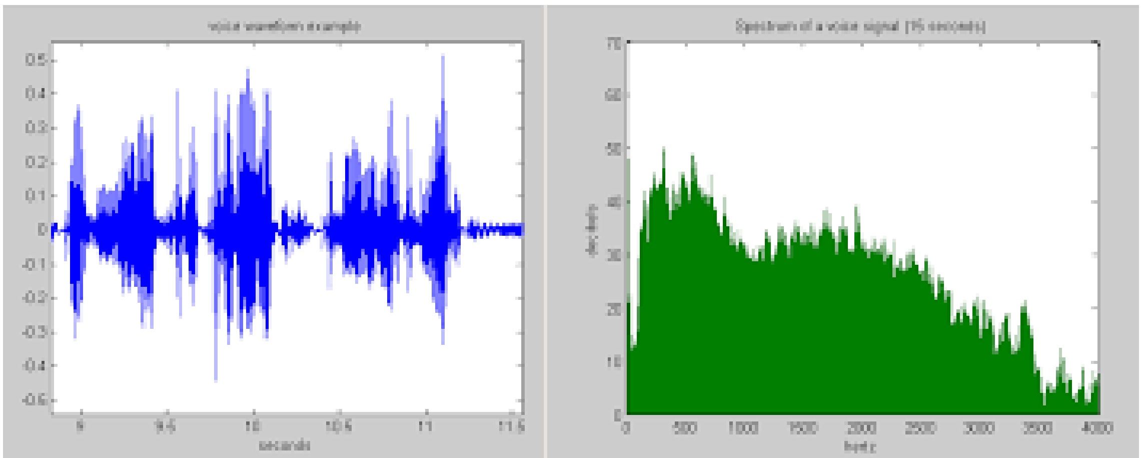
Signal and Systems

Lab 2: Signal analysis with DFT



Data Science and Engineering, 2019-2020

Contents

1 Introduction

This lab focuses on the signal analysis with DFT and its application to speech signals.

During the lab session you will have to write a script in Python to answer the questions included in this document. Start your code as follows:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.fftpack import fft, ifft
from scipy.signal import hamming, convolve
from scipy.io import wavfile
import pyaudio

def sound(array, sf=8000):
    iarray = (array*2**15).astype('int16')
    p = pyaudio.PyAudio()
    stream = p.open(format=pyaudio.paInt16, channels=1, rate=sf, output=True)
    stream.write(iarray.tobytes())
    stream.stop_stream()
    stream.close()
    p.terminate()

def record(duration=3, sf=8000):
    nsamples = duration*sf
    p = pyaudio.PyAudio()
    stream = p.open(format=pyaudio.paInt16, channels=1, rate=sf, input=True,
                    frames_per_buffer=nsamples)
    buffer = stream.read(nsamples)
    array = np.frombuffer(buffer, dtype='int16')
    stream.stop_stream()
    stream.close()
    p.terminate()
    return array/float(2**15)
```

If you are using Spyder (recommended), an IDE currently shipped with the Anaconda installer (http://anaconda.com/downloads), separating sections with #%% will allow you to execute each section independently. To know how to use the function `fft`, type `fft?` in the IPython console.

When using Matplotlib to plot graphs, instead of the default inline drawing, you may want to have the figures in a separate window so that interactions can be enabled. To display figures in a separate window, in Spyder click Tools, Preferences, Ipython Console, Graphics and under Graphics Backend select "automatic" instead of "inline".

For interactive plots with Jupyter Notebooks execute the command `%matplotlib notebook`.

IMPORTANT NOTE: Sections 2.1 and 2.2 contain a summary of the theory already seen in class that you need to do the first part of the lab. Before starting the second part of the lab, you must have carefully read section 2.3.

2 Related theory

2.1 The Discrete Fourier Transform (DFT)

The discrete Fourier transform (DFT) of a sequence $x[n]$ is defined as

$$X_N[k] = DFT_N\{x[n]\} = \sum_{n=0}^{N-1} x[n]e^{-j2\pi\frac{k}{N}n}, \qquad 0 \leq k \leq N-1 \tag{1}$$

which generates N frequency samples.

As it can be seen, the DFT *observes* only samples of $x[n]$ in the interval $n \in [0, N-1]$. That is, the signal is implicitly windowed with a rectangular window of length N. In addition to this implicit windowing, we can apply an additional window $w[n]$ of length $L \leq N$ (e.g., rectangular, Hamming, Hanning, etc.), obtaining the windowed signal:

$$x_w[n] = x[n] \cdot w[n] \tag{2}$$

As the length of $x_w[n]$ is smaller than N, the **DFT of $x_w[n]$ is equal to the sampling of the Fourier Transform of $x_w[n]$**, that is:

$$X_{w,N}[k] = DFT_N\{x_w[n]\} = X_w(F)|_{F=\frac{k}{N}} = X(F) \otimes W(F)|_{F=\frac{k}{N}}, \qquad 0 \leq k \leq N-1 \tag{3}$$

where $\otimes$ denotes the periodic convolution and $X_w(F)$, $X(F)$ and $W(F)$ are the Fourier transform of $x_w[n]$, $x[n]$ and $w[n]$, respectively. In particular, if $w[n]$ is a rectangular window of length L, we have that

$$W(F) = e^{-j\pi F(L-1)}\frac{\sin(\pi FL)}{\sin(\pi F)} \tag{4}$$

The inverse DFT is given by

$$x_w[n] = IDFT_N\{X_{w,N}[k]\} = \frac{1}{N}\sum_{k=0}^{N-1} X_{w,N}[k]e^{j2\pi\frac{k}{N}n}, \qquad 0 \leq n \leq N-1 \tag{5}$$

which generates N temporal samples. Note that the last $N-L$ samples of $x_w[n]$ (i.e., $L \leq n \leq N-1$) are equal to zero due to the window $w[n]$.

2.2 Frequency analysis of periodic sequences

A sequence $x[n]$ is periodic if there exists an integer P such that $x[n+P] = x[n]$ for any value of n. The lowest value of P holding the above identity is the **period** of $x[n]$ and its inverse $F_0 = \frac{1}{P}$ the (discrete) **fundamental frequency** of $x[n]$ (or, alternatively, $f_0 = F_0 \cdot f_s$ its analog counterpart).

