

# Лабораторная работа № 3. Аперiodические сигналы.

3530901/80201, Шелаев Н. Р.

30 мая 2021 г.

# Оглавление

<b>1</b>	<b>Линейный chirp</b>	<b>4</b>
<b>2</b>	<b>Спектрограмма</b>	<b>6</b>
<b>3</b>	<b>Утечки</b>	<b>8</b>
<b>4</b>	<b>Упражнения</b>	<b>10</b>
4.1	Задание 1 . . . . .	10
4.2	Задание 2 . . . . .	11
4.3	Задание 3 . . . . .	12
4.4	Задание 4 . . . . .	13
4.5	Задание 5 . . . . .	14
4.6	Задание 6 . . . . .	15
<b>5</b>	<b>Вывод</b>	<b>20</b>

# Список иллюстраций

1.1	Начало сигнала . . . . .	4
1.2	Конец сигнала . . . . .	5
1.3	Спектр чирпа . . . . .	5
2.1	Спектрограмма чирпа . . . . .	7
3.1	Полученный сигнал . . . . .	8
3.2	Сегмент сигнала с утечкой . . . . .	9
3.3	Исправление утечки . . . . .	9
4.1	Исправление утечки различными функциями . . . . .	10
4.2	Спектрограмма полученного сигнала . . . . .	12
4.3	Спектр пилообразного чирпа . . . . .	13
4.4	Спектрограмма сигнала глиссандо . . . . .	14
4.5	Спектрограмма полученного сигнала . . . . .	15
4.6	Спектрограмма 5 гласных звуков . . . . .	16
4.7	Сегмент со звуком А . . . . .	17
4.8	Сегмент со звуком Э . . . . .	17
4.9	Сегмент со звуком И . . . . .	18
4.10	Сегмент со звуком О . . . . .	18
4.11	Сегмент со звуком У . . . . .	19

# Листинги

1.1	Линейный чирп . . . . .	4
1.2	Строим спектр чирпа . . . . .	5
2.1	Функция для построения спектрограммы . . . . .	6
3.1	Спектр сигнала . . . . .	8
3.2	Исправление сигнала функцией Хэмминга . . . . .	9
4.1	Борьба с утечкой . . . . .	10
4.2	Создание нового класса . . . . .	11
4.3	Пилообразный чирп . . . . .	12
4.4	Получение сигнала . . . . .	13
4.5	Новый класс для изменения частоты сигнала . . . . .	14
4.6	Получение требуемого сигнала . . . . .	14
4.7	Получение сигнала . . . . .	15
4.8	Сегменты для каждого звука . . . . .	16

# Глава 1

## Линейный чирп

Строим линейный чирп.

```
1         from thinkdsp import Chirp
2
3         signal = Chirp(start = 220, end = 880)
4         wave = signal.make_wave(duration = 2)
5         wave.segment(start = 0, duration = 0.01).plot()
6         wave.segment(start = 0.9, duration = 0.01).plot()
7
```

