-\infty}^{\infty} rect\left(\frac{t + 2\pi k}{\pi}\right) \quad (38)$$

In other words:

$$x(t) = rect\left(\frac{t}{\pi}\right) * \sum_{k=-\infty}^{\infty} \delta(t - 2\pi k) \quad (39)$$

It's spectrum is:

$$X(\omega) = \mathcal{F} \left[ rect\left(\frac{t + 2\pi k}{\pi}\right) * \sum_{k=-\infty}^{\infty} \delta(t - 2\pi k) \right] \stackrel{\text{from (5)}}{=} \mathcal{F} \left[ rect\left(\frac{t + 2\pi k}{\pi}\right) \right] \mathcal{F} \left[ \sum_{k=-\infty}^{\infty} \delta(t - 2\pi k) \right] \quad (40)$$

Using (25) and (8) we get:

$$X(\omega) = \pi sinc\left(\frac{\pi\omega}{2}\right) \sum_{k=-\infty}^{\infty} \mathcal{F}[\delta(t - 2\pi k)] \stackrel{(37)}{=} \pi sinc\left(\frac{\pi\omega}{2}\right) \cdot \frac{2\pi}{2\pi} \sum_{k=-\infty}^{\infty} \delta(\omega - k) \quad (41)$$

### 1.4 Unit step function analysis

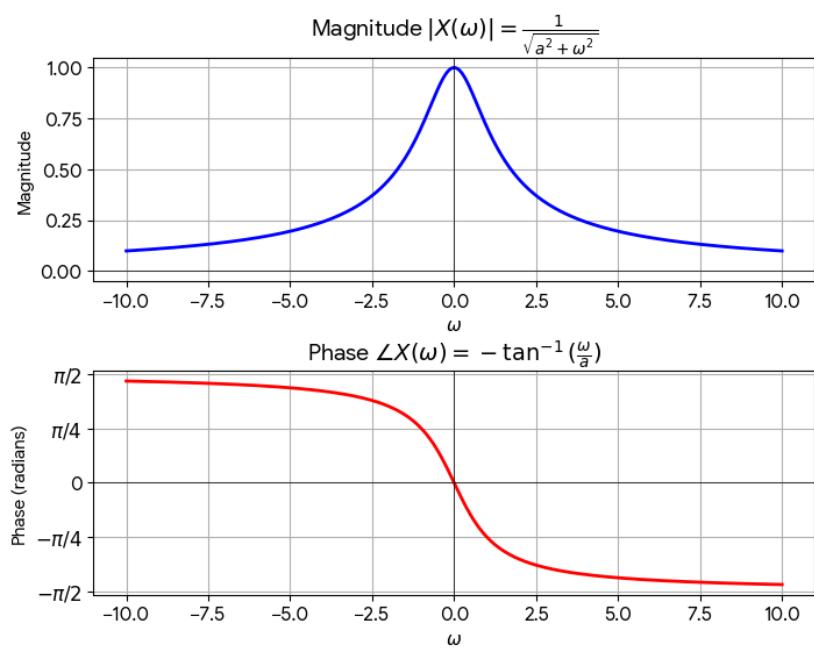
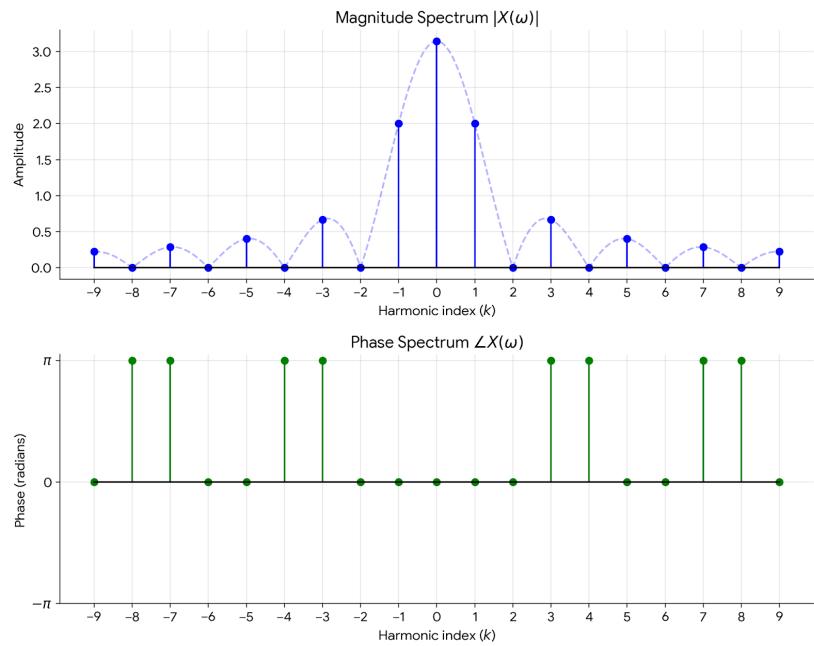
$$X(\omega) = \mathcal{F}[e^{-at} u(t)] = \int_{-\infty}^{\infty} e^{-at} u(t) e^{-j\omega t} dt = \int_0^{\infty} e^{-(a+j\omega)t} dt \stackrel{\text{integration}}{=} \left[ \frac{e^{-(a+j\omega)t}}{-(a+j\omega)} \right]_0^{\infty} \quad (42)$$

