Discrete Convolution Visualization Utilizing a Jupyter Notebook

Edward Carl Greco, Jr. and Zahra Zamanipour
Electrical Engineering Department, Arkansas Tech University

Abstract

To engage students in active learning of discrete convolution, a Jupyter notebook was generated and shared with students in the Signals & Systems course. Discrete convolution operation and its properties along with two general examples, medication administration, and the statistical probability density distributions, are presented in this paper. The examples were chosen to be different from standard examples in textbooks so that students would appreciate the application of convolution to other areas. The notebook provided a mathematical and visual explanation of the convolution operation. Python codes were developed, and their results were embedded in the Jupyter notebook. The students were encouraged to modify and rerun the codes in real-time. The notebook environment provides an excellent opportunity for them to learn and validate their understanding of the basic concepts of discrete convolution. Students received credit for using the notebook and answering survey questions anonymously. Results showed that students found that this Jupyter notebook was effective in helping them understand discrete convolution and its properties.

Keywords

Jupyter Notebook, Visualization, Discrete Convolution, Active Learning

Introduction

Convolution provides the mathematical framework for linear systems in the Signals & Systems course, while it has also found utilization in mathematics of many fields, such as probability, statistics, and image processing for feature extraction using convolutional neural networks (CNN). It involves combining two signals to obtain a third for example where an input signal is convolved with the impulse response of a linear time-invariant system to obtain the output response [1,2]. It is possible to easily express the convolution operator mathematically; nevertheless, the concept behind it is frequently difficult to comprehend. The purpose of this work was to provide students with a better understanding of the convolutional concept by using a Jupyter notebook in Colab (Google Collaboratory). Jupyter notebook combines executable code in an environment with rich text in a single document, and the notebook can be stored on a Google Drive or on GitHub [3] to provide a convenient format to share with others. Jupyter notebook on Colab used Python3 as the default kernel, and all code segments were written in Python3. However, it was not necessary to be proficient in Python to execute the code in the notebook or to make changes to the input data to observe the results of those changes.

Methodology

Convolution is the mathematical operation that combines two functions to obtain a third. It can be applied to both continuous and discrete time signals. Although convolution is also applicable to continuous signals with integral calculus, discrete signals were utilized in this study. The mathematical definition of convolution followed by two practical examples are presented in the notebook including: the administration of daily medication and the probability density distribution of two or more independent random variables such as sum of the face values on two or more dice. Convolution operator properties, namely commutative, distributive, and associative properties, and the superposition principle are explained mathematically and applied to the examples with detailed explanation. The convolutions and their properties were coded with Python [4] for presentation in the Jupiter notebook. These interactive computations can be run in real-time while modifications were possible. The Python libraries namely numpy, scipy, pandas, and matplotlib were used in this work and must be loaded prior to the execution of subsequent code segments. Results of the codes are shown in the plots visually indicating the convolution operator concept. The full description of the developed Jupyter notebook, available on GitHub [5], is explained in the following paragraphs.

Mathematical Definition

Discrete Convolution is a mathematical operation that maps two discrete sequences, x(n) and h(n), into a third discrete sequence, y(n) [2].

$$y(n)=h(n) * x(n) - \infty < n < +\infty$$

$$y(n)=\sum_{k=-\infty}^{\infty} h(k)x(n-k)=\sum_{k=-\infty}^{\infty} h(n-k)x(k)$$

Where y(n) is the result of the convolutional sum of h(n) with x(n), where " $*$ " represents the convolutional operator. If both sequences are causal, that is x(n)=h(n)=0 ∀ n<0, then the limits on the summation are restrained between 0 and n.

$$y(n)=h(n) * x(n)=\sum_{k=0}^{n} h(k)x(n-k) \forall n \geq 0$$

Convolution is a linear operation and has the associative, commutative, and distributive properties which will be explored in detail below [6,7].

Convolution can be applied to areas other than signal processing and can best be described by examples. Consider a patient who receives medication over several consecutive days with a different dose each day. We will see that the number of units of medication administered to a group of patients over several days can be obtained from the convolution of the daily medication dosage regime with the patient population.

In another example, assume you have two fair dice. When the fair dice are rolled, the probability that any number 1 to 6 appears at the face value of each die is 1/6 since there is an equal probability that any of the six possible values will appear. Therefore, the sum of the face values of the two dice will be in the range between two and twelve where two is the result of two ones and twelve from two sixes. It will be shown that the probability density function of the sum of the face values is the convolution of the two individual probability distribution sequences, p(n) = 1/6 for n=1, …, 6, where p(n) is the probability for each side of a fair dice.

