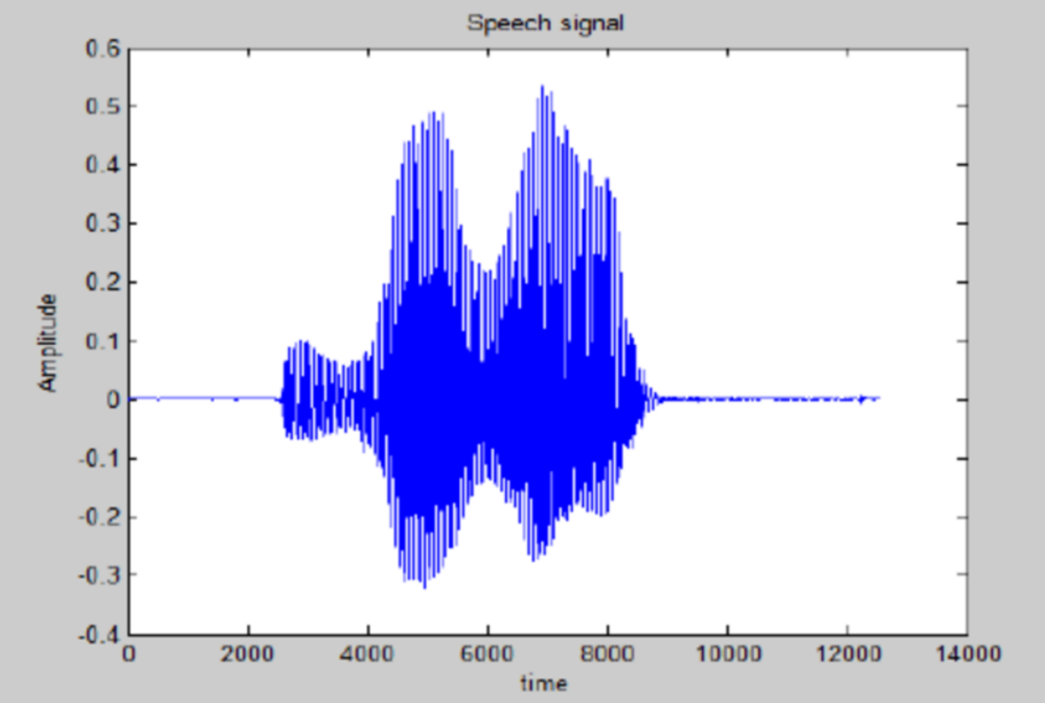
Speech Signals

Speech signals are sound signals, defined as pressure variations travelling through the air. These variations in pressure can be described as waves and correspondingly they are often called sound waves. In the context of speech processing and analysis, we assume the sound signals are captured by a microphone and converted to a digital signal.

A speech signal is then represented by a sequence of numbers x_n , which represent the relative air pressure at time-instant $n \in \mathbb{N}$. This representation is known as pulse code modulation often abbreviated as PCM. The accuracy of this representation is then specified by two factors; 1) the sampling frequency (the step in time between n and $n+1$) and 2) the accuracy and distribution of amplitudes of x_n.[1]

.