A periodic sequence $x[n]$ can be expressed as the convolution ($*$) of a basic sequence $x_b[n]$ with an infinite train of deltas:

$$x[n] = \sum_{i=-\infty}^{\infty} x_b[n - iP] = x_b[n] * \sum_{i=-\infty}^{\infty} \delta[n - iP] \tag{6}$$

The Fourier Transform (FT) of a periodic train of discrete deltas in the time domain is a periodic train of Dirac deltas in the frequency domain:

$$\sum_{i=-\infty}^{\infty} \delta[n - iP] \overset{FT}{\leftrightarrow} \frac{1}{P} \sum_{r=-\infty}^{\infty} \sum_{m=0}^{P-1} \delta\left(F - \frac{m}{P} - r\right) = \frac{1}{P} \sum_{l=-\infty}^{\infty} \delta\left(F - \frac{l}{P}\right) \tag{7}$$

Note that in Eq. 7 we make explicit that, for a fixed r, we have one period of the Fourier Transform, and the sum over r expands the periodicity (as any Discrete Time Fourier Transform).

Using Eq. 7, the Fourier Transform of $x[n]$ is given by

$$X(F) = X_b(F) \frac{1}{P} \sum_{r=-\infty}^{\infty} \sum_{m=0}^{P-1} \delta\left(F - \frac{m}{P} - r\right) = \sum_{r=-\infty}^{\infty} \sum_{m=0}^{P-1} \frac{1}{P} X_b\left(\frac{m}{P}\right) \delta\left(F - \frac{m}{P} - r\right) \tag{8}$$

That is, the Fourier Transform of a periodic signal is composed of Dirac deltas at harmonic frequencies of $F_0 = \frac{1}{P}$.

Given a window $w[n]$ of length L samples, the windowed periodic signal $x_w[n] = x[n] \cdot w[n]$ has the following Fourier transform:

$$X_w(F) = X_b(F) \otimes W(F) = W(F) * \sum_{m=0}^{P-1} \frac{1}{P} X_b\left(\frac{m}{P}\right) \delta\left(F - \frac{m}{P}\right) = \sum_{m=0}^{P-1} \frac{1}{P} X_b\left(\frac{m}{P}\right) W\left(F - \frac{m}{P}\right) \tag{9}$$

The Fourier Transform of a windowed periodic signal is composed of a set of replicas of $W(F)$ at harmonic frequencies of $F_0 = \frac{1}{P}$. The complex amplitude of each replica is proportional to the value of the Fourier Transform of the basic signal $x_b[n]$ at the corresponding harmonic.

The N points DFT of the windowed periodic signal, with $N \geq L$, is obtained by sampling (Eq. 9) as follows:

$$X_{w,N}[k] = X_w(F)|_{F=\frac{k}{N}} = \sum_{m=0}^{P-1} \frac{1}{P} X_b\left(\frac{m}{P}\right) W\left(\frac{k}{N} - \frac{m}{P}\right) \tag{10}$$

2.3 The speech signal

Fig. 1 shows an example of a speech signal that has been sampled at 8 kHz. Two types of segments can be clearly identified. The first one, known as **voiced sounds**, corresponds to high energy and almost periodic segments of the signal whereas the second type corresponds to low energy segments that are called **voiceless sounds**.

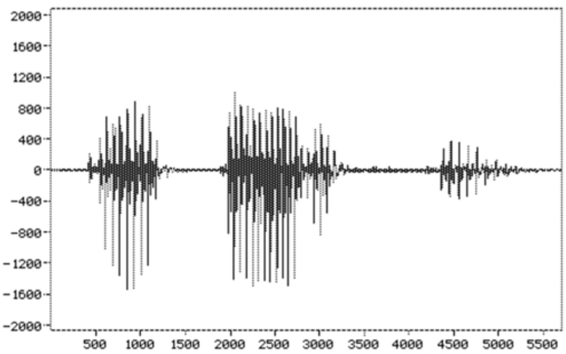


Figure 1: Example of speech signal sampled at 8 kHz.

The speech signal is produced by expelling air from the lungs through the vocal tract. Voiced sounds are **periodic** sounds produced by the vibration of the vocal folds. Their fundamental frequency is determined by the frequency of the vocal folds vibration. It can be modified and forms the basis of the prosodic intonation and the singing ability. In average, the fundamental frequency (**pitch**) of voiced sounds is around 130 Hz for men and around 220 Hz for women.