Given that  $a > 0$ , the upper limit vanishes:

$$X(\omega) = 0 - \frac{1}{-(a+j\omega)} = \frac{1}{a+j\omega} \quad (43)$$

Its magnitude and phase are:

$$|X(\omega)| = \sqrt{\frac{1}{a-j\omega} \cdot \frac{1}{a+j\omega}} = \frac{1}{\sqrt{a^2 + \omega^2}} \quad (44)$$



$$\angle X(\omega) \stackrel{\text{angle identity}}{=} -\angle \frac{1}{X(\omega)} = -\angle(a + j\omega) = -\tan^{-1}\left(\frac{\omega}{a}\right) \quad (45)$$

Analyzing the filter type by checking limits:

$$\lim_{\omega \rightarrow 0} |X(\omega)| = \frac{1}{a}, \quad \lim_{\omega \rightarrow \infty} |X(\omega)| = 0 \quad (46)$$

Since low frequencies pass and high frequencies are attenuated, this is a Low Pass Filter.

## 2 Technical Part

In this section we will shortly answer the answer related to the technical part that was implemented using `main.py`. The outputs of this section would be provided along side the code in the `outputs` directory, this directory will include both the resulting audio files and higher resolution diagrams. Also most of the arbitrary chosen parameters in this exercise are configurable via the arguments to the script (use `--help` to see the possible options).

### 2.1 Initial Analysis

This section focuses on actually loading the the audio and drawing some plots for it. TBH we didn't see any clear difference between taking every even sample or using the `scipy.signal.resample` function, neither in sound quality or in the plots. By reading the documentation of `scipy.signal.resample` it looks like it should provide better anti-aliasing but on the other hand assumes the signal is periodic.

#### 2.1.1 Resampled audio plots

The following diagrams (1,2) were generated using the command

```
python ./main.py --audio-file ./recordings/combined.wav --question a
```

We can visibly see that the Pitch Contour is missing on areas without speech meaning we couldn't find a matching pitch for that section.

### 2.2 Adding noise

We were now requested to add noise to our signal. The following diagrams (3, 4, 5) show the original audio vs the noised audio and then the spectrogram of the noise and the spectrogram of the noised audio. The diagrams were generated using the command:

```
python ./main.py --audio-file ./recordings/combined.wav --question b:
```

### 2.3 Spectral Subtraction

In this section we were requested to perform spectral subtraction in order to clear the noise we just add, we choose -17.95 dB as our noise threshold and used two different methods to clear the noise:

1. Classic averaging of the noise and then reduction
2. Interpolation of the noise between unnoised section

The rest of the exercise used the first method but we added both the code and the resulting diagrams for the second method as well. The following diagrams (6, 7) were generated using the following command:

```
python ./main.py --audio-file ./recordings/combined.wav --question c:
```

The following diagram (8) were generated using the following command:

```
python ./main.py --audio-file ./recordings/combined.wav --question c  
--use-interpolate-for-spectracl:
```

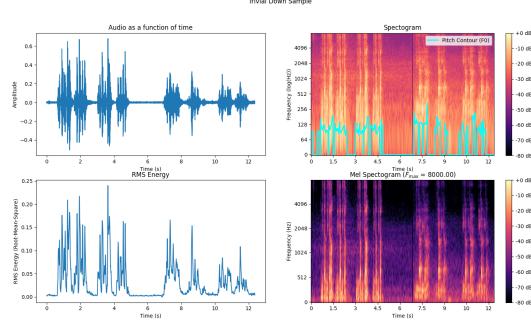


Figure 1: Plot of 1.c.trivial\_down\_sample.wav

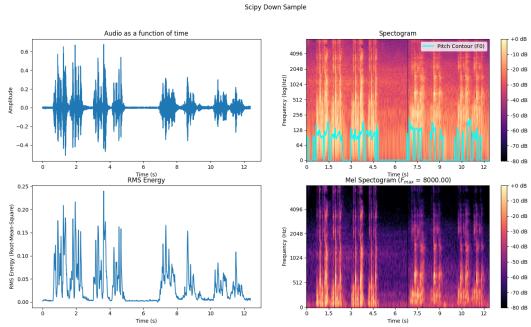


Figure 2: Plot of 1.c.scipy\_down\_sample.wav

## 2.4 Auto Gain Control

In this section we were requested to apply Auto Gain Control on the previous cleaned audio. We chose a target rms of (-6 dB) and chose not to amplify areas detected as noise. The following diagrams (9, 10) were generated using the following command:

```
python ./main.py --audio-file ./recordings/combined.wav --question d
```