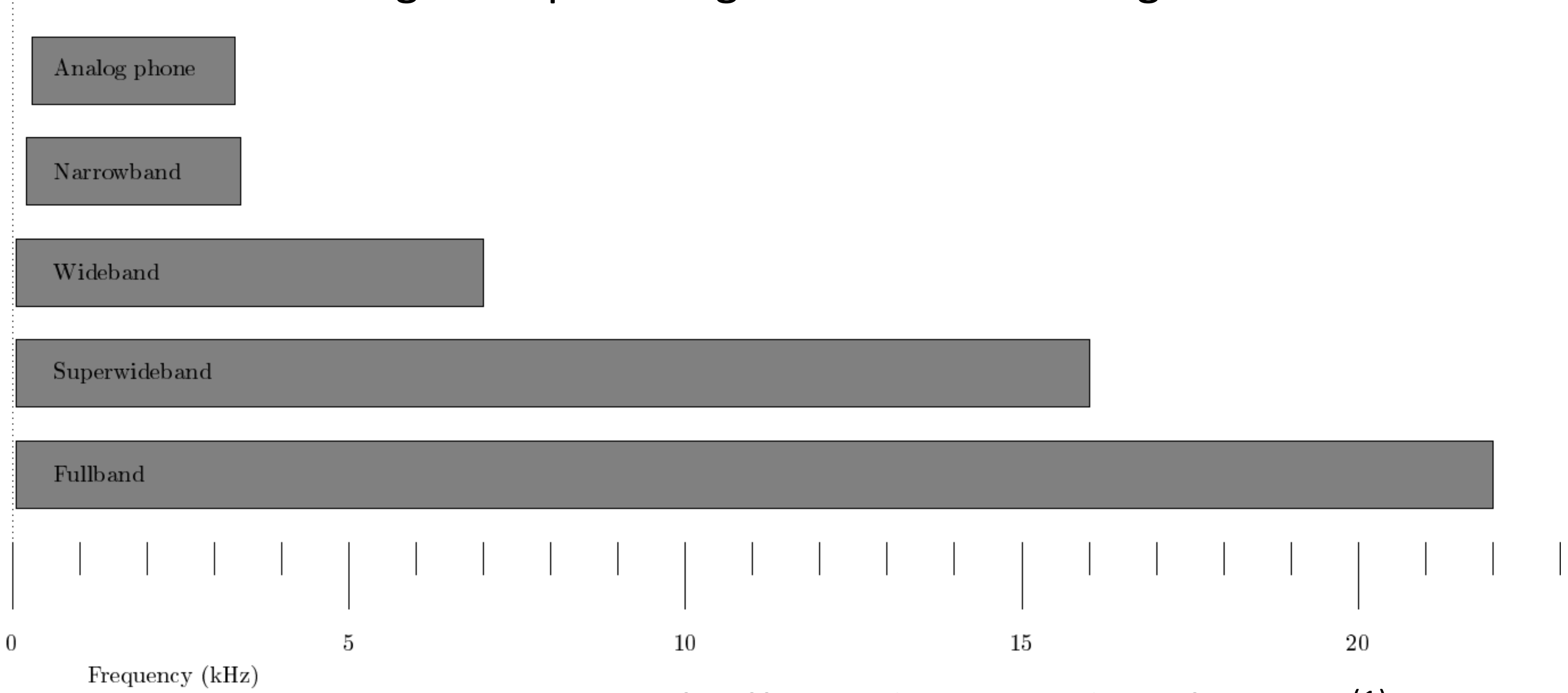
Example of speech signal[2]

Sampling

Sampling frequency for a speech signal is the time step between n^{th} and n+1$^{\text{th}}$ index point in our signal.

Nyquist frequency is half the sampling frequency and defines the upper end of the largest bandwidth which can be uniquely represented by that signal. Hence, for example if the sampling frequency is 8000 Hz, then signals in the frequency range 0 to 4000 Hz can be uniquely described with this sampling frequency.

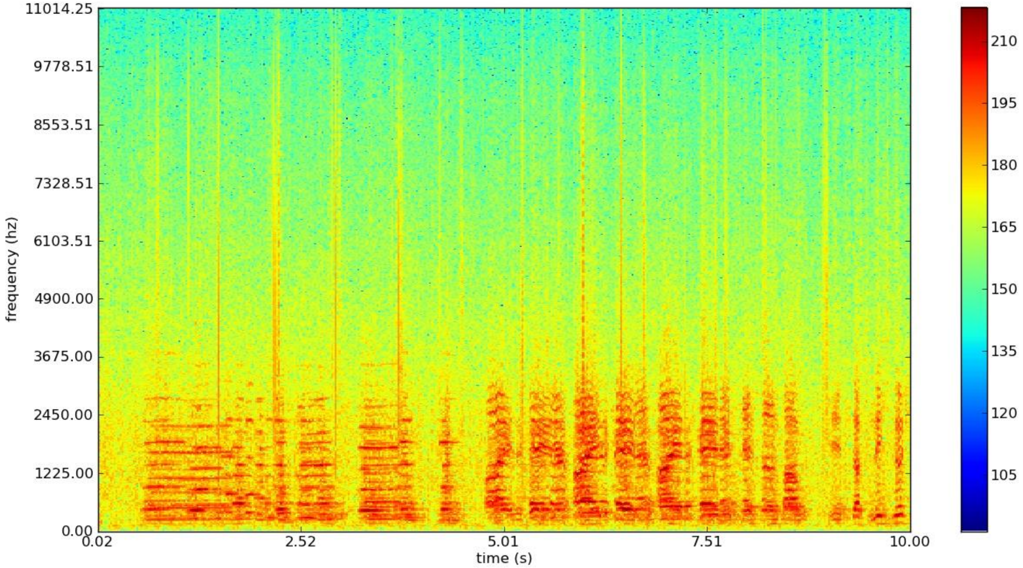
A sampling rate of 8 kHz is known as narrow-band. Most energy remains below 8 kHz hence a sampling rate of 16 kHz suffices for most tasks. Super-wide band and full band correspond to 32 kHz and 44.1 kHz. These higher rates are especially useful when considering non-speech signals like music and generic audio.[1]



