ECE 351 - Lab 3 - Discrete Convolution

Ethan Reeder

2 February 2021

Contents

1 Purpose

The purpose of this lab was to get more familiar with convolution and how it works by using Python to graph and calculate various convolutions.

2 Procedure

This lab began by using the code for the step and ramp functions from last lab to create the following functions in Python:

$$f_1(t) = u(t-2) - u(t-9) \tag{1}$$

$$f_2(t) = e^{-}t * u(t) \tag{2}$$

$$f_3(t) = r(t-2)[u(t-2) - u(t-3)] + r(4-t)[u(t-3) - u(t-4)] \tag{3}$$

The code for these functions in Python is shown in Listing 1 in the Methodology section, and the plots for these functions are in Figure 1 in the Results section.

Next, a function was created to convolute two functions with one another. This code was then used to convolve f_1 with f_2, f_2 with f_3, and f_1 with f_3. These convolutions were then compared with the convolutions resulting from using the scipy.signal.convolve() function. The graphs for the convolutions are in Figures 2 through 4 in the results section, and the code for the convolution function is shown in Listing 2 in the Methodology section.

3 Methodology

Code for defining the three functions using step and ramp functions:

```
def f1(t):
    return (step(t-2) - step(t-9))

def f2(t):
    return (np.exp(-t) * step(t))

def f3(t):
    return (ramp(t-2)*(step(t-2) - step(t-3)) +
```

```
        ramp(4-t)*(step(t-3) - step(t-4)))

steps = 1e-2
t = np.arange(0,20+steps,steps)
NN = len(t)
tExtended = np.arange(0, 2*t[NN-1], steps)

x = f1(t)
y = f2(t)
z = f3(t)
```

Code for my defined convolution function:

```
def convolute(f1, f2):
    Nf1 = len(f1)
    Nf2 = len(f2)
    f1Extended = np.append(f1, np.zeros((1, Nf2 -1)))
    f2Extended = np.append(f2, np.zeros((1, Nf1 -1)))
    result = np.zeros(f1Extended.shape)

    for i in range(Nf2 + Nf1 - 2):
        result[i] = 0
        for j in range(Nf1):
            if(i - j + 1 > 0):
                try:
                    result[i] += f1Extended[j]*f2Extended[i - j + 1]
                except:
                    print(i , j)

    return result
```