The speech production system can be modeled in terms of signals and systems as described in Fig. 2, where the vocal tract is modeled as a linear invariant filter with impulse response $h[n]$. The filter input signal can be either a periodic train of impulses $t[n]$ in the case of voiced sounds or a random noise $r[n]$ in the case of voiceless sounds.

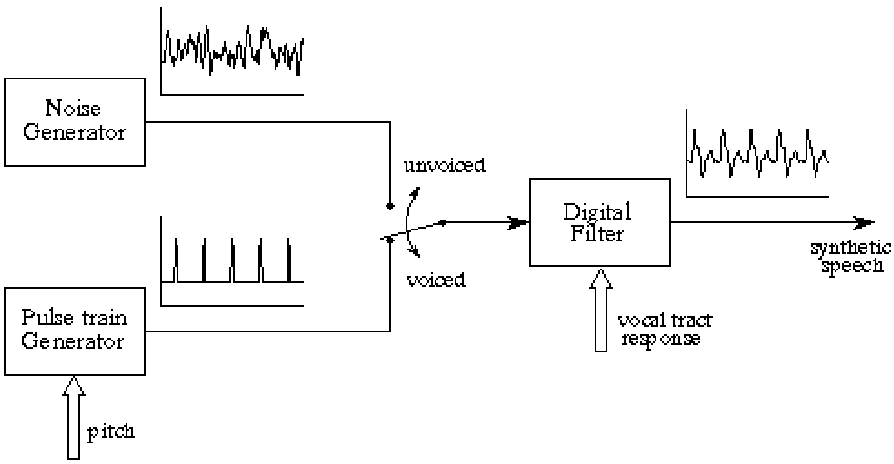


Figure 2: Model of the human speech production system.

The function of the filter is to amplify or to attenuate some frequencies (depending on the resonances of the cavities). The amplified frequencies (resonances) are known as "**formants**" and are characteristic of specific sounds. They are essentially defined by the shape of the pharyngeal and oral resonance cavities. See Fig. 3 for a detailed view of the human speech production system.

Vowels are voiced sounds. The periodic signal produced by the vibration of the vocal cords propagates through the vocal tract that amplifies or attenuates the harmonic frequencies of the fun-

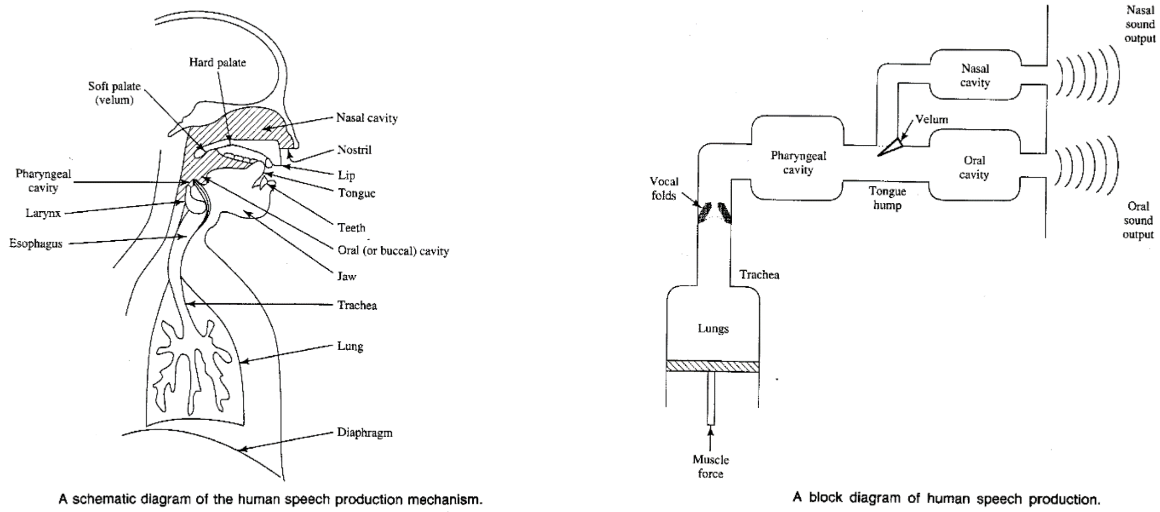


Figure 3: Illustration of the human speech production system.

damental frequency. Vowels can be characterized by two formants. These formants depend on the specific configuration of the vocal tract (articulation) that produces the vowel.

The articulation can be described in terms of the following three factors:

- The **height**, which is the vertical position of the tongue relative to either the roof of the mouth or the aperture of the jaw. The height controls the size of the constriction generated by the tongue hump.
- The **backness**, which is the position of the tongue during the articulation of a vowel relative to the back of the mouth. The backness controls the size of the resonant chambers.
- The **roundness** of the lips.

Fig. 4 shows the articulation of the 6 Spanish vowels and the associated height and backness.

Figure 4: Pictures showing the articulation of the five Spanish vowels

The Fourier transform of a vowel shows two resonance frequencies (formants), which will be denoted by f_1 and f_2. Surprisingly, the two formants characterizing a vowel fall into a triangle as shown in the vowel diagram of Fig. 5.

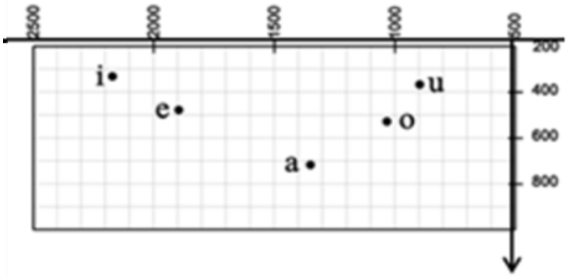


Figure 5: Vowel diagram providing two first formants for the 5 Spanish vowels.

Fig. 6 shows an example of a vowel segment sampled at 8 kHz that has been windowed (upper plot), and the magnitude (in dBs) of its Fourier transform (lower plot).

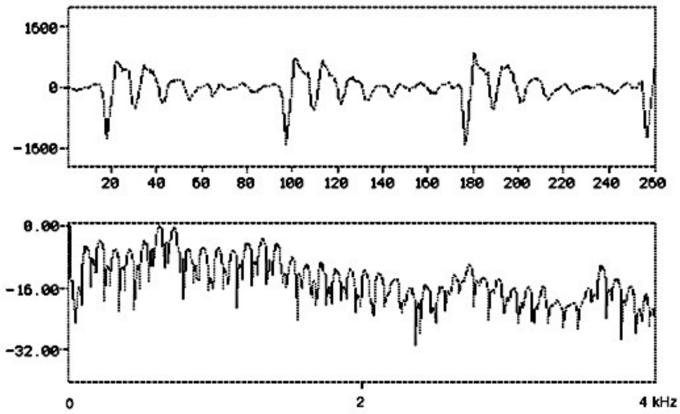


Figure 6: Example of a windowed vowel and its DFT (lower plot)

3 Lab work part 1: DFT and windows

1. Introduce the name of the two members of the group in the Atenea quiz. **Open just one Atenea quiz.**

Windowing of the DFT

Theoretical questions

Let us first characterize a sinusoid, $x[n] = \cos(2\pi F_0 n)$, in the frequency domain. Consider $F_0 = 0.25$ and that we have only $L = 8$ samples of the previous sinusoid. i.e. $y[n] = x[0], ..., x[7]$.

2. Represent (in a paper, without Python) $X(F) = FT\{x[n]\}$, for F between 0 and 1.

3. Represent approximately $|Y(F)|$ in the interval $[0, 1)$ indicating the position of the zeros. From the plot of $|Y(F)|$, obtain the DFT of $N = 16$ samples.

Using Python, generate $L = 8$ samples of the sinusoid $x[n] = \cos(2\pi F_0 n)$ with $F_0 = \frac{m}{L}$, being m an integer. Plot the magnitude of the DFT of size L and size $N = 16$.