## 2.5 Time Stretching

This section was implemented by:

1. Performing `stft` on the input
2. Creating an interpolation function using `scipy.interpolate.interp1d`
3. Creating samples using the formula `1.5x`
4. Performing `istft` to get the final result

For fun we also generated different speeds The following diagram (11, 12, 13) were generated using the following command:

```
python ./main.py --audio-file ./recordings/combined.wav --question e
--speed-up-factor X
```

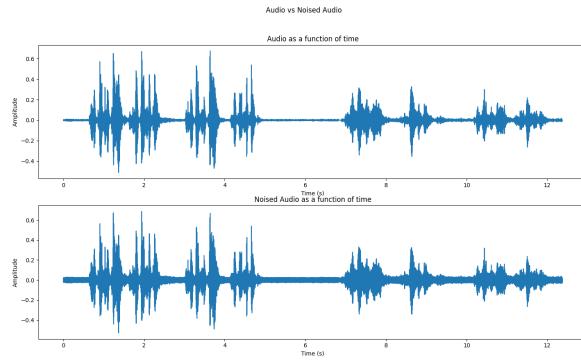


Figure 3: Plot of `1.c.scipy_down_sample.wav` vs `2.noisy_sample.wav`

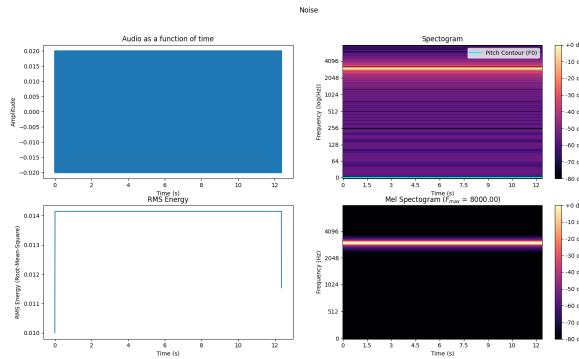


Figure 4: Plot of `stationary_noise.wav`

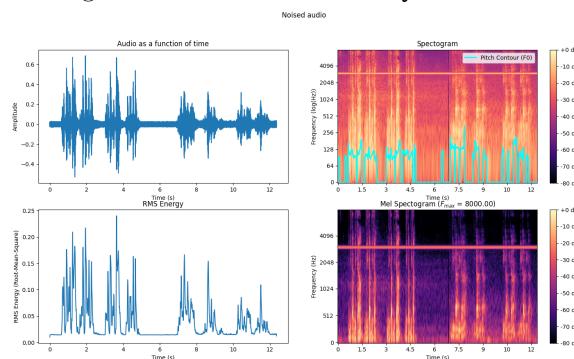


Figure 5: Plot of `2.noisy_sample.wav`

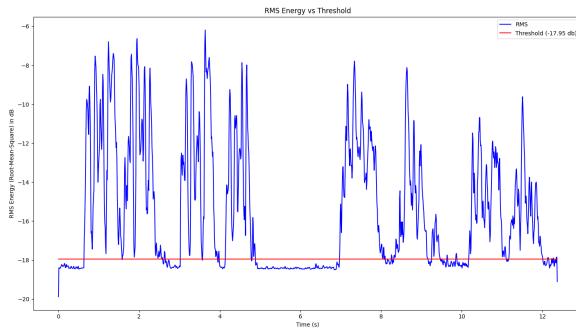