4 Results

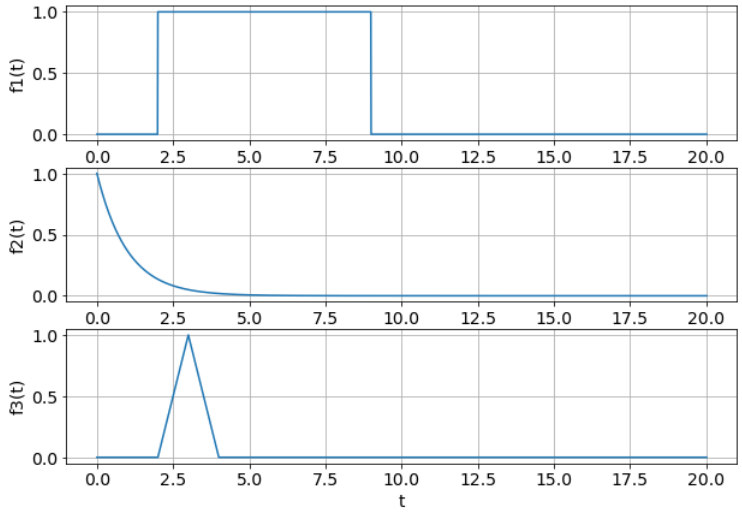

Figure 1: Plots of f1, f2, and f3

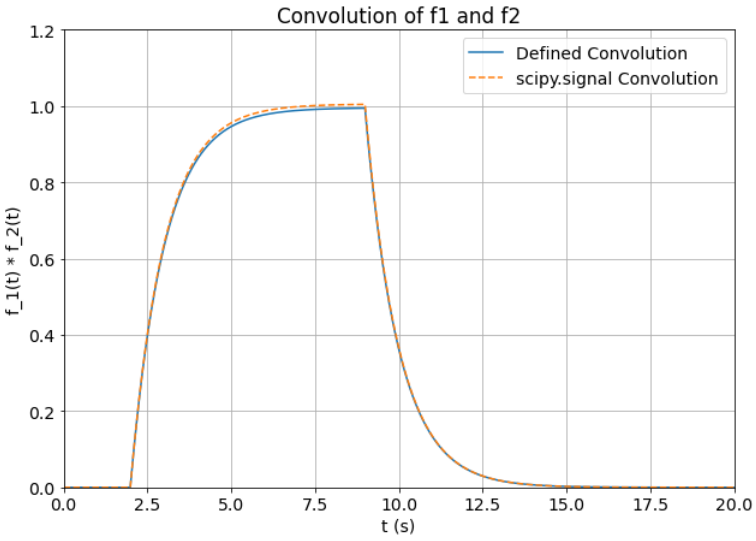

Figure 2: Plots f1 convolved with f2

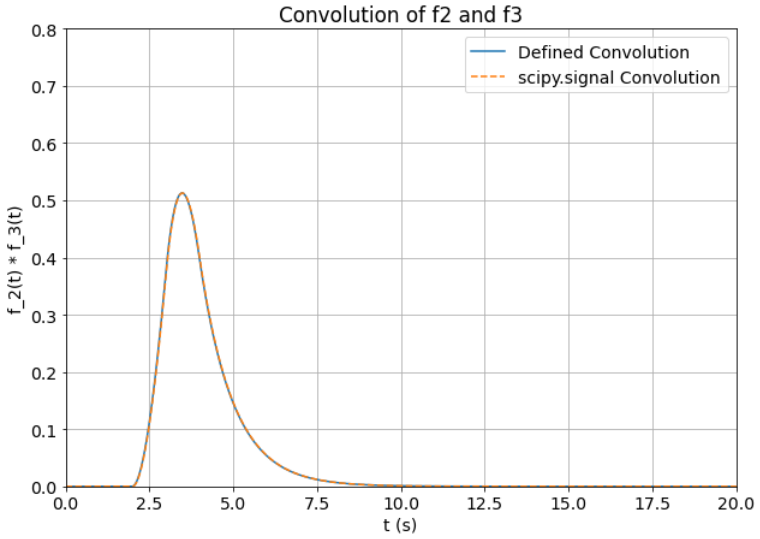

Figure 3: Plots f2 convolved with f3

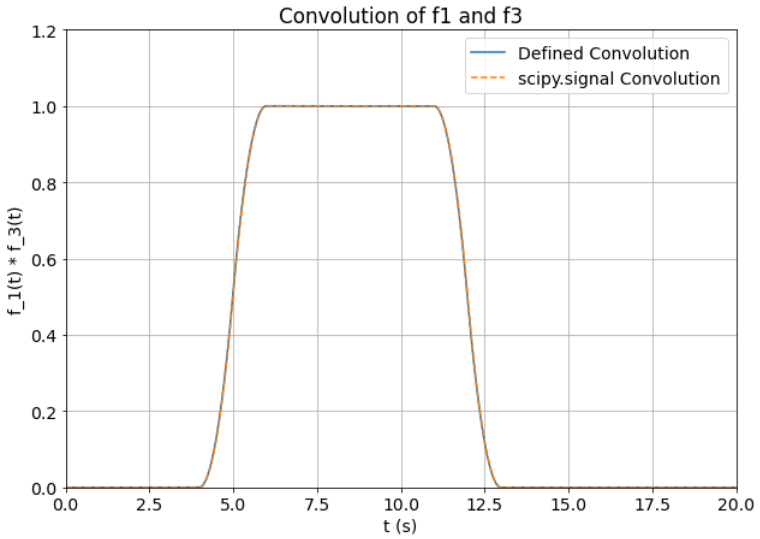

Figure 4: Plots f1 convolved with f3

5 Questions

1. Did you work alone or with classmates on this lab? If you collaborated to get the solution, what did that process look like?

I worked with Austin Crofoot a little on this lab. We were in the same room on the Zoom call together, but were both coding separately on our own laptops. Whenever either of us hit a wall in the coding or encountered a weird error that we couldn't figure out on our own, we collaborated to get past that obstacle.

2. What was the most difficult part of this lab for you, and what did your problem-solving process look like?

The most difficult part of this lab was wrapping my head about how to be explicit about what's happening with a convolution. Similar to how in Lab 2, it was a little interesting to get the differential function to perform properly, it was difficult to get the convolution function to work, since I'm not used to having to be very clear about each and every step in what's happening, and instead just have a general idea of what's going on for convolution.

For problem-solving, though I wasn't able to get a fully-working convolution function on my own, it involved breaking the entire operation down into more easily manageable and code-able chunks, such as determining the length of the two functions separately to then get the length of the entire convolution.

3. Did you approach writing the code with analytical or graphical convolution in mind? Why did you choose this approach?

I approached writing the code with graphical convolution in mind. While I understand them both relatively the same amount, it's a lot easier for me to visualize what's happening in a graphical convolution, which in turn makes it easier for me to put into code what I'm trying to get to happen.

4. Leave any feedback on the clarity of lab tasks, expectations, and deliverables.

The lab was clear and the expectations were understood.

6 GitHub Link

https://github.com/EReeder35

7 Conclusion

In conclusion, this lab was a great opportunity to further my knowledge with Python, and also allowed me to deepen my understanding on what exactly a convolution is. Though I wasn't able to come up with my own original code for the convolution function, I do understand what's going on in the code that the TA supplied us at the end of the lab, and also understand why the convolution graphs look the way they do.