```
Fo = #select the frequency of the sinusoid
L =  #select the value of L
N =  #select the size of the DFT
n = np.arange(L)
x = np.cos(2*np.pi*Fo*n)
X1 = fft(x)
X2 = fft(x, N)
plt.figure(1)
plt.stem(np.abs(X1)) #Plot of |X1[k]|
plt.figure(2)
plt.stem(np.abs(X2)) #Plot of |X2[k]|
```

Repeat for $F_0 \neq \frac{m}{L}$.

For both, $F_0 = \frac{m}{L}$ and $F_0 \neq \frac{m}{L}$, try to estimate the frequency and amplitude of the sinusoid from the magnitude of the DFT

4. How does the estimation of the frequency of the sinusoid from the DFT change with L and/or N?
5. How does the estimation of the amplitude of the sinusoid from the DFT change with L and/or N?

Fourier transform of windows

Often it is necessary to calculate the features of a signal from a short segment. To improve the detection of some features, the segment is usually modified by multiplying by a function (window). In this section, we are going to study the effect (in both temporal and frequency domain) of two common window functions: a rectangular window $w_r[n]$ and a Hamming window $w_h[n]$.

For any window, the width of the main lobe of its Fourier transform determines the frequency **resolution** of the window, i.e. its ability to detect spectral components of similar amplitude which are close in frequency. The ratio between the magnitude of the main lobe and the magnitude of the first secondary lobe determines the **sensitivity** of the window, i.e. its ability to detect spectral components with very different amplitudes.

Plot the Fourier transform of a rectangular window and a Hamming window both of the same length L. If you use `plt.plot` instead of `plt.stem` to represent the DFT, matplotlib will linearly interpolate between samples.

