

Лабораторная работа №1
Звуки и сигналы

Смирнов Никита

5 апреля 2021 г.

Оглавление

1	Упражнение 1.1	4
2	Упражнение 1.2	5
2.1	Скачивание звука и работа с ним	5
2.2	Спектр звука	7
2.3	Фильтрация звука	8
3	Упражнение 1.3	10
3.1	Создание сложного сигнала	10
3.2	Добавление новой частоты	12
4	Упражнение 1.4	13
5	Выводы	15

Список иллюстраций

2.1	Исходный звук	6
2.2	Исходный звук	6
2.3	Спектр сегмента звука	7
2.4	Увеличенный масштаб	8
2.5	Спектр сегмента звука	9
3.1	Спектр сегмента звука	11
3.2	Визуализация сегмента звука	12
4.1	Визуализация ускоренного звука	14

Листинги

2.1	Загрузка и прослушивание звука	5
2.2	Визуализация звука	5
2.3	Изменение и прослушивание звука	6
2.4	Визуализация укороченного звука	6
2.5	Спектр сегмента звука	7
2.6	Основные и доминирующие частоты	7
2.7	Фильтрация и воспроизведение звука	8
2.8	Визуализация фильтрации	8
3.1	Создание сложного сигнала из 4 элементов	10
3.2	Воспроизведение сложного сигнала	11
3.3	Визуализация сигнала	11
3.4	Добавление новой частоты и воспроизведение	12
4.1	Загрузка и прослушивание звука	13
4.2	Функция stretch	13
4.3	Прослушивание ускоренного звука	13
4.4	Визуализация ускоренного звука	13

Глава 1

Упражнение 1.1

В данном упражнении нам нужно открыть `chap01.ipynb`, прочитать пояснения и запустить примеры. Поэтому я просто изучил все примеры с комментариями.

Глава 2

Упражнение 1.2

2.1 Скачивание звука и работа с ним

С предложенного нам сайта скачан звук проезжающей машины. Ссылка на соответствующий звук:

<https://freesound.org/people/gmetaxas/sounds/347662/>.

Далее звук был загружен, прослушан, и получена его визуализация.

```
1 wave =  
    thinkdsp.read_wave('347662__gmetaxas__motor-sound-road-no-effect.wav')  
2 wave.normalize()  
3 wave.make_audio()
```

Листинг 2.1: Загрузка и прослушивание звука

```
1 wave.plot()
```

Листинг 2.2: Визуализация звука

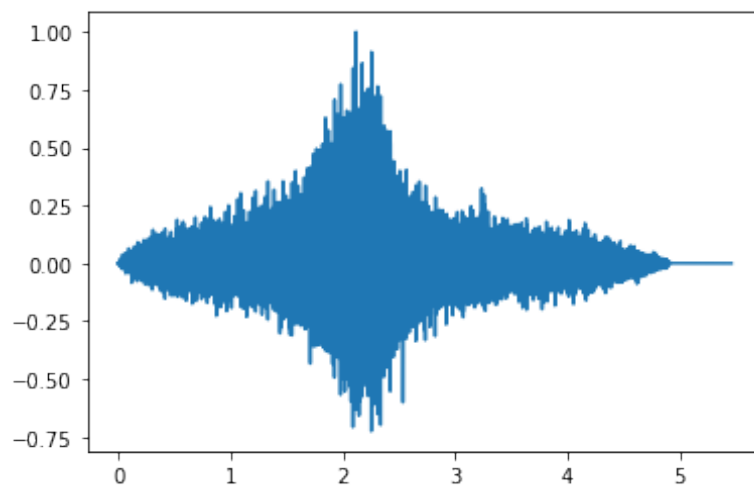


Рис. 2.1: Исходный звук

Берем полусекундный сегмент.

```
1 segment = wave.segment(start=1, duration=2)
2 segment.make_audio()
```

Листинг 2.3: Изменение и прослушивание звука

```
1 segment.plot()
```

Листинг 2.4: Визуализация укороченного звука

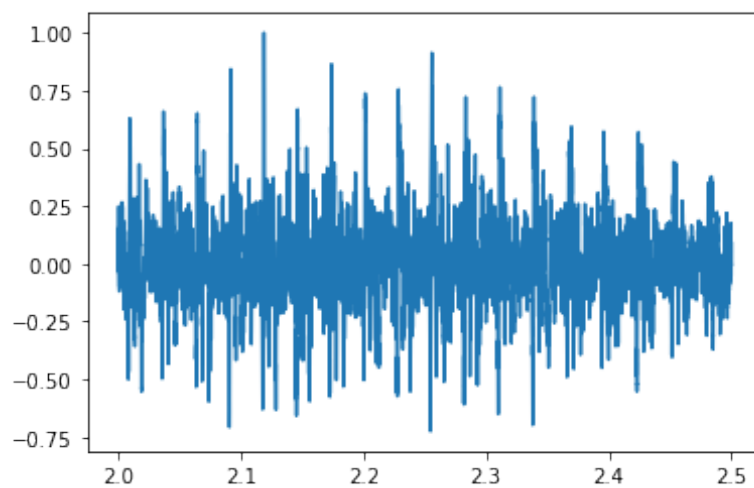


Рис. 2.2: Исходный звук

2.2 Спектр звука

Теперь рассмотрим спектр нашего полусекундного сегмента звука.

```
1 spectrum = segment.make_spectrum()  
2 spectrum.plot(high=5000)
```

Листинг 2.5: Спектр сегмента звука

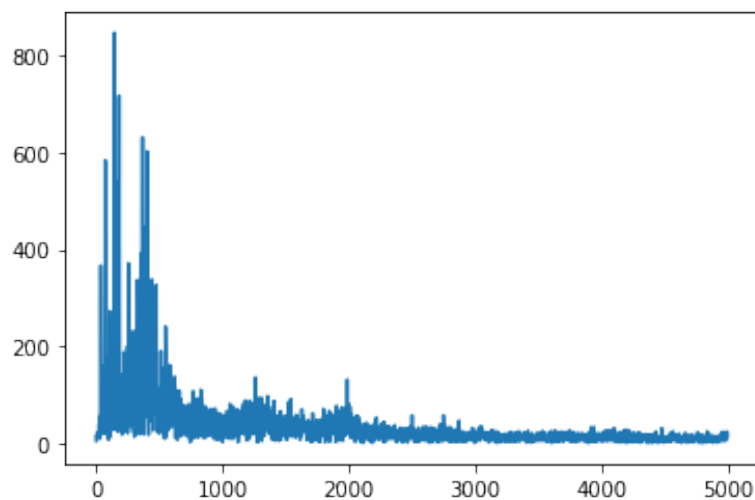


Рис. 2.3: Спектр сегмента звука

Увеличим масштаб.

```
1 spectrum = segment.make_spectrum()  
2 spectrum.plot(high=1000)
```

Листинг 2.6: Основные и доминирующие частоты

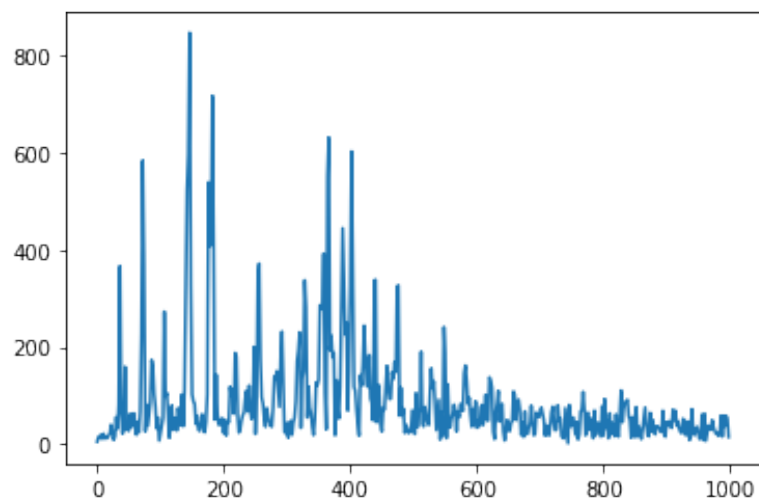


Рис. 2.4: Увеличенный масштаб

2.3 Фильтрация звука

Применим фильтр нижних частот.

```
1 spectrum.low_pass(400)
2 spectrum.make_wave().make_audio()
```

Листинг 2.7: Фильтрация и воспроизведение звука

```
1 spectrum.make_wave().plot()
```

Листинг 2.8: Визуализация фильтрации

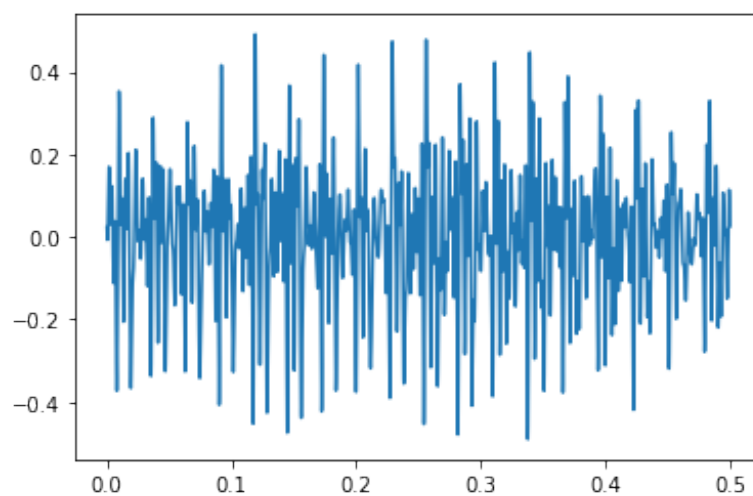


Рис. 2.5: Спектр сегмента звука

Видно, что график изменился, а звук стал, как из туннеля.

Глава 3

Упражнение 1.3

3.1 Создание сложного сигнала

Нужно создать сложный сигнал из объектов `SinSignal` и `CosSignal`.

```
1 import thinkdsp
2
3 signal = (thinkdsp.SinSignal(freq=300, amp=0.1) +
4           thinkdsp.SinSignal(freq=400, amp=1.5) +
5 thinkdsp.CosSignal(freq=300, amp=1.5) +
6           thinkdsp.CosSignal(freq=100, amp=1.8))
7 signal.plot()
```

Листинг 3.1: Создание сложного сигнала из 4 элементов

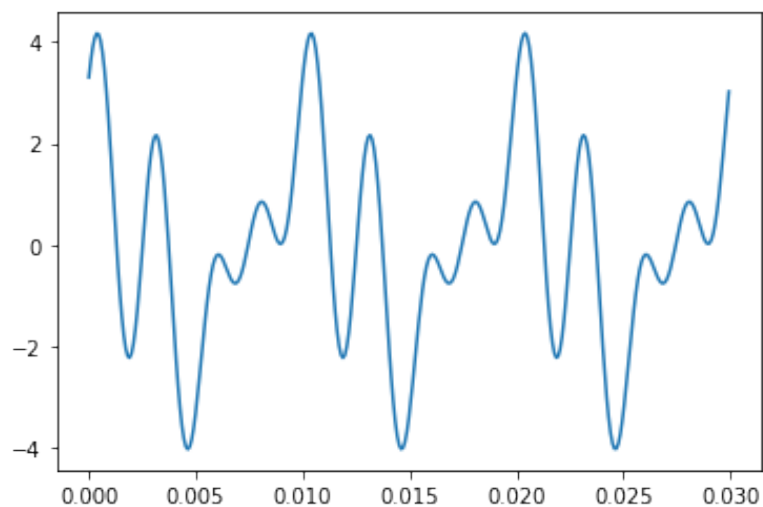


Рис. 3.1: Спектр сегмента звука

Теперь нужно получить звук.

```

1 wave = signal.make_wave(duration=2)
2 wave.apodize()
3 wave.make_audio()

```

Листинг 3.2: Воспроизведение сложного сигнала

Наж звук схож со звуком при звонке. Выведем спектр полученного звука.

```

1 spectrum = wave.make_spectrum()
2 spectrum.plot(high=1000)

