University of Washington ECE Department
EE 235 Lab 5 – Frequency Domain

In this lab, we will learn how to transform signals from the time domain to the frequency domain and try to identify the frequencies of $e^{j\omega t}$ that comprise a periodic signal. The concepts we'll focus on are finding the Fourier Series coefficients a_k of x(t), identifying frequency components of x(t), and understanding the relationship between Fourier Series coefficient index k and frequency ω.

Lab 5 Turn-in Checklist

- 2 pre-lab exercises
- 3 lab assignments with 2 TA check-offs
- Lab report, submitted as a Jupyter notebook following the format provided in earlier labs

Note: All assignments except the prelab should be completed in groups of 2-3 people.
The pre-lab exercises are to be completed individually and submitted via Canvas before lab section.

Pre-lab

Read the Lab 5 Background document, then complete the following exercises.

1. Consider the sum of sinusoids **d0(t) = sin(2π(941)t) + sin(2π(1336)t)**

   a. Find the Fourier Series representation of this signal, i.e. its Fourier series coefficients $\mathbf{a_k}$, given that $\mathbf{w_o = 2\pi}$. (No need to prove $\mathbf{w_o = 2\pi}$.**)** For what values of k are the coefficients non-zero?

   b. Each Fourier series coefficient index **k** (with nonzero $\mathbf{a_k}$) corresponds to a frequency $\boldsymbol{\omega} = \mathbf{kw_o}$ in radians. The relationship between frequency in radians **ω** and frequency **f** in Hertz (Hz) **is ω=2πf.** What are the frequencies of the two sinusoids in Hz?

   c. If sample this sine wave with fs=8000 Hz, and then you take its Fourier transform using an FFT with 4096 points, what is the frequency in Hz of the nearest FFT sample to the true frequency? Find the value for each of the two sinusoids above.

2. The telephone touch-tone system uses signals composed of different frequencies to indicate the digit pressed. This signaling scheme is called "dual-tone multi-frequency" (DTMF) signaling. In DTMF signaling, each touch-tone on the keypad is represented as the sum of two sinusoidal tones, where one tone is at a low frequency and the other is high. DTMF keypad frequency pairs are given below:

	High frequencies			
Low frequencies		1209 Hz	1336 Hz	1477 Hz
	697 Hz	**1**	**2**	**3**

	770 Hz	**4**	**5**	**6**
	852 Hz	**7**	**8**	**9**
	941 Hz		**0**	

For example, to generate key '0', we need the signal **d0(t)** above. To generate key '2', we need the signal **d2(t) = sin(2π(697)t) + sin (2π(1336)t).**

Write Python code to create a 2-D Numpy array **tone_freqs**, where each frequency pair is ordered (lower frequency, higher frequency).

Lab Assignments

This lab has 3 assignments. Each should be given a separate code cell in your Notebook, and each should be associated with a markdown cell with subtitle and discussion. As in previous labs, your notebook should start with a markdown title and overview cell, which should be followed by an import cell that has the import statements for all assignments. In this lab, you will need to use most of the import commands that you have used in earlier labs. As always, you should add comments to your code for clarity.

You will again be working with concepts from previous labs, so you may want to refer back to the background files for those labs or the py_ref document.

Assignment 1: Identifying Component Frequencies of a Signal

In this Assignment, we'll create and analyze the signal d0(t) from the prelab. Write your code in a new cell for Assignment 1, inside your Lab 5 notebook.

A. Using sampling frequency **fs=8000**, create a time samples vector **t** for $0 \le t < 0.25$. Use this vector to create **d0(t) = sin(2π(941)t) + sin(2π(1336)t)**.

B. With the *simpleaudio* module, write a WAV file where **d0** is played twice, with an 0.5-second pause in between. You will need to amplify the signal to make it audible; try a scaling factor of 50. Play the file to hear the telephone tones.

C. Find the Fourier Transform **d0_ft** of **d0** using the numpy fft and fftshift functions. Create a frequency vector **f** that corresponds the positive and negative frequency range. Create a <u>2x1</u> subplot, and plot **d0_ft vs. f** as Plot 1.

D. If this was an ideal continuous-time Fourier transform, then all **d0_ft(i)=0** for **i** other than the sinusoid frequencies. However, because of time and frequency sampling on the computer, you can get non-zero values. Also, if this was a real communications problem, the signal would have some noise. So, in order to automatically find the component frequencies in the signal from the DFT, we'll do an energy threshold test.

a. Use the masking strategy described in the background document with a threshold of 900 to find the d0_ft array indices associated with FT peaks. Convert the resulting indices to Hz.

b. Sinusoids should have symmetric positive and negative frequency complex exponentials, but we are only interested in the positive frequencies. Print the values of the positive frequencies.

E. As in lab 4, create a noise vector that is the same length as the signal, using a standard deviation of 1

```
n = np.random.normal(0,1,len(d0))
```

and add it to **d0** to create **d0n**. Compute the DFT as **d0n_ft** and plot **d0n_ft vs. f** as Plot 2

Assignment Check-Off #1 of 3: Demonstrate this Assignment to the lab TA by playing the audio file and showing the FFT plots.

Report discussion:
To identify component frequencies, we used a threshold value of 900. Using your FFT magnitude plot, explain why a threshold of 100 would not work. Would that threshold identify more or fewer frequencies than the threshold of 900?
Comment on how the addition of noise changes the FFT plot.
What would the FFT look like if instead the signal had been
$\mathbf{v(t) = 1 + \sin(2\pi(941)t) + \sin(2\pi(1336)t)}$?

Assignment 2: Classifying Touch-Tone Telephone Signals

In this assignment, you will write a function to classify a tone signal by finding its component frequencies, then test it. Use a separate cell for the function and the testing. in-line comments and good form to keep your logic clear.

A. In a separate cell, define a function **classify** which takes inputs: tone signal **x** and sampling rate **fs**. The output should be the digit associated with the signal.

a. Using pre-lab, define the tone frequency matrix **tone_freqs**. Remember to order the pairs.

b. Use the procedure outlined in Assignment 1 to extract the component frequencies from tone signal **x**, with **N = 4096** fft samples and a threshold value of **750.** Use the variable names given below:

- **xfft_abs** as the shifted magnitude output of the **FFT** of **x**.
- **index_tone** as the indices in **xfft_abs** where the element of **xfft_abs** is greater than a threshold value **900**.

- **freq_tone** as the array of frequencies in **xfft_abs** with spikes exceeding the threshold value. **freq_tone** should be in Hz.

Recall that sinusoids should have symmetric positive and negative frequencies. Store last positive frequencies of **freq_tone** in **pos_freq**.

c. Now that we have the two component frequencies of **x**, we can classify its touch tone using your pre-lab code. Write a loop that does the following:

- Loop through the digits, and find the total squared error measurement **digit_error** between the touch-tone frequency pair and **pos_freq** (this error measurement is the sum of the two squared errors: $(f_L-f_L(d))^2+(f_H-f_H(d))^2$, where f_L is the frequency of the low tone, f_H is the frequency of the high tone and $f_*(d)$ are the corresponding frequencies detected in the signal.
- Track the minimum error of the frequency pairs evaluated so far as **min_e**, with the corresponding touch tone **min_e_tone**, by comparing the current total error to our existing **min_e** and updating as needed.
- Return the touch-tone digit identified by **min_e_tone**.

B. Start a new cell. Using time sample vector **t** for the range **0 ≤ t < 0.25** and sampling rate **fs = 8000**, test the **classify** function. Create the tone signals **d4**, **d5**, **d8**, and **d9**, and run classify on each to verify that the correct digit is detected.

Assignment Check-Off #2 of 2: Demonstrate **classify** function test to the lab TA

Report discussion: If the signal had noise, as in assignment 1, what could you do to ensure high detection accuracy?

Assignment 3: Decoding a Phone Number from Touch-Tone Signals

The ECE head TA wants to tell you about a San Diego restaurant, but would rather you decode its phone number from a sound file. You can use your function from assignment 2. As always, start a new cell for this assignment.

A. Load file **phone.csv**. The file has 11 time slices corresponding to 11 digits. Each individual time-slice has 0.25 seconds of the DTMF signal followed by 0.25s of pause or silence. The sample frequency used for creating the file was 8kHz.

B. We will be decoding the phone number in a loop, so we'll need a buffer to store each digit as we decode it. Declare a vector **phone_num** with 11 elements, each element **-1**.

C. Write a loop to fill in **phone_num**. For each element (digit), you will extract one of the **11** time slices (consisting of a tone signal and a silence signal), decode it, and print out the digit. For each time slice, compute the corresponding start and end indices (**start_index**, **end_index**) in **x**. Use these to extract the data samples for that time slice, storing them to **signal**. Classify the samples, and save your decoded tone number to the appropriate index in **phone_num**.

D. Print the sequence of digits.

Report discussion: Provide the decoded message in your lab report. Optionally, if you are ambitious, figure out the name of this well-know fast-food restaurant.

Team Report

When you've tested and cleaned up all your code (remember, you should only submit code for the Assignments, each in their own cell), go to 'File' then 'Download as', then select '.ipynb'. The file you download is a Notebook that your TA will be able to open and grade for you, once you submit it on Canvas. Remember, only one notebook per team! Make sure that your notebook is titled Lab5-XYZ.ipynb, where XYZ are the initials of the lab partners. You may want to also download the file as pdf to have a nicer documentation of your records.

Submit the .ipynb file via Canvas.