Frequency-range of different bandwidth-definitions[1]

Spectrograms

- A spectrogram is a 2-d visual representation of the spectrum of frequencies in a signal as they vary with time.
- The frequency content of a signal is displayed on the vertical axis, time on the horizontal axis, and the intensity or amplitude of each frequency component represented by colors or shades.
- They are valuable tools for studying and understanding complex signals, particularly in fields like speech processing, music analysis, and audio signal processing.



Example of a spectrogram(1)

Mathematics behind spectrograms

Fourier Transform:

- The Fourier Transform decomposes a signal into its constituent frequencies. It transforms a signal from the time domain to the frequency domain.
- The continuous Fourier Transform of a signal x(t) is given by $X(f) = \int_{-\infty}^{\infty} x(t) e^{-i2\pi ft} dt$.

Short Time Fourier Transform (STFT):

- STFT is a time-dependent extension of the Fourier Transform. It represents how the frequency content of a signal changes over short, overlapping time intervals.
- To compute the STFT, the function to be transformed is multiplied by a window function which is nonzero for only a short period of time. The Fourier Transform of the resulting signal is taken, then the window is slid along the time axis until the end resulting in a two-dimensional representation of the signal.(2)
- The STFT of a signal x(t) is given by $X(\tau, f) = \int_{-\infty}^{\infty} x(t)\omega(t-\tau) e^{-i2\pi ft} dt$ where $\omega(t)$ is the window function.
- Spectrogram is the square of magnitude of STFT of the signal.

Window Function:

- A window function is a mathematical function that is zero-valued outside some chosen interval.
- When a signal is multiplied by a window function, the product is also zero-valued outside this interval. Effectively, we are viewing the signal through a “window,” hence the name of the function.(3)
- Typically, windows functions are symmetric around the middle of the interval, approach a maximum in the middle, and taper away from the middle.
- Examples: Rectangular Window ($\omega(n) = 1$ for $0 \leq n \leq N$ and 0 otherwise),

  Hann Window ($\omega(n) = 0.5 - 0.5\cos(\frac{2\pi n}{N-1})$), Gaussian Window ($\omega(n) = e^{-\left(\frac{n-\frac{N-1}{2}}{\frac{\sigma(N-1)}{2}}\right)^2}$)
- The STFT is subject to a trade-off between time and frequency resolution due to the choice of window length and shape.
- With a shorter window, we get a STFT with higher time resolution helping us to analyze rapidly evolving sounds such as speech or percussive instruments, however we have a lower frequency resolution making it difficult to distinguish between closely spaced frequency components.(4)
- A longer window provides higher frequency resolution enabling us to identify small frequency differences which is particularly useful in analyzing steady-state sounds like sustained musical notes or constant background noise, however there is a downside of a lower time resolution.(4)