First consider the following examples that will show how convolution describes the amount of medication required for a group of patients over a series of several days. These examples will verify the linearity, commutative, and distributive properties of convolution.

Daily Medication Delivered

The medication administered daily can be obtained from the number of patients admitted each day combined with the units of medication administered daily to each patient. Assume the medication was given to each individual patient over a three-day sequence with patients admitted over a four-day period.

On the initial day: the units of medication administered, y(0), is obtained from the number of patients admitted on the initial day, x(0), times units of medication for the initial day, h(0) (indexing begins with 0 corresponding to initial day).

On the following (second) day: the units of medication administered, y(1), is obtained from the patients admitted on following day, x(1), times the units of medication received for their initial day's dosage, h(0), plus the patients admitted on the initial day, x(0), times the units of medication for their following day's dosage, h(1). The number of units of medication administered for each day is defined by the following equations. The daily medication administered over consecutive days is summarized in the following equations:

$$y(0)=h(0)x(0)$$
$$y(1)=h(0)x(1)+h(1)x(0)$$
$$y(2)=h(0)x(2)+h(1)x(1)+h(2)x(0)$$
$$y(3)=h(0)x(3)+h(1)x(2)+h(2)x(1)$$
$$\vdots$$
$$y(n)=\sum_{k=0}^{n} h(k)x(x-k)$$

where daily medication units, y(n) in general obtained from h(n) (daily medication units) and x(n) (daily patient admissions).

Figure 1 shows the convolution of the daily medication administered to a group of patients admitted over a four-day sequence. The number of patients admitted each day is contained in array x, and the units of medication administered each day is defined by array h. The Jupyter notebook including the Python code is located on GitHub and the link to it is provided in reference [5].

Commutative Property of Convolution

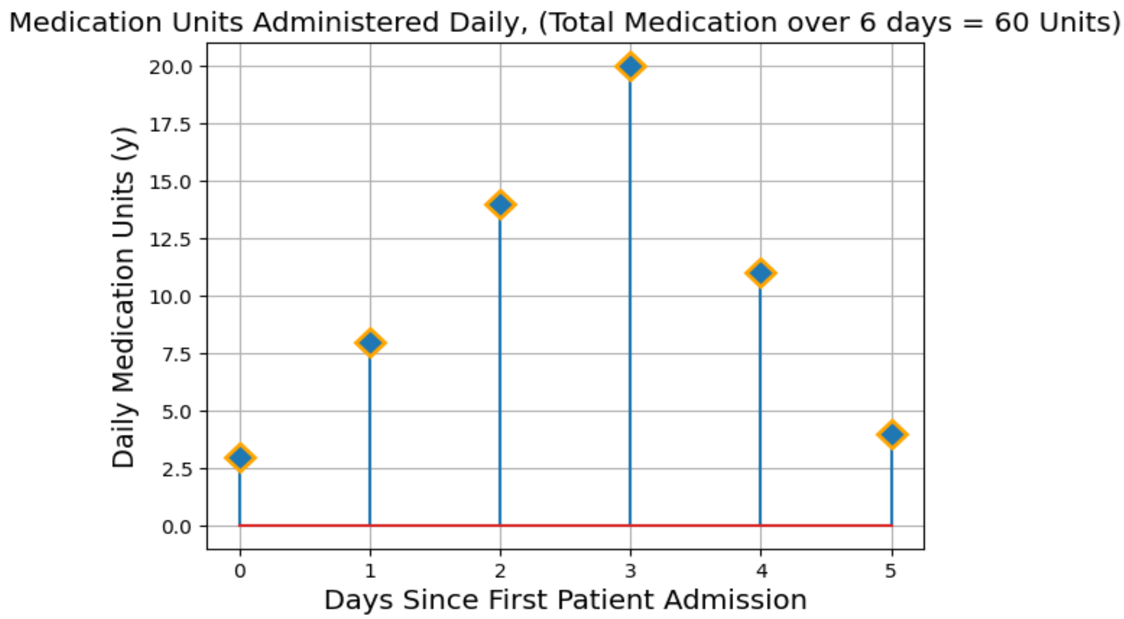
Convolution has a commutative property, so its order does not affect its result. This is demonstrated mathematically below.