Листинг 1.1: Линейный чирп

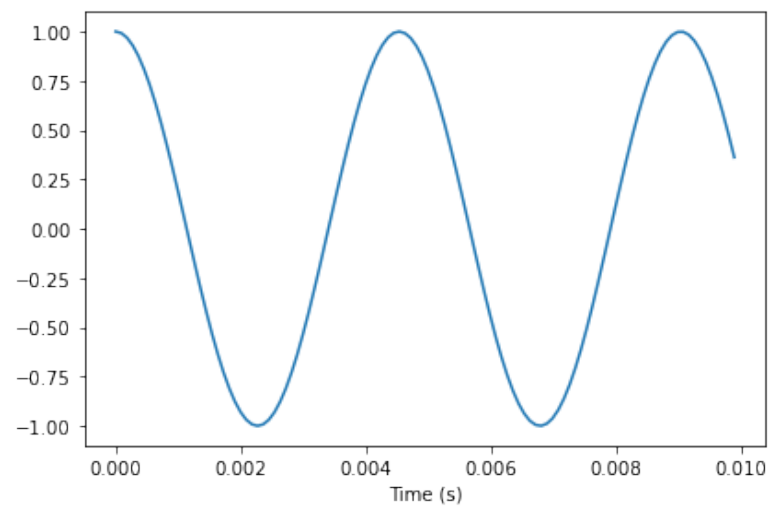


Рис. 1.1: Начало сигнала

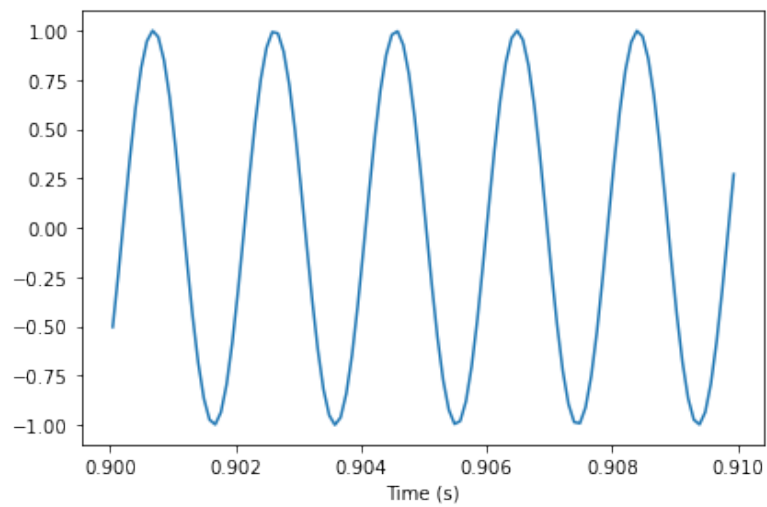


Рис. 1.2: Конец сигнала

```

1 signal = Chirp(start = 220, end = 440)
2 wave = signal.make_wave(duration = 1)
3 spectrum = wave.make_spectrum()
4 spectrum.plot(high = 700)
5

```

Листинг 1.2: Строим спектр чирпа

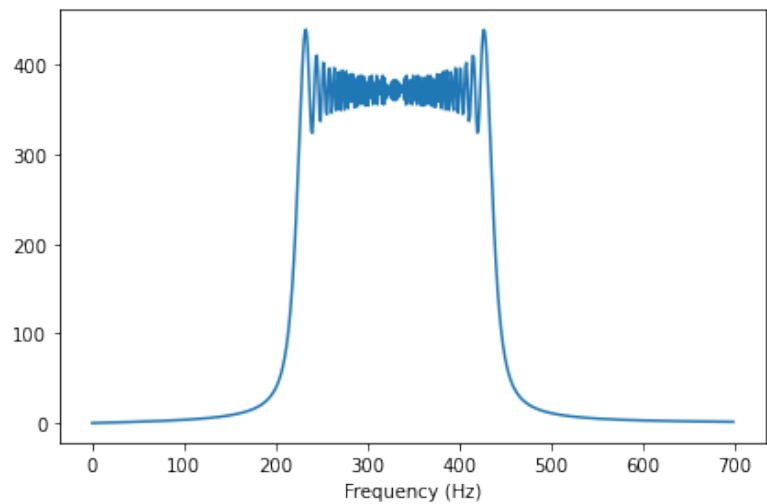


Рис. 1.3: Спектр чирпа

Слушая этот чирп, мы заметили, что высота звука сначала резко нарастает, а затем рост замедляется.

## Глава 2

# Спектрограмма

Знакомимся с одним из способов визуализации кратковременного преобразования Фурье (КВПФ).

```
1         def plot_spectrogram(wave, seg_length):
2             spectrogram = wave.make_spectrogram(seg_length)
3             print('Time resolution (s)', spectrogram.time_res
4         )
5             print('Frequency resolution (Hz)', spectrogram.
6         freq_res)
7             spectrogram.plot(high = 700)
8             decorate(xlabel = 'Time(s)', ylabel = 'Frequency
9         (Hz)')
10
11     signal = Chirp(start = 220, end = 440)
12     wave = signal.make_wave(duration=1, framerate=11025)
13     plot_spectrogram(wave, 512)
```