Creating spectrograms using Python

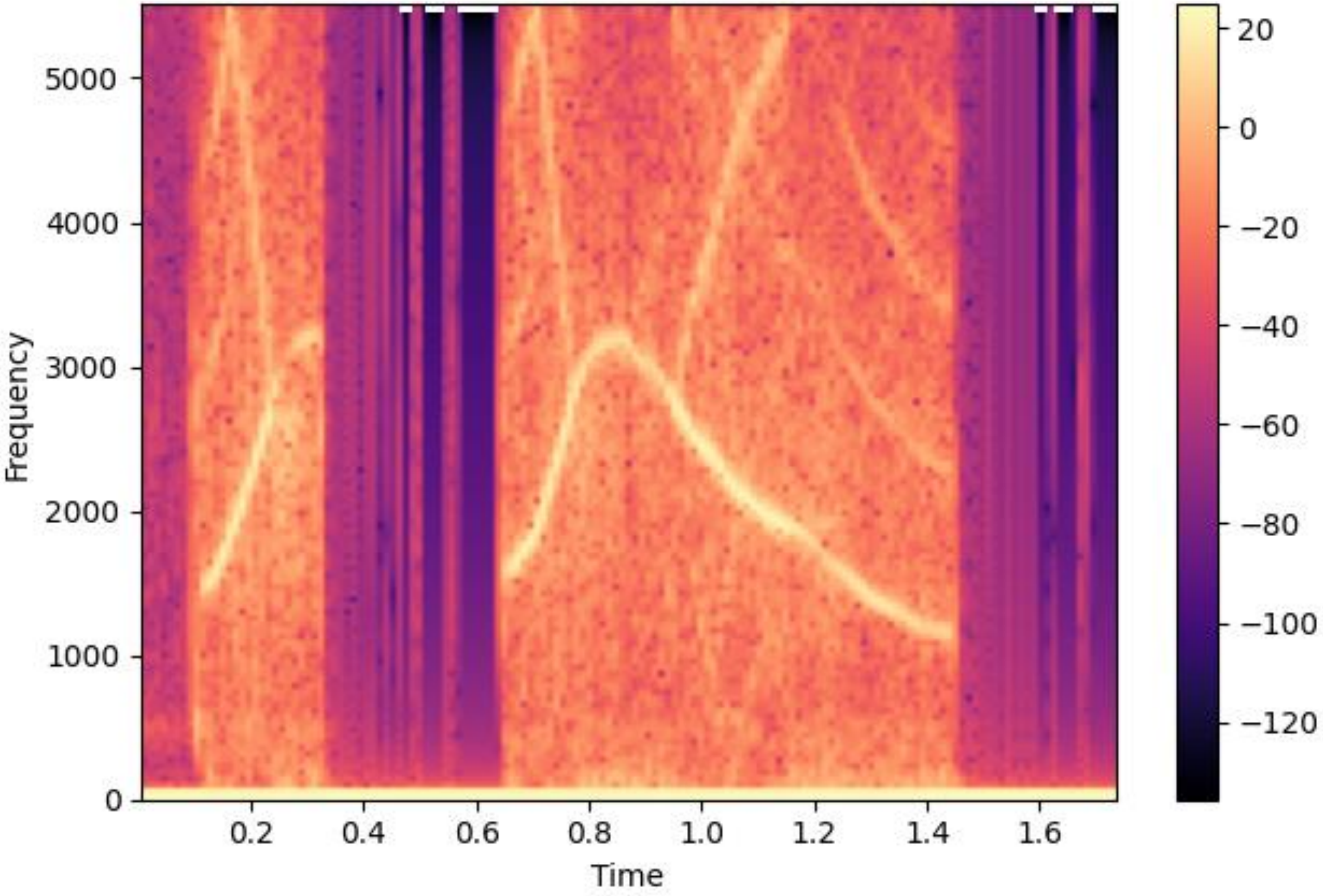
Code for creating a basic spectrogram in python is given below:

```
#Necessary imports
import  numpy  as  np
import matplotlib.pyplot as plt
import scipy as sp

#converting .wav audio file to numpy array
fs, aud = sp.io.wavfile.read('whistle_1.wav’)

#computing the stft of our audio clip as well as plotting the spectrogrm
stft, freq, time, spec = plt.specgram(aud,Fs = fs)
plt.xlabel('Time')
plt.ylabel('Frequency’)
plt.colorbar(spec)
plt.show()
```

Output:

For the following audio clip:

When we run our code, we get the following output:


Frequency
Time