$$y(0)=h(0)x(0)=x(0)h(0)$$
$$y(1)=h(0)x(1)+h(1)x(0)=x(0)h(1)+x(1)h(0)$$

$$y(2)=h(0)x(2)+h(1)x(1)+h(2)x(0)=x(0)h(2)+x(1)h(1)+x(2)h(0)$$

$$\vdots$$

$$y(n)=\sum_{k=0}^{n} h(k)x(n-k)=\sum_{k=0}^{n} x(k)h(n-k)=h(n) * x(n)=x(n) * h(n)$$




```
Daily Patient Admissions       : x(n) = [1, 2, 3, 4]
Daily Medication Dosage        : h(n) = [3, 2, 1]
Daily Medication Admisistered  : y(n) = h(n) * x(n) = [ 3  8 14 20 11  4]
```

Figure 1- The units of medication administrated daily is the resultant convolution of the sequence of patients admitted daily, x(n), with the units of medication given to each patient daily, h(n). The total units of medication delivered over the course of six days is the cumulative sum of the daily units.

Linearity of Convolution Operator

Convolution operators satisfy both additivity and homogeneity (scaling) properties required for a linear operator. In our medication example, if the number of patients was scaled by a constant factor each day, then the medication administered increased by that same amount. For example, increasing the number of patients by two increased the medication by two.

$$y_{scaled}(n)=h(n) * (2\times x(n))=2\times(h(n) * x(n))=2\times y(n)$$

$$y_{scaled}(n)=\sum_{k=0}^{n} h(k)(2\times x(n-k))=2\times x(n-k)=2\times\sum_{k=0}^{n} h(k)x(n-k)=2\times y(n)$$

Scaling the input scales the output by the same factor which satisfies the homogeneity (or scaling) property of convolution.

Similarly, to verify the additivity property, it assumed that medication is administered for two groups of patients: x1-patient group 1, x2-patient group 2, and h-medication dosage, and mathematically, we have:

$$y_{1+2}(n)=h(n) * (x_1(n)+x_2(n))=h(n) * x_1(n)+h(n) * x_2(n)$$

$$y_{1+2}(n)=y_1(n)+y_2(n)$$

which verifies the additivity property. The convolution operator meets the properties of additivity and homogeneity (scaling), verifying it as a linear operator. The linearity properties were demonstrated graphically in the Jupiter notebook.

Distribution Property

To verify the distribution property of convolution operator, it is assumed that patients are given separate medication in the mornings and evenings:

$$y_{Morning}(n)=h_{Morning}(n) * x(n)$$

$$y_{Evening}(n)=h_{Evening}(n) * x(n)$$

$$y_{DailyTotal}(n)=h_{Morning}(n) * x(n)+h_{Evening}(n) * x(n)=(h_{Evening}(n)+h_{Morning}(n)) * x(n)$$

Where $y_{Morning}(n)$ (morning) and $y_{Evening}(n)$ (evening) medication dosage and $y_{DailyTotal}(n)$ is the combined daily dosage for the patient population.

The distribution property of convolution is verified by a real-time code embedded in the notebook. Figure 2 demonstrates this verification where the morning medication is shown in Figure 2(a), the evening medication in Figure 2(b), and the total daily medication in Figure 2(c).

Associative Property of Convolution

The relationship between the input and output of a system with cascaded processes is independent of the order in which the processes occur. Assuming two separate processes, $h_1(n)$ and $h_2(n)$ in the first equation the input, x(n), is convolved with $h_1(n)$ followed by the convolution with $h_2(n)$. In the second, the input $h_2(n)$, is convoluted with $h_1(n)$ followed by the convolution with x(n). The result is independent of the order of convolution. This is the associative property of convolution.

$$y(n)=h_2(n) * (h_1(n) * x(n))=h_2(n) * w_1(n)$$

$$y(n)=(h_2(n) * h_1(n)) * x(n)=h(n) * x(n) where h(n)=h_2(n) * h_1(n)$$

Discrete convolution is applicable to non-time domain signals. For example, the face value on a rolled dice can be modeled as random variable with an associated probability. The probability that the number, n = 1 to 6, obtained from a single fair die is p(n) = 1/6. For two dice the