Листинг 2.1: Функция для построения спектрограммы

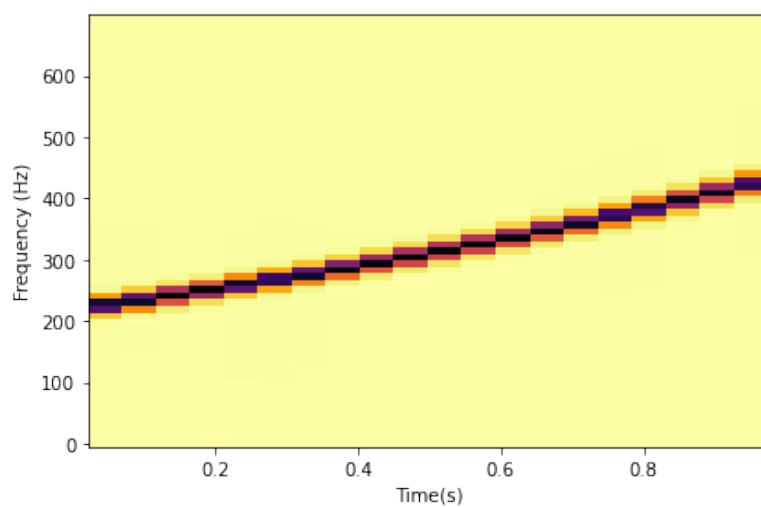


Рис. 2.1: Спектрограмма чирпа

Разрешение по времени спектрограммы зависит от длительности сегментов, соответствующих ширине ячеек спектрограммы. Разрешение по частоте спектрограммы - это частотный интервал между элементами спектра с одинаковой высотой ячеек.



## Глава 3

### Утечки

Знакомимся с утечками сигнала и способами их убрать.

```
1         from thinkdsp import SinSignal
2
3         signal = SinSignal(freq = 440)
4         duration = signal.period * 30.25
5         wave = signal.make_wave(duration)
6         wave.plot()
7         spectrum = wave.make_spectrum()
8         spectrum.plot(high = 880)
9
```

Листинг 3.1: Спектр сигнала

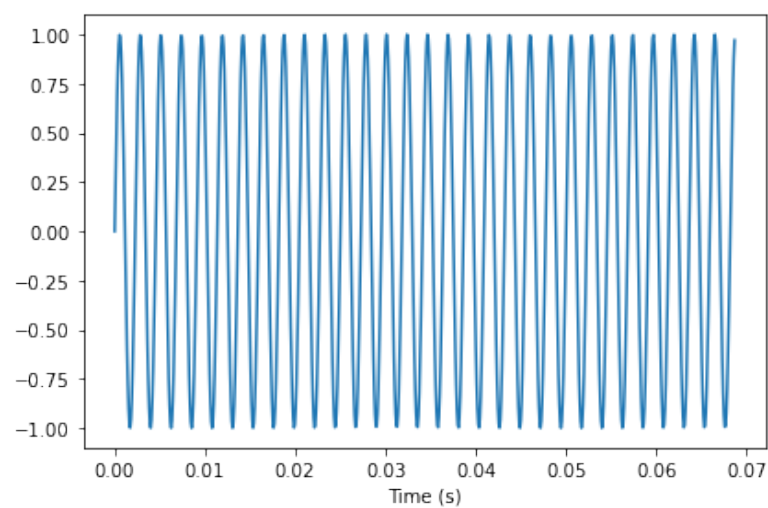


Рис. 3.1: Полученный сигнал

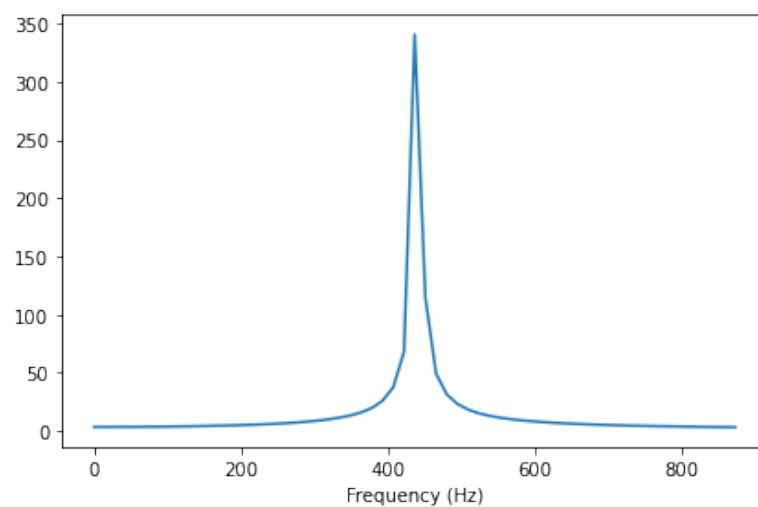


Рис. 3.2: Сегмент сигнала с утечкой

```

1 wave.hamming()
2 spectrum = wave.make_spectrum()
3 spectrum.plot(high = 880)
4

