

Лабораторная работа 10

Чернышев Ярослав

1 июня 2021 г.

Оглавление

1	Задание 10.1	2
2	Задание 10.2	9

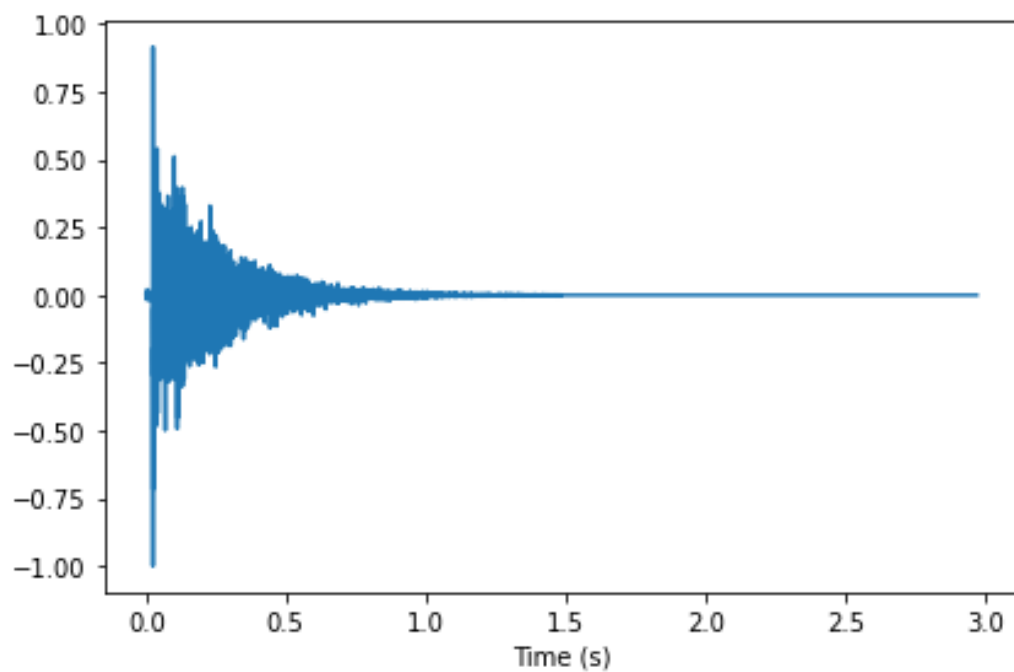
Глава 1

Задание 10.1

В разделе учебника 10.4 была рассмотрена операция линейной свертки, а в разделе 10.3 выполнялась циклическая. Преобразовать ее в линейную возможно при помощи вставки нулей в конец. Возьмем уже рассмотренные звуки выстрела и скрипки, обрежем их до 2^{16} и дополним нулями до 2^{17} :

```
1 from thinkdsp import read_wave
2 response = read_wave('180960__kleeb__gunshot.wav')
3 start = 0.12
4 response = response.segment(start=start)
5 response.shift(-start)
6 response.truncate(2**16)
7 response.zero_pad(2**17)
8 response.normalize()
9 response.plot()
10 decorate(xlabel='Time (s)')
```

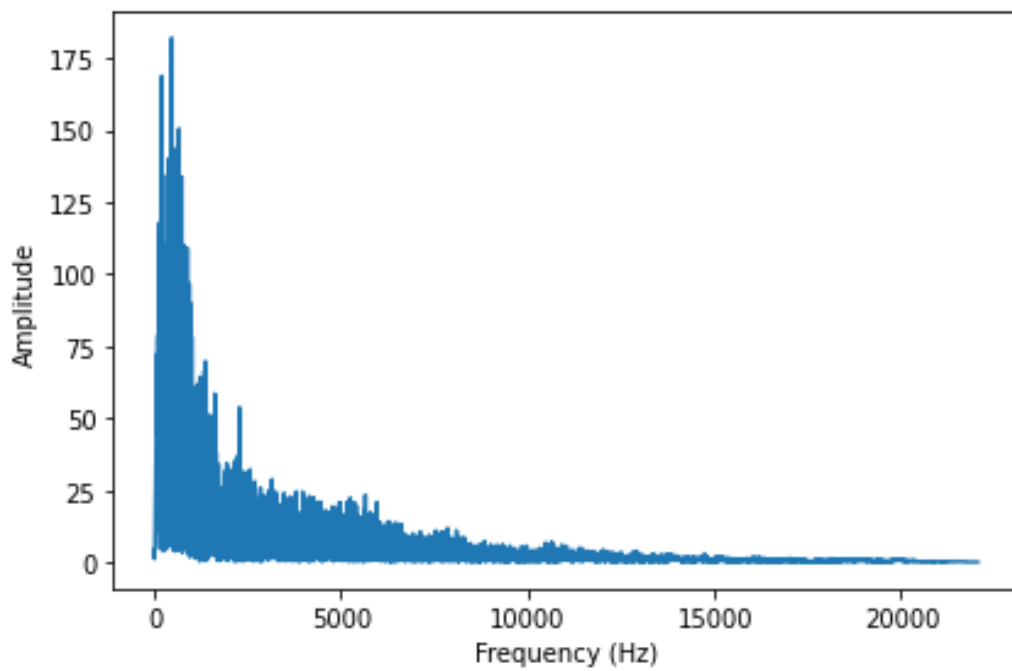
Результат:



