University of Washington ECE Department

EE 235 Lab 1 – Introduction to Python and Signals

In this lab, you will work through a series of exercises to introduce you to Python with an emphasis on the basics that you need to know for working with signals. You will learn to use variables, perform basic operations, and write your own scripts to run in Jupyter Notebook.

Lab 1 Turn-in Checklist

- Pre-lab: Show that you have installed or can access the necessary software by creating a Jupyter notebook that demonstrates basic Python functions and uploading it to canvas.
- 5 Assignment check-offs with TA
- Lab 1 Jupyter notebook with code for each assignment in individual cells and with markdown cells used for the lab overview and response to report questions.

Note: All assignments except the prelab should be completed in groups of 2-3 people.

Pre-lab

Read Sections 1-3 of the Background document. Set up your laptop in order to be able to use Python and Jupyter notebook. Section 2 describes one option, which is to use Anaconda. You can also use EE remote desktop.

Create a Jupyter notebook (prelab1.ipynb) that demonstrates the use of print, playing audio, and creating documentation using a markdown cell to your TA, as outlined in section 3 of the Background document. You will need to download the audio file provided on the Lab 1 assignment web page and save it in the same directory as your Lab 1 notebook. Save the prelab1 notebook, and submit the file to the lab 1 assignment on Canvas.

Download the template notebook for lab 1 (TemplateL1.ipynb) and rename it Lab1-XYZ, where XYZ are the initials of the lab partners. Your Lab 1 notebook should be organized according to this template.

Lab Assignments

This lab has 5 assignments. Each should be given a separate code cell in your Notebook, and each should be followed by a markdown cell with discussion. 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.

Assignment 1: Creating and Manipulating Arrays

Read Section 4 of the Background document. This assignment will have four parts, A-D, each of which should be indicated with comments, following the guidelines in the **Lab 1 template**. In creating arrays, you should use *numpy,* so don't forget to import it.

A. Create an array **y1** with values 4, 6, 2.

B. Extract the value 6 from **y1** to store in **ca.** In **cb**, store the last two elements of **y1**. In **d**, store the number of elements in **y1**. Print **ca, cb** and **d**.

C. Create two arrays of length 5. The first (**x1**) should have all elements take the value 2, and the second (**x2**) should range from [-2,2]. Use the **ones** and **arange** attributes of *numpy*. Print **x1** and **x2**.

D. Create **arrp = x1+x2,** and **arrc** as the concatenation of **x1** and **x2.** Print **arrp** and **arrc.**

Run the code in this cell and verify that the results are what you expect.

Assignment Check-Off #1 of 5: Demonstrate this Assignment to the lab TA

Report discussion: Comment on the difference between concatenation and addition of two vectors. What are the constraints in terms of when these functions can be applied?

Assignment 2: Amplitude Operations on Signals

Read Section 5.0-5.2 of the Background document. Again, following the guidelines in the **Lab 1 template**, start a new cell and write a script to meet the following specifications.

A. Create digital versions **x** and **y** of the functions below over $0 \le t \le 3$, with a sampling frequency $f_s = 2$.

$$x(t) = 0.5t \qquad\qquad y(t) = t^2$$

B. Assign **z** to be the digital version of $z(t) = x(t) - 2y(t)$

C. In **w1**, store the value of $z(t)$ at $t = 2$. Print **w1**.

D. Store the values of **z** for $0 \le t \le 1.5$ in **w2**. Print **w2**.

Note: You may need to cast your indices to integers. Use *index = int(index)*

```
x = [ 0.     0.25   0.5    0.75   1.     1.25   1.5 ]
y = [ 0.     0.25   1.     2.25   4.     6.25   9.  ]
z = [ 0.    -0.25  -1.5   -3.75  -7.    -11.25 -16.5 ]
w1 =  -7.0
w2 = [ 0.    -0.25  -1.5   -3.75]
```

Show that your output matches the below:

Assignment Check-Off #2 of 5: Demonstrate this Assignment to the lab TA

Report discussion: Discuss how the results of this problem would change if you used $f_s=1$ instead of $f_s=2$.

Assignment 3: Working with Sound Files

Read Section 5.3 of the Background document. Start a new cell following the guidelines in the **Lab 1 template**, dividing it into Parts A-D.

A. Read in the files "train32.wav" and "tuba11.wav" using either the scipy wavfile package or simpleaudio, saving the audio vectors and sampling frequencies in variables **y1, y2, fs1, fs2,** respectively. Print the number of channels to confirm that the train file has 1 channel and the tuba file has 2.

B. Play the two sounds in their correct form and then with the sampling frequencies swapped.

C. Determine the length of the two signals, and extract a section of the longer one that is the same length as the shorter one, called **y3**. Add two equal-length, single-channel signals to create **y4**. Play and write out **y4** using sampling frequency **fs1**.

D. Concatenate the signals in **y1** and **y2**, adding a **4 second pause** between them. Call the new signal **y5.** Play the new signal and write it out to a file using sampling frequency **fs1**.

Assignment Check-Off #3 of 5: Demonstrate this Assignment to the lab TA by playing the signals **y4** and **y5**

Report discussion: Comment on how the audio changes when the incorrect sampling frequency is used. You should notice two ways in which the sound changes. The train sound is mono, and the tuba sound is stereo. How did that impact your solution to parts C and D?

Assignment 4: Plotting Comparisons

Read Section 6 of the Background document. Start a new cell following the guidelines in the **Lab 1 template**, dividing it into Parts A and B.

A. Define a time vector signal over **-2 ≤ t ≤ 4**, let **fs = 5**. Then create digital versions of the following signals for this time span:
$w(t) = |t|$
$x(t) = 2 - t$
$y(t) = -0.5t^2$

B. Plot all three signals in one figure with a grid and a legend. Be sure to label the time axis and title the figure.

Assignment Check-Off #4 of 5: Show the plots to the lab TA

Report discussion: Read the plot documentation and explain how you would change the command to distinguish the signals using different line styles (solid, dashed, dotted) to make the figure easier to read for someone who is color blind.

Assignment 5: Plotting Sound Files in Subplots

Start a new cell following the guidelines in the **Lab 1 template**, dividing it into Parts A and B.

A. Create time sample vectors for the audio signals from Assignment 3 (**y1** and **y2)** using the original sampling frequency for each, and name them **t1** and **t2. (**You will need to use the signal lengths you found in that assignment.)

B. Plot the two signals side by side using a **1x2** subplot. Adjust the x and y axis limits to have the same ranges in both plots. Be sure to title the plots and label axes appropriately. For the time axis, you should plot time in msec, for which you will need to multiple the time array by 1000 as in the example in the **Background** document. Do not use grids on the subplots. Adjust the spacing to avoid overlap.

Assignment Check-Off #5 of 5: Show the plots to the lab TA

Report discussion: Comment on the differences between the plots of the signals. Can you see the difference in sampling rate in these plots?

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 Lab1-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.