```

Листинг 3.2: Исправление сигнала функцией Хэмминга

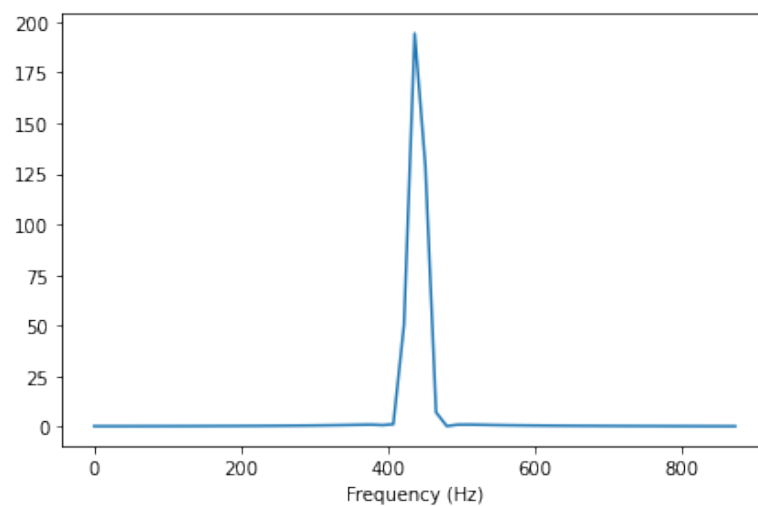


Рис. 3.3: Исправление утечки

# Глава 4

## Упражнения

### 4.1 Задание 1

Использование различных функций для исправления утечки сигнала.

```
1     for window_func in [np.bartlett, np.blackman, np.  
    hamming, np.hanning]:  
2         wave = signal.make_wave(duration)  
3         wave.ys *= window_func(len(wave.ys))  
4         spectrum = wave.make_spectrum()  
5         spectrum.plot(high = 880, label = window_func.  
    __name__)  
6
```

Листинг 4.1: Борьба с утечкой

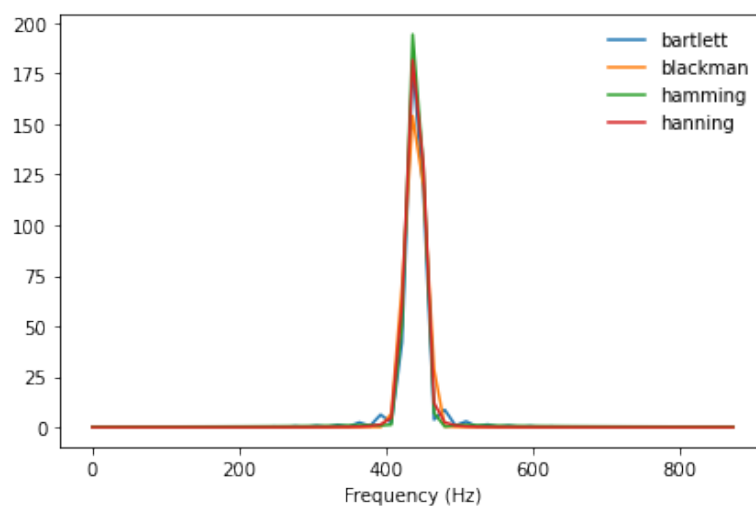


Рис. 4.1: Исправление утечки различными функциями

Все 4 функции хорошо справляются с уменьшением утечки, но фильтр Хэмминга рассеивает наименьшее количество энергии.