Далее, вычислим спектр сигнала:

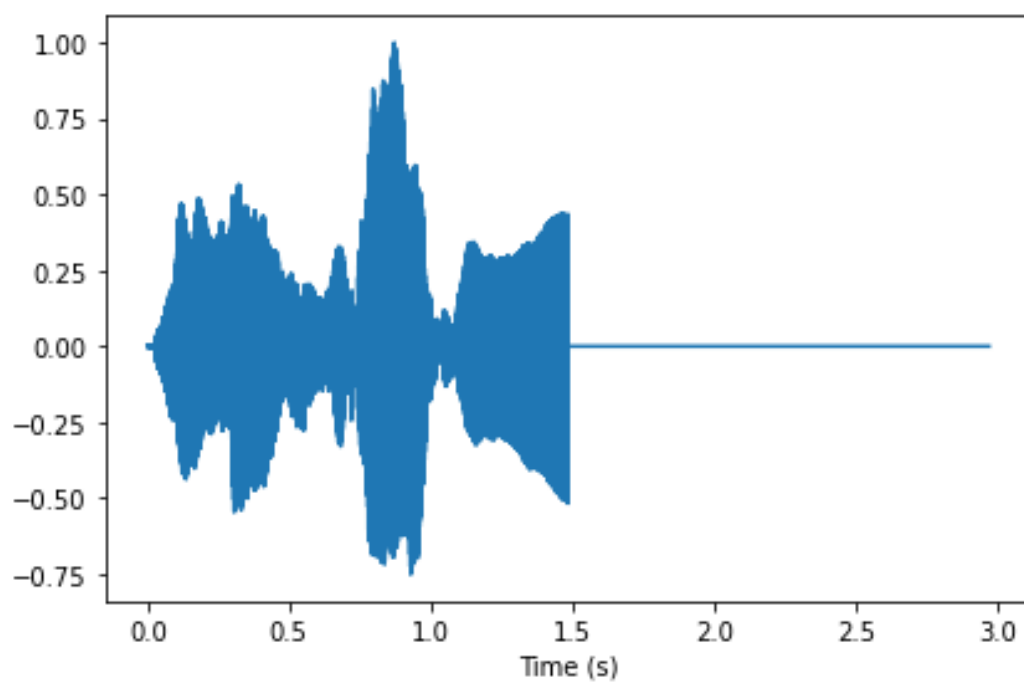
```
1 transfer = response.make_spectrum()  
2 transfer.plot()  
3 decorate(xlabel='Frequency (Hz)', ylabel='Amplitude')
```

Результат:

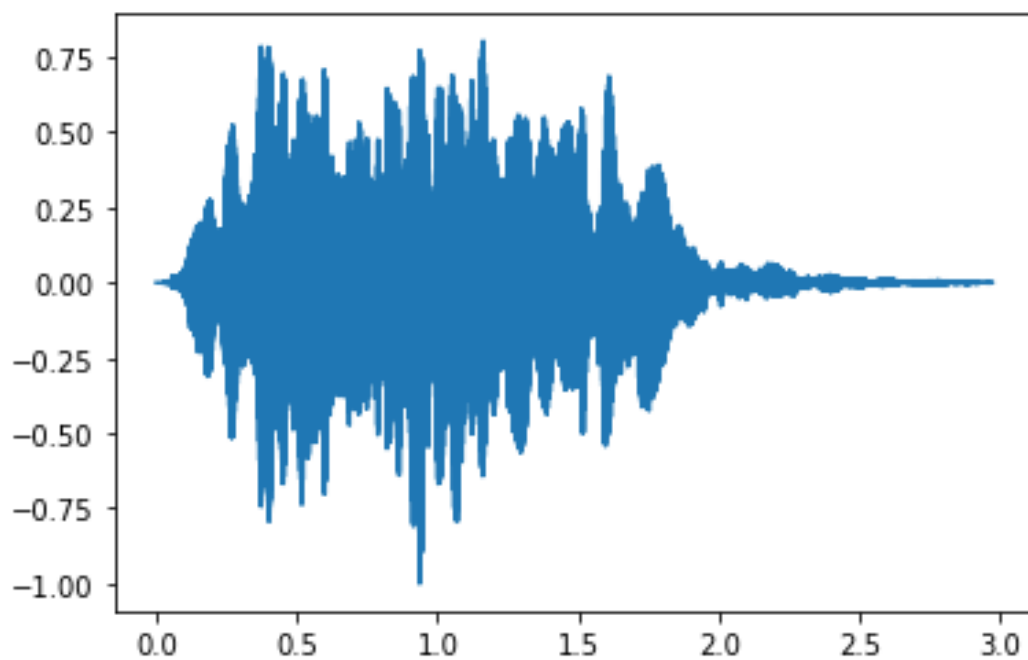


Возьмём другой сигнал:

```
1 violin = read_wave('92002__jcveliz__violin-origional.wav')
2 start = 0.11
3 violin = violin.segment(start=start)
4 violin.shift(-start)
5 violin.truncate(2**16)
6 violin.zero_pad(2**17)
7 violin.normalize()
8 violin.plot()
9 decorate(xlabel='Time (s)')
```

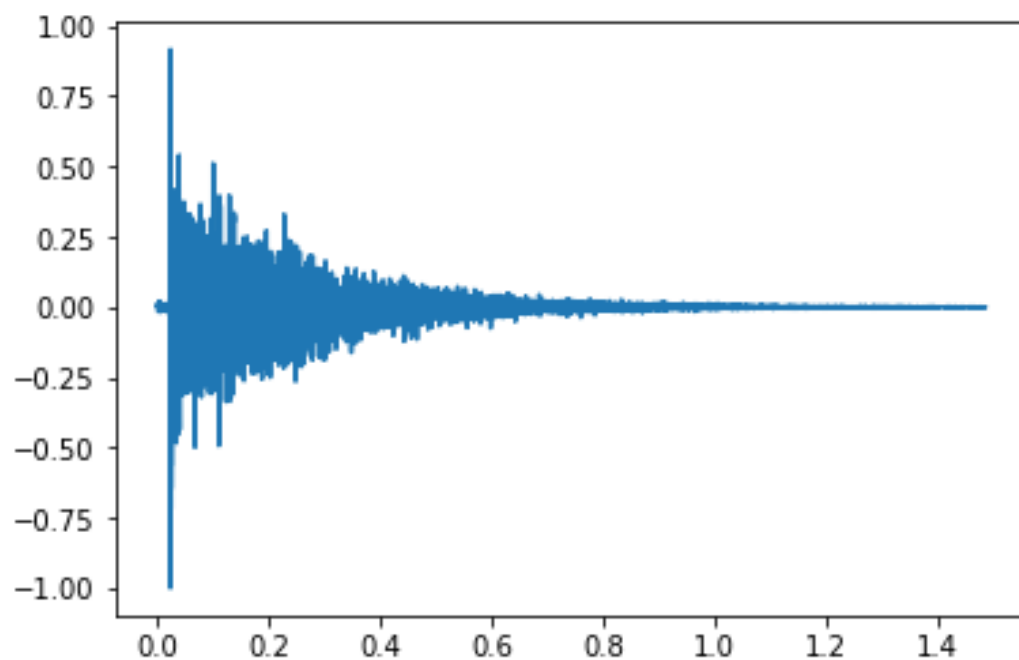


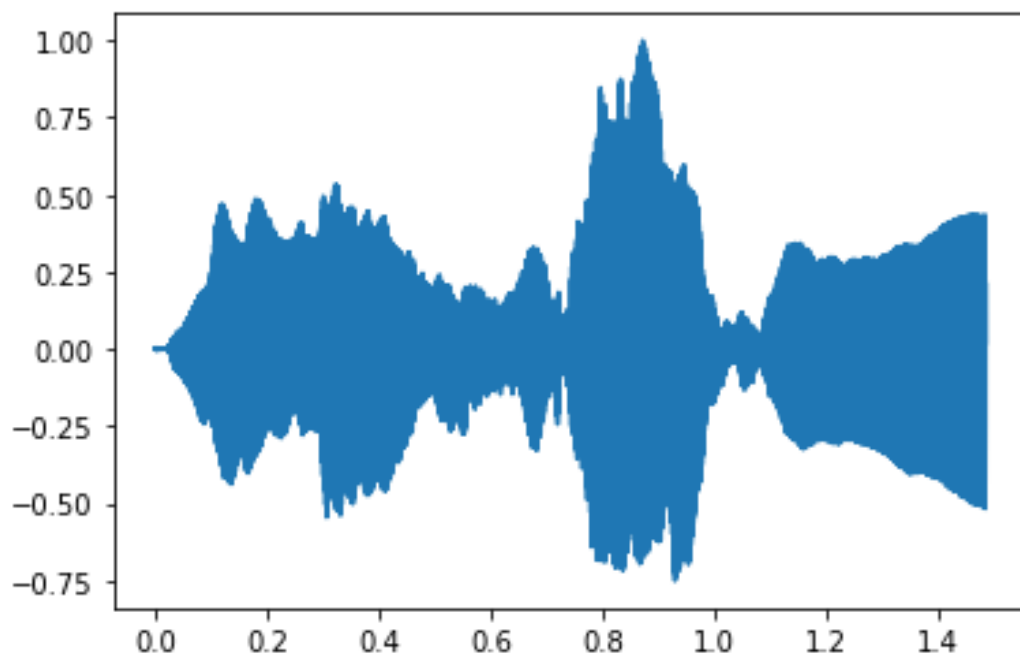
Вычислим на его основе спектр, умножим на передаточную функцию, преобразуем обратно в волну и нормализуем:



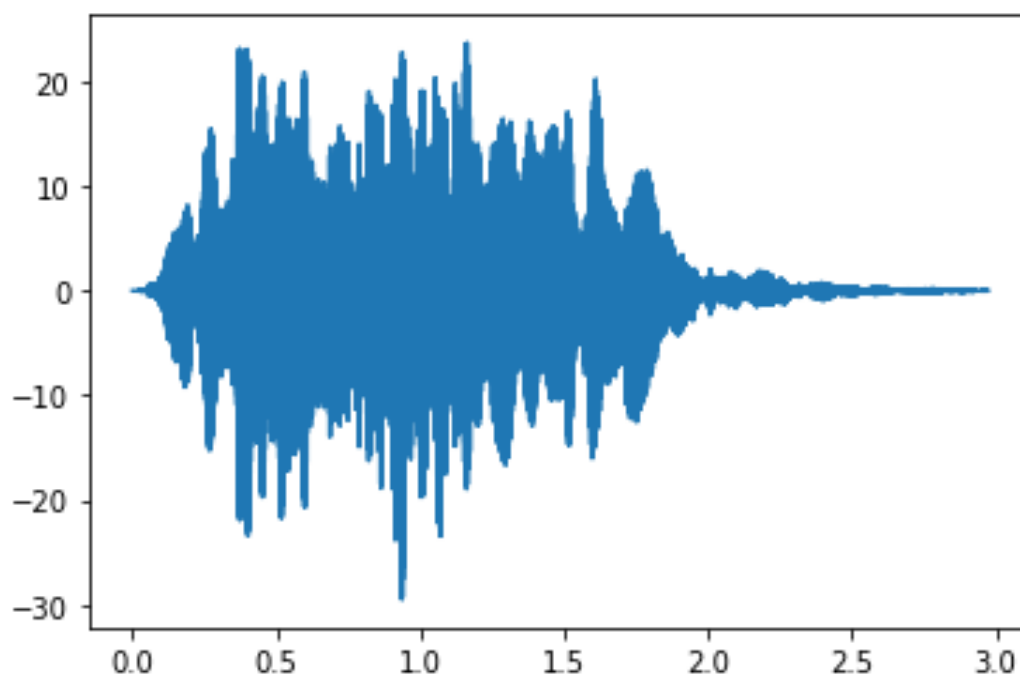
При прослушивании созданной на основе этой волны записи лишней ноты не слышно.