```

Листинг 3.3: Визуализация сигнала

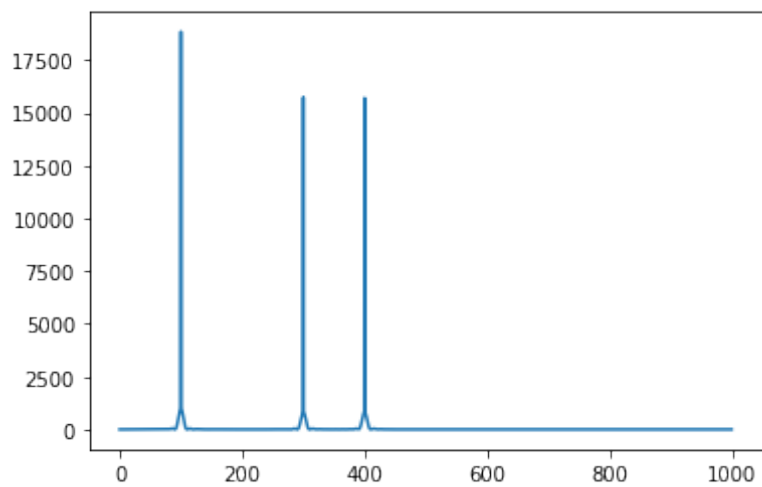


Рис. 3.2: Визуализация сегмента звука

3.2 Добавление новой частоты

Изменим наш сигнал.

```
1 signal += thinkdsp.SinSignal(freq=1000)
2 signal.make_wave().make_audio()
```

Листинг 3.4: Добавление новой частоты и воспроизведение

Теперь слышно добавленную новую частоту, при чём более высокую, потому что `freq=1000`. Теперь звук более похож на набор цифр при звонке через стационарный телефон.

Глава 4

Упражнение 1.4

Подготовим звук.

```
1 wave =  
    thinkdsp.read_wave('sounds/414062__felix-blume__machine-gears.wav')  
2 wave.normalize()  
3 wave.make_audio()
```

Листинг 4.1: Загрузка и прослушивание звука

Теперь сделаем функцию `stretch`.

```
1 def stretch(wave, factor):  
2     wave.ts *= factor  
3     wave. framerate /= factor
```

Листинг 4.2: Функция `stretch`

Попробуем прослушать полученный звук, введя 0.25.

```
1 stretch(wave3, 0.25)  
2 wave.make_audio()
```

Листинг 4.3: Прослушивание ускоренного звука

По таймеру в колабе время сократился с 5 до 2 секунд.

```
1 wave.plot()
```

Листинг 4.4: Визуализация ускоренного звука

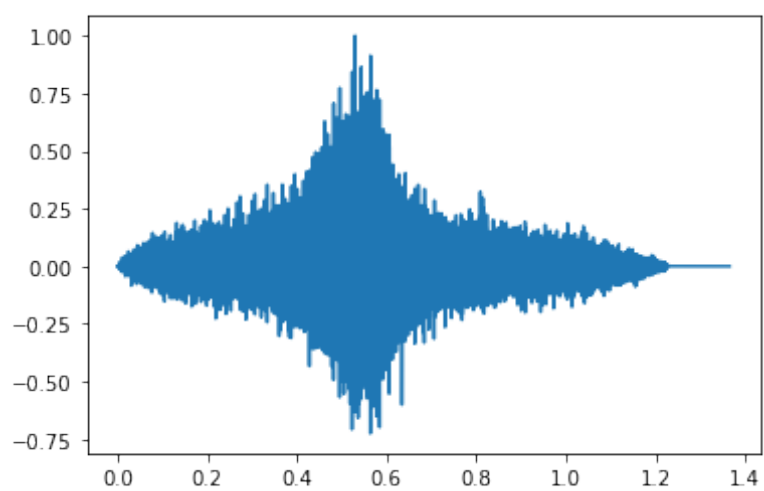


Рис. 4.1: Визуализация ускоренного звука

Глава 5

Выводы

Во время выполнения лабораторной работы получены навыки работы со звуками, волнами и спектрами.