```
plt.close('all')
L = #select the value of L
N = #select the size of the DFT: N>>L
wr = np.ones(N) #a rectangular window
wh = hamming(L) #a Hamming window
Wr = fft(wr, N)
Wh = fft(wh, N)
F = np.linspace(0, 1, N)
plt.figure(1)
plt.plot(abs(Wr)) #or plt.plot(F, abs(Wr))
```

```
plt.figure(2)
plt.plot(abs(Wh))  #or plt.plot(F, abs(Wh))
```

6. How does L impact on the frequency transform of the window?

7. Which window has better frequency resolution?

8. Which window has better sensitivity?

Window effect: detecting frequency components in unknown mixtures

Download the folder 'Señales'. The file 'mixture.npy' contains 75 signals, each one with 32 samples of an unknown mixture of two sinusoids: $A_1 \cos(2\pi F_1 n) + A_2 \cos(2\pi F_2 n)$, with $F_1 < F_2$.

We will detect the frequency components in them. To that end, we will apply a rectangular or a Hamming window (as one window may be better than the other, depending on the signal).

```
plt.close('all')
id = XXX #You need to modify this line: id is the identifier of the selected signal
s = np.load('mixture.npy')
x1 = s[id] #applying a rectangular window, i.e. no windowing
x2 = s[id]*hamming(32) #applying a Hamming window
...
```

Take a signal between 0 and 24. Apply first a rectangular window to the signal and represent the absolute value of its DFT (choose a DFT size[1] $N >> 32$). Repeat with a Hamming window.

9. Estimate F_1

10. Estimate F_2

11. Estimate the amplitudes of the sinusoids

Take now a signal between 25 and 49. Apply first a rectangular window to the signal and represent the absolute value of its DFT. Repeat with a Hamming window.

12. Estimate F_1

13. Estimate F_2

14. Which window seems to be more useful for the estimation?

Take now a signal between 50 and 74.

15. Estimate F_1

16. Estimate F_2

17. Which window seems to be more useful for the estimation?

[1] As we use the FFT algorithm to compute the DFT, select N such that N is a power of 2.

4 Lab work part 2: Speech signals

1. Introduce the name of the two members of the group in the Atenea quiz. **Open just one Atenea quiz**

Theoretical questions

The objective of this section is to study the Fourier transform of vowels and see the relationship of the module with the production of the signal. In view of the figure 6 of this document, answer the following theoretical questions (no Python):

2. Can you see the windowing effect on the DFT of the signal? What is the fundamental frequency of the vocal folds? (Explain how you can estimate it from both the time-domain signal and its DFT in Figure 6)

3. Locate the two most important formants of this vowel figure 6

Study of the structure of speech by means of synthetic signals

Generate 1000 samples of the signal $e[n]$, a train of impulses separated by P samples (you may use instructions `e=np.zeros(1000), e[::P]=1`). Let be $h[n] = A_1^n \cos(2\pi F_1 n) + A_2^n \cos(2\pi F_2 n)$ a filter modelling the vocal tract and $y[n] = e[n] * h[n]$ (`y=convolve(e,h)`).

Plot $h[n]$, $y[n]$ and the DFT of both $h[n]$ and $y[n]$. Consider for instance[2] $A_1 = 0.95$, $A_2 = 0$, $F_1 = 0.1$, $F_2 = 0.3$, and $P = 60$.

4. What can you say about $y[n]$?

5. Can you observe spectral lines in $H[k]$ and/or $Y[k]$?

6. What are the differences between $H[k]$ and $Y[k]$?

7. Select only one period of $y[n]$ and plot the magnitude of its DFT. Compare it with $H[k]$ and $Y[k]$

Frequency analysis of vowels

Load one of the wavefiles 'unknownX' and inspect its characteristics. Plot the segment that contains the vowel.

```
sf, data = wavfile.read('unknownX.wav') #YOU MAY NEED TO CHANGE THE FILE IN CLASS
data = data/2**15
plt.plot(data)
idx1 = XXX #Modify this line
idx2 = XXX #Modify this line
x = data[idx1:idx2] #Segment that contains the vowel.
...
```

[2] You may be asked to use a different set of values in class.

8. As surely you have observed, the segment that contains the vowel is practically a periodic signal. What is the length in samples of one period of the vowel?

Now, compute the DFT of the segment that contains the vowel **(apply a Hamming window** before computing the DFT). Compute also the DFT of only one period of the vowel.

The absolute value of the FT of the signal may not capture features of the frequency representation of the signal that are of low amplitude. It is known that the subjective perception of the loudness of sound is a logarithmic function of the intensity of the sound. Therefore a natural representation of the DFT is a representation that reflects the perception of the sound, i.e. a representation in a logarithmic scale: $20\log_{10}(|X[k]|)$ (dB). **Use this representation when you plot the magnitude of the FFTs**, i.e. use `20*np.log10(abs(X)))`.

9. Explain the spectral lines that appear in one of the FFTs.
10. Relate the spectral lines to the fundamental frequency (pitch) of the vowel.
11. Is the speaker more likely to be a man or a woman?

One period of the vowel is approximately equal to the impulse response of the vocal tract for this vowel. Therefore, it will capture the frequency characteristics of the vocal tract. From the DFT of one period, identify the most important formants and calculate their analogue frequencies (you may need to change the selected period of the vowel to observe the formants better).

12. What is the analog frequency of the most important formant?
13. What is the analog frequency of the second formant?
14. Load any other 'unknownX.wav' file, and following a similar procedure, relate the formants observed in the DFT plot to the vowels diagram and try to guess which vowel is.

"What's in a name?"

Finally, take one of the two name.wav files recorded by you and your partner (if you have time you can work with both of them) and inspect its characteristics. Can you distinguish the voiced/unvoiced segments? Select a segment containing a vowel, estimate the fundamental frequency, formants, etc.

15. Explain your observations/estimations and upload the .wav file used for the analysis.