## 4.2 Задание 2

Пилообразный сигнал с линейно увеличивающейся (или уменьшающейся) частотой.

```
1         from thinkdsp import normalize, unbias
2
3         class SawtoothChirp(Chirp):
4
5             def evaluate(self, ts):
6                 freqs = np.linspace(self.start, self.end, len
(ts))
7                 dts = np.diff(ts, prepend = 0)
8                 dphis = self.PI2 * freqs * dts
9                 phases = np.cumsum(dphis)
10                cycles = phases / self.PI2
11                frac, _ = np.modf(cycles)
12                ys=normalize(unbias(frac), self.amp)
13                return ys
14
15        signal = SawtoothChirp(start = 220, end = 880)
16        wave = signal.make_wave(duration=1, framerate=8000)
17        wave.apodize()
18        sp = wave.make_spectrogram(256).plot()
19
```

Листинг 4.2: Создание нового класса

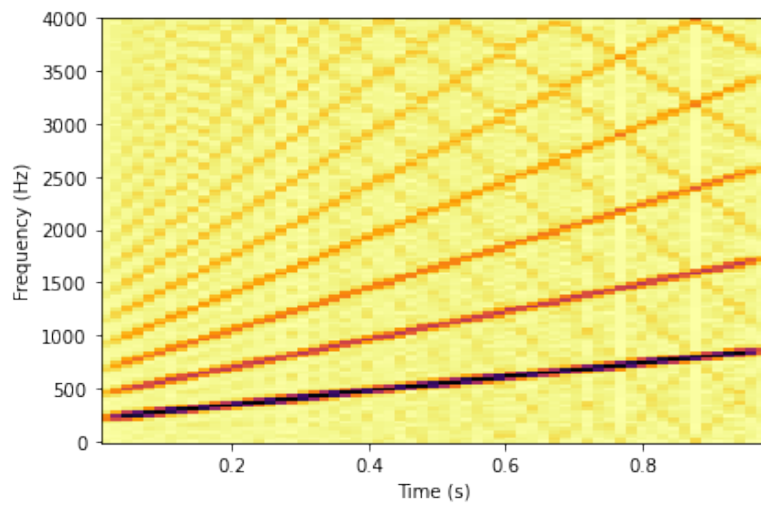


Рис. 4.2: Спектрограмма полученного сигнала

### 4.3 Задание 3

Создание пилообразного чирпа.

```
1      signal = SawtoothChirp(start = 2500, end = 3000)
2      wave = signal.make_wave(duration=1, framerate=20000)
3      wave.make_spectrum().plot()
4
```

Листинг 4.3: Пилообразный чирп

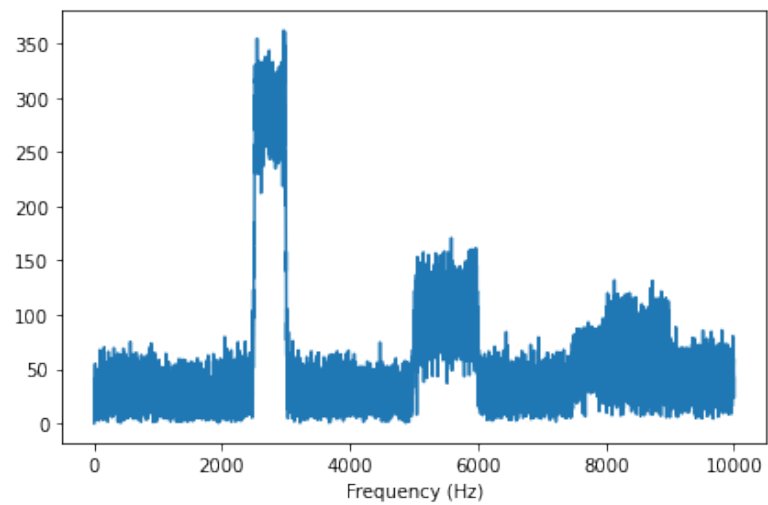


Рис. 4.3: Спектр пилообразного чирпа

## 4.4 Задание 4

Изучение сигнала глissандо

```
1 wave=read_wave('72475__rockwehrmann__glissup02.wav')
2 wave.make_audio()
3 wave.make_spectrogram(512).plot(high = 5000)
4
```

Листинг 4.4: Получение сигнала

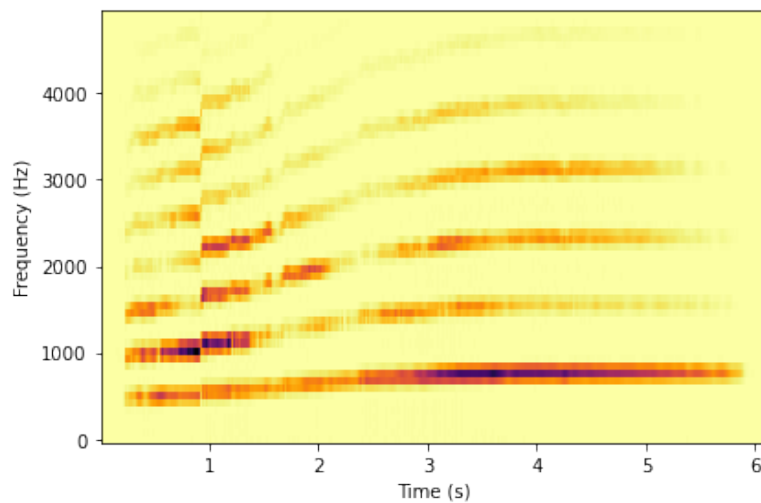


Рис. 4.4: Спектрограмма сигнала глиссандо

## 4.5 Задание 5

Изменение частоты сигнала глиссандо во времени.

```

1      class TromboneGliss(Chirp):
2
3          def evaluate(self, ts):
4              l1, l2 = 1.0 / self.start, 1.0 / self.end
5              lengths = np.linspace(l1, l2, len(ts))
6              freqs = 1 / lengths
7              dts = np.diff(ts, prepend = 0)
8              dphis = 2 * np.pi * freqs * dts
9              phases = np.cumsum(dphis)
10             ys = self.amp * np.cos(phases)
11             return ys
12

```

Листинг 4.5: Новый класс для изменения частоты сигнала

```

1      low = 262
2      high = 349
3      signal = TromboneGliss(high, low)
4      wave1 = signal.make_wave(duration = 1)
5      wave1.apodize()
6      signal = TromboneGliss(low, high)
7      wave2 = signal.make_wave(duration = 1)
8      wave2.apodize()
9      wave = wave1 | wave2
10     sp = wave.make_spectrogram(1024)

```

```
11     sp.plot(high = 1000)
12
```

Листинг 4.6: Получение требуемого сигнала

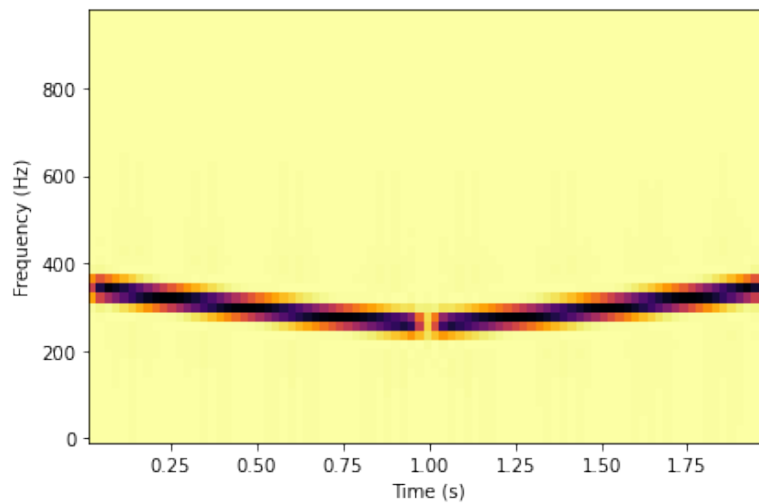


Рис. 4.5: Спектрограмма полученного сигнала

Полученная спектрограмма больше напоминает линейный чирп.

## 4.6 Задание 6

Спектрограмма гласных звуков.

```
1     wave = read_wave('87778__marcgascon7__vocals.wav')
2     wave.make_spectrogram(1024).plot(high = 1000)
3
```