Figure 6: Plot of rms of 1.noisy\_sample.wav vs the chosen threshold

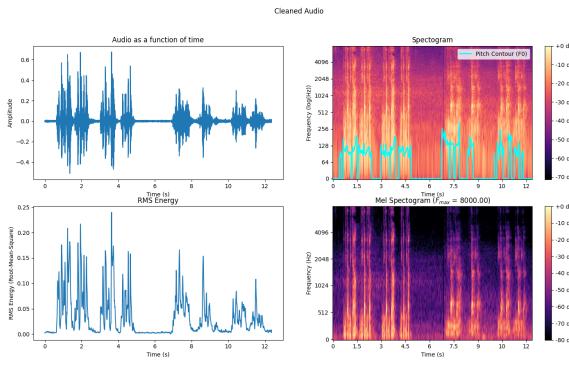


Figure 7: Plot of 3.cleaned\_audio.wav

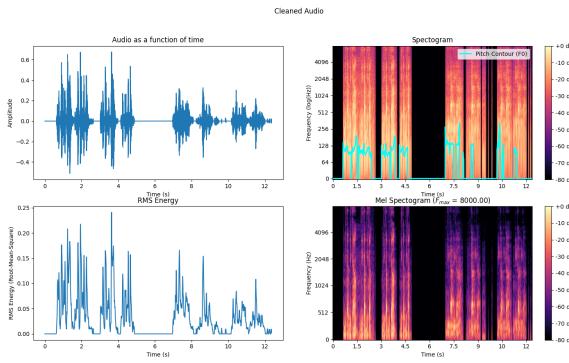


Figure 8: Plot of rms of 3.cleaned\_audio\_interpolation.wav vs the chosen threshold

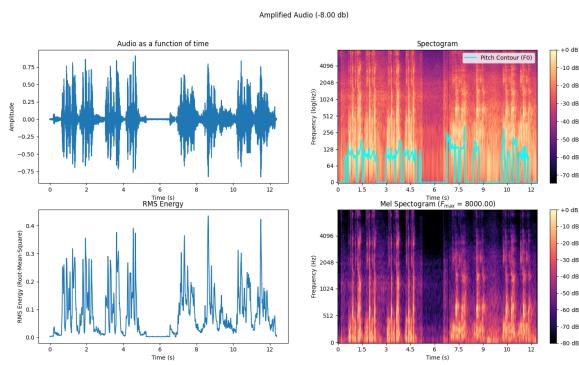


Figure 9: Plot of `4_cleaned_and_amplified_audio.wav`

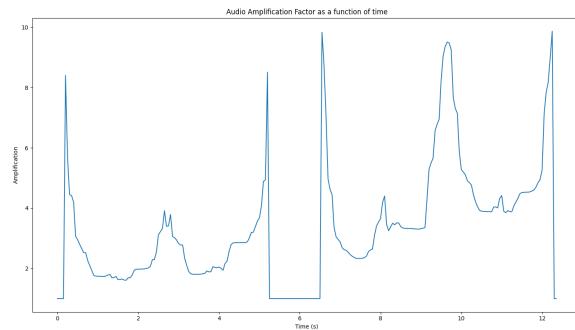


Figure 10: Plot of the amplification `3_cleaned_audio.wav` experienced as a function of time

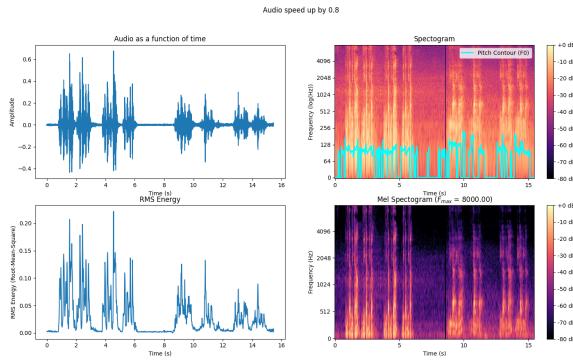


Figure 11: Plot of 5\_speed\_up\_by\_0.8\_audio.wav

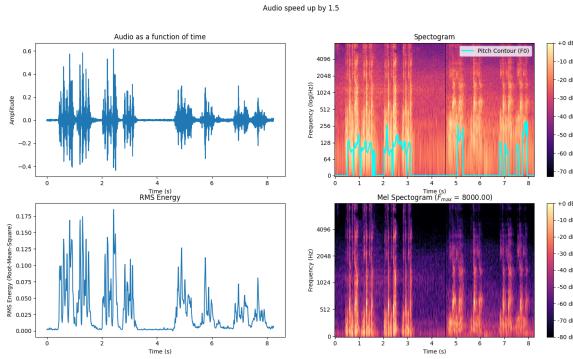


Figure 12: Plot of 5\_speed\_up\_by\_1.5\_audio.wav

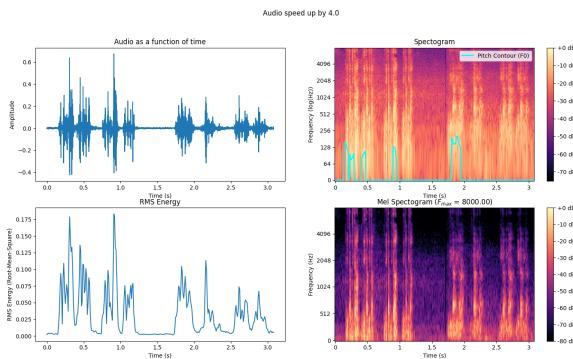


Figure 13: Plot of 5\_speed\_up\_by\_4\_audio.wav