Теперь попробуем достичь аналогичного результата с помощью `fftconvolve`. Сначала, уберём лишние нули у двух рассмотренных ранее сигналов с помощью `truncate`:



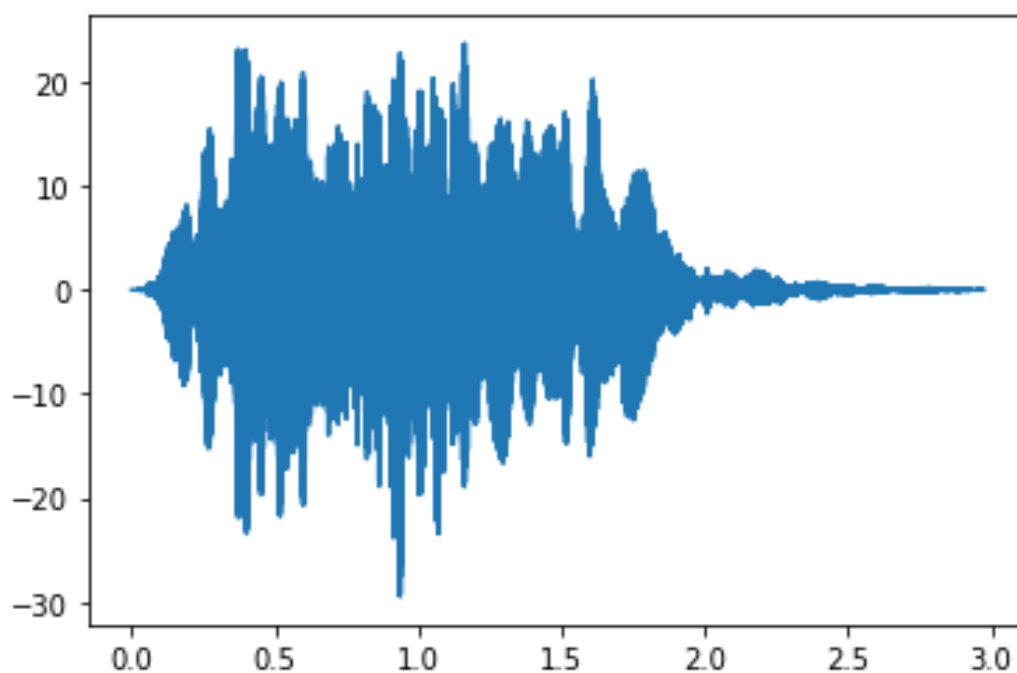


К полученным сигналам применим `convolve`:



Как можно видеть, результаты схожи. Далее, протестируем `scipy.signal.fftconvolve`:

```
1 ys = scipy.signal.fftconvolve(violin.ys, response.ys)
2 output3 = Wave(ys, framerate=violin.framerate)
```



И вновь схожий результат. `output2.maxdiff(output3)` показывает, что разница между двумя последними результатами - $2.3092638912203256 \times 10^{-14}$.

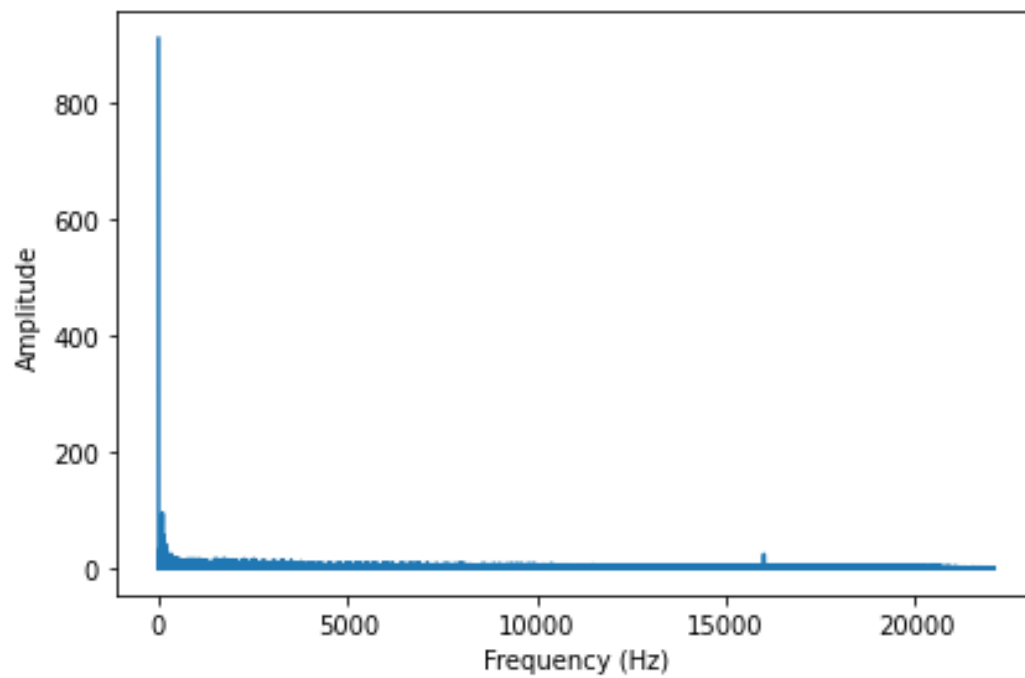
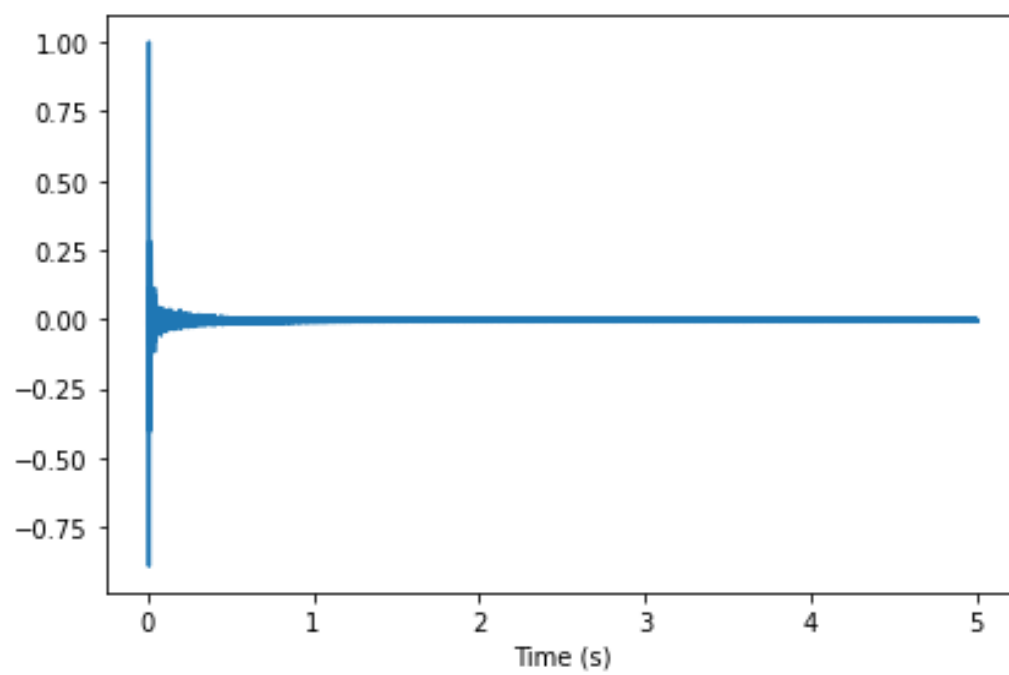
Глава 2

Задание 10.2

Возьмём запись, которая будет использована в качестве импульсной характеристики:

```
1 response = read_wave('stalbans_a_mono.wav')
2 start = 0
3 duration = 5
4 response = response.segment(duration=duration)
5 response.shift(-start)
6 response.normalize()
7 response.plot()
8 decorate(xlabel='Time (s)')
```

Таким образом, её волна и ДПФ соответственно:

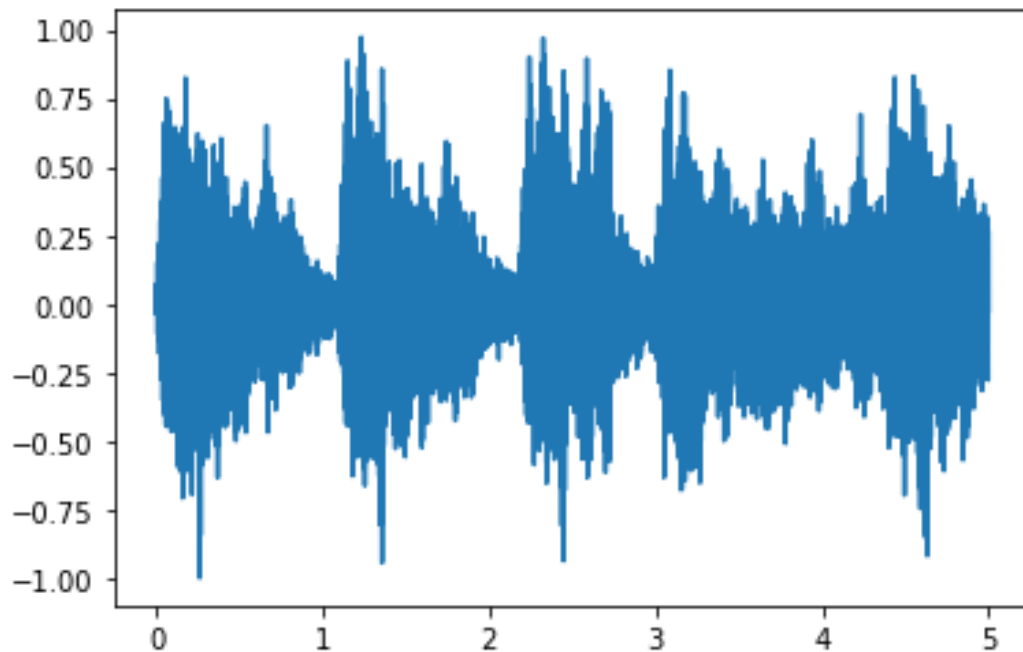


Далее, возьмём вторую запись:

```

1 wave = read_wave('170255__dublie__trumpet.wav')
2 start = 0.0
3 wave = wave.segment(start=start)
4 wave.shift(-start)
5 wave.truncate(len(response))
6 wave.normalize()
7 wave.plot()
8 decorate(xlabel='Time (s)')

```

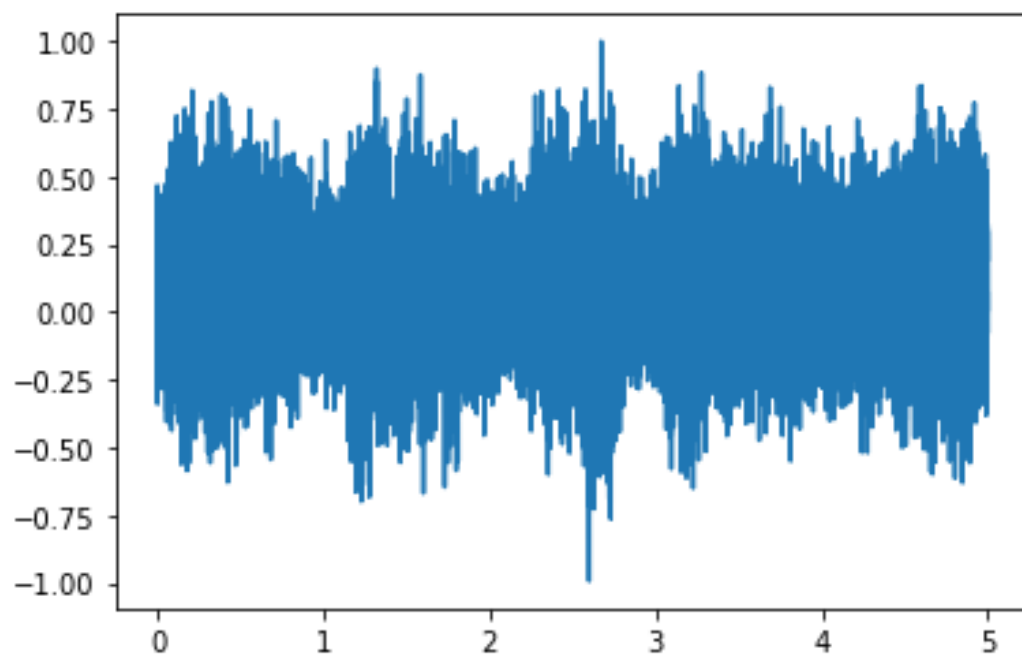


Можно видеть, что длины звуков и частотные компоненты фрагментов также совпадают. На основе перемноженных спектра и передаточной функции получаем новую волну:

```

1 spectrum = wave.make_spectrum()
2 output = (spectrum * transfer).make_wave()
3 output.normalize()

```



Кроме этого, эту же операцию можно провести с помощью `convolve`:

```
1 convolved2.normalize()  
2 convolved2.make_audio()
```