Листинг 4.7: Получение сигнала



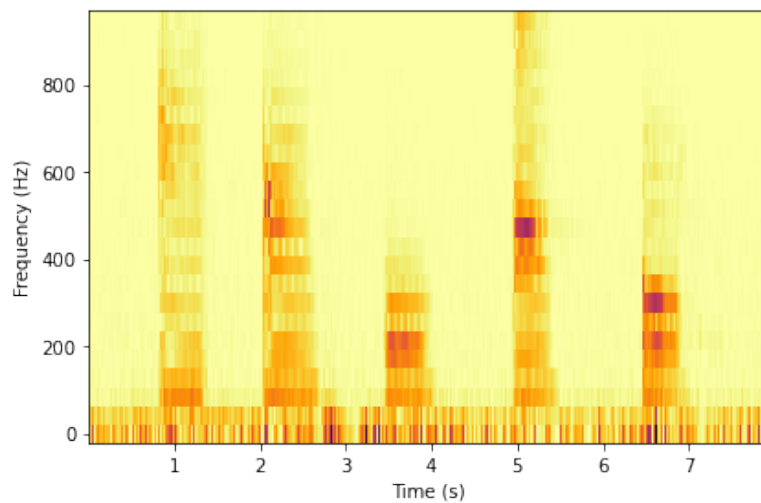


Рис. 4.6: Спектрограмма 5 гласных звуков

```

1 segment = wave.segment(start = 1, duration = 0.25)
2 wave.make_spectrogram(1024).plot(high = 1000)
3 segment = wave.segment(start = 2.2, duration = 0.25)
4 segment.make_spectrum().plot(high = 1000)
5 segment = wave.segment(start = 3.5, duration = 0.25)
6 segment.make_spectrum().plot(high = 1000)
7 segment = wave.segment(start = 5.1, duration = 0.25)
8 segment.make_spectrum().plot(high = 1000)
9 segment = wave.segment(start = 6.5, duration = 0.25)
10 segment.make_spectrum().plot(high = 1000)
11

```

Листинг 4.8: Сегменты для каждого звука

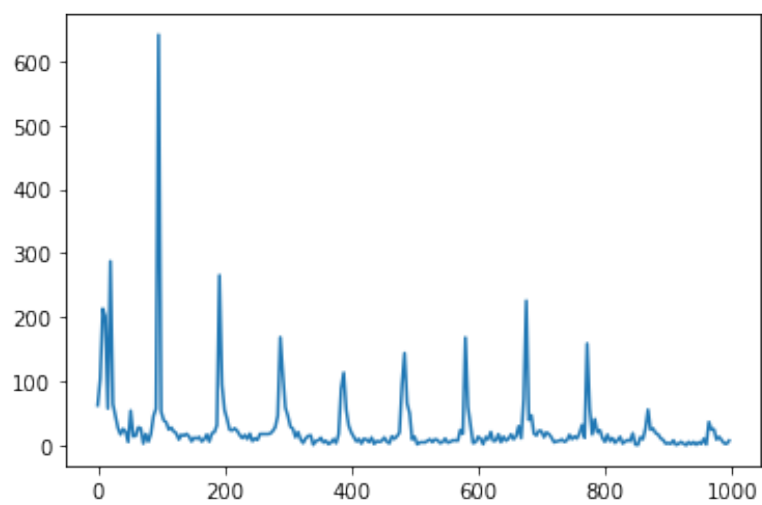


Рис. 4.7: Сегмент со звуком А

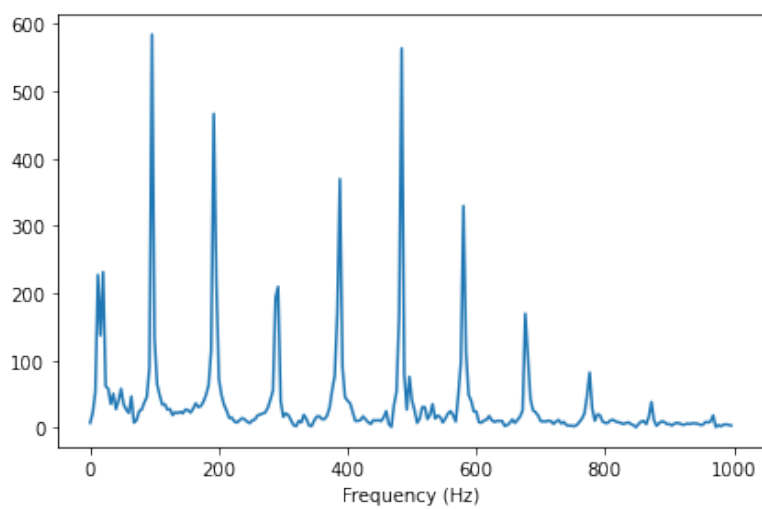


Рис. 4.8: Сегмент со звуком Э

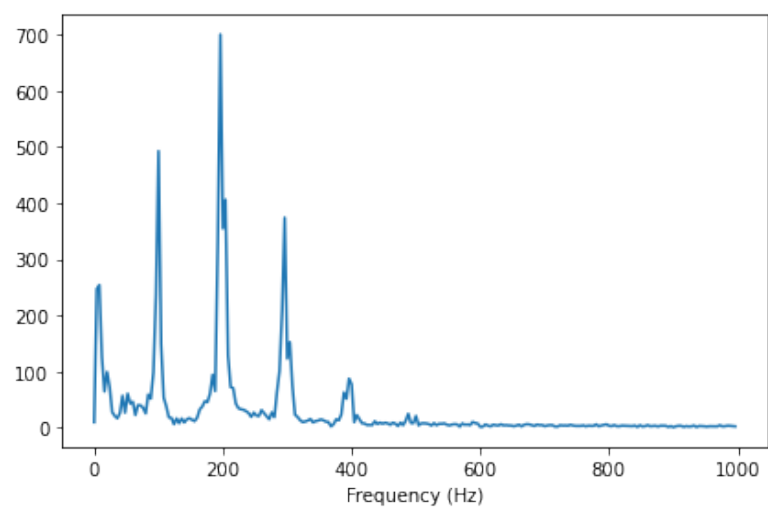


Рис. 4.9: Сегмент со звуком И

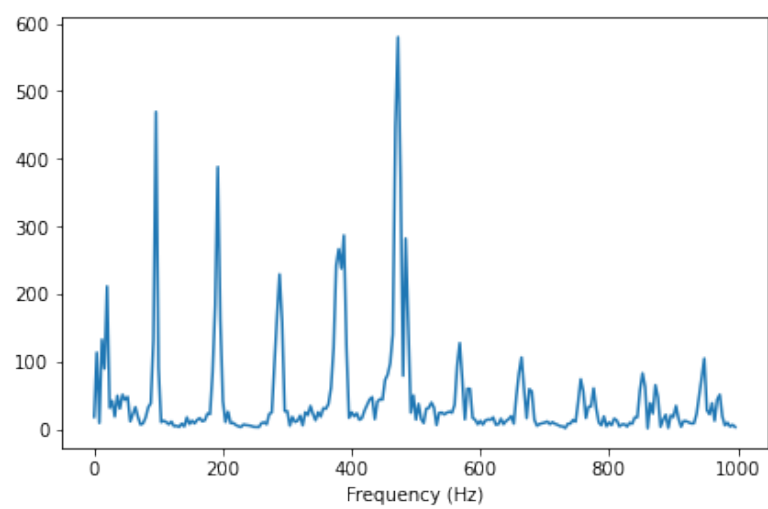


Рис. 4.10: Сегмент со звуком О

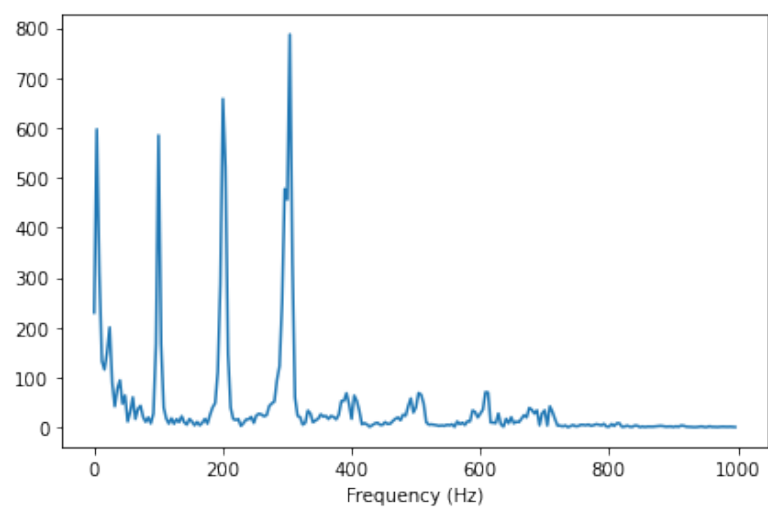


Рис. 4.11: Сегмент со звуком У

По спектрограммам и сегментам очень сложно определить гласную.

## Глава 5

### Вывод

В данной работе мы познакомились с сигналом **чирпа**, спектрограммами и утечками сигналов. Утечки сигналов можно уменьшить, если использовать различные виды **окон**. В конце увидели спектрограмму 5 разных гласных звуков.