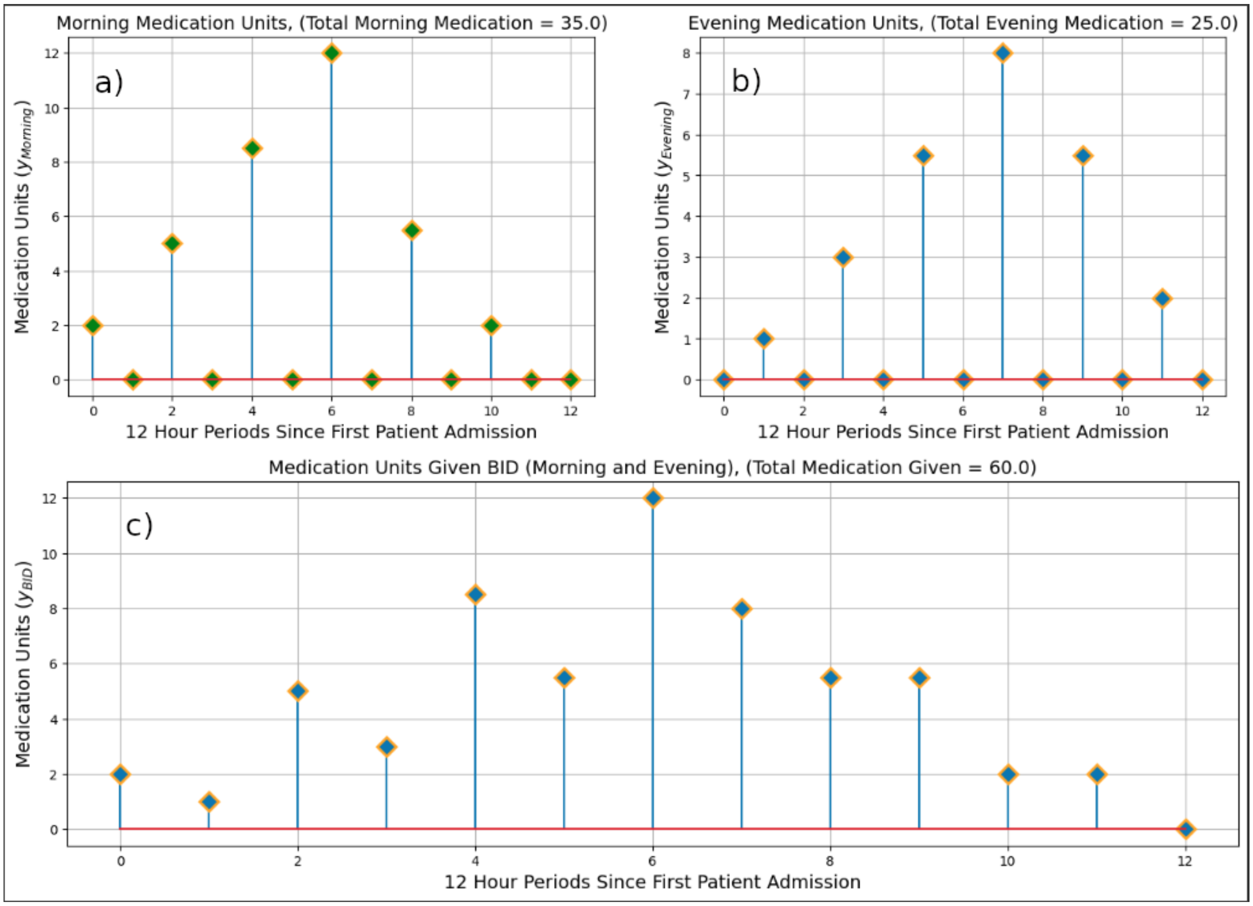


Figure 2- Verification of distribution property of convolution operator

probability of the sum of their values is the convolution of the sequences $p_1(n) = p_2(n) = 1/6$, where these are the probabilities that the face number n on the die is [1, 2, 3, 4, 5, or 6]. There is one possible combination for sum = 2: [(1,1)], and sum = 12: [(6,6)], p= (1/6) (1/6) = 1 /36. There are two combinations for sum = 3: [(1,2), (2,1)], p= (2/36), and three possibilities for sum = 4: [(1,3), (2,2), (3,1)], p= (3/36), up to six possibilities for sum = 7, p= (6/36). Therefore, probability density distribution for the random variable representing the sum of the face values of two dice is discrete convolution of the individual probability density distributions for each dice.

If a third dice is included in the roll, the sum of their values will be in the range (3 to 18), and the probability density distribution for the random variable representing their sum is the discrete convolution of the three individual distributions. Assumed dice is biased such that the probability of one side is 1/2 the probability of the opposite side. Since opposite sides sum to seven, if a one is twice the probability of a six then the unfair dice will have the probability sequence in p_3 shown below. The probability density function (pdf) for the sum of the values on the three dice is given by the following equations.

$$pdf(n) = \left(p_1(n) * p_2(n)\right) * p_3(n) = w_1(n) * p_3(n)$$

$$pdf(n) = \left(p_1(n) * p_3(n)\right) * p_2(n) = w_2(n) * p_2(n)$$

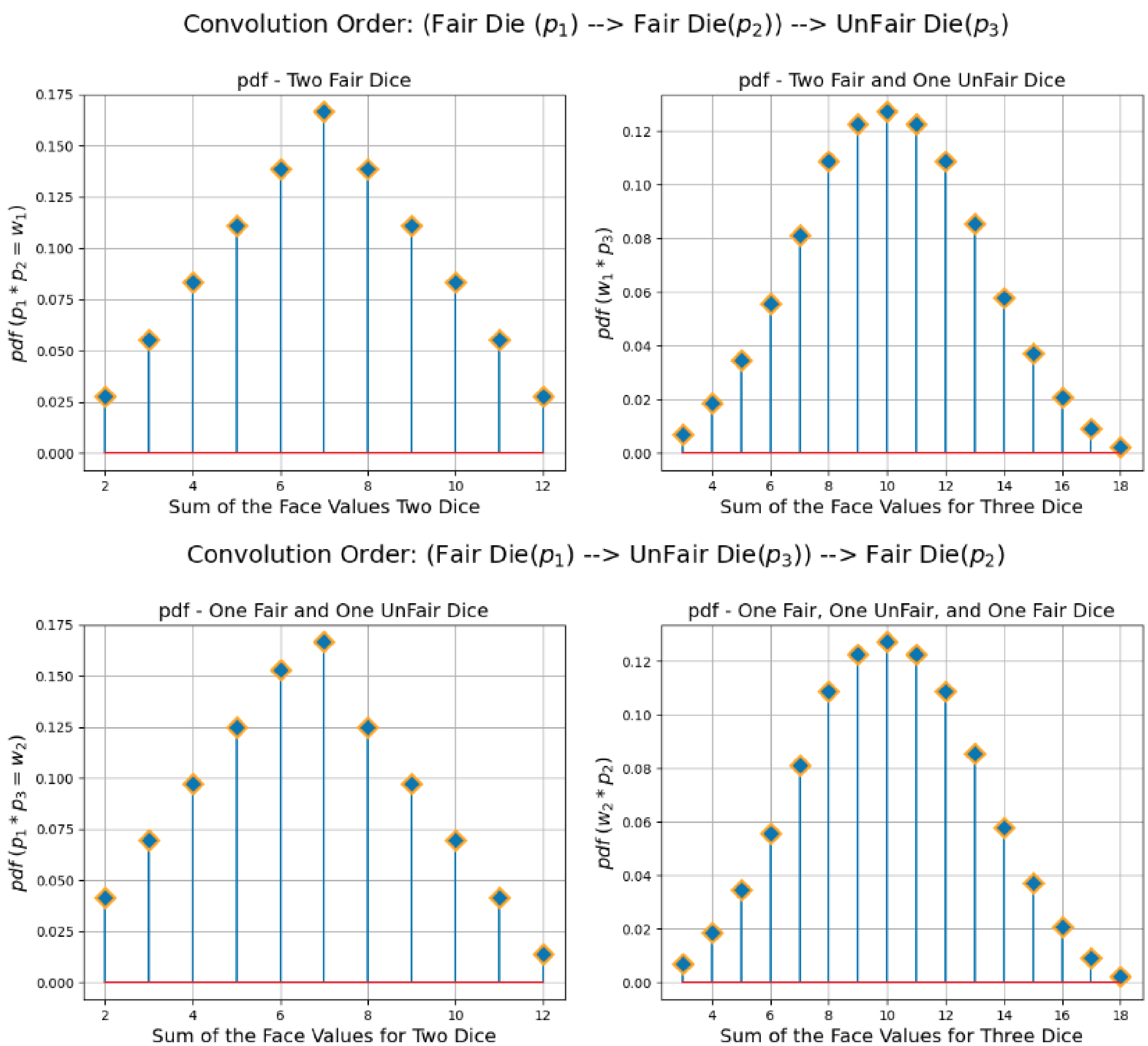


Figure 3- Verification of associative property of convolution. The right most final pdf distributions are the same for both the top and bottom although the left most initial distributions differ.

Figure 3 compares the probability density function of convolution of two fair dice with one unfair dice versus convolution of one fair dice and one unfair dice with one fair dice. Results show the final probability density functions are independent of the order of convolution.

Results and Discussions

Jupyter notebook was shared with students in a Signal and Systems course. They were asked to review it to reinforce and enhance their understanding of convolution and its properties. The codes are embedded in the notebook for real-time, interactive simulation. It provided the students the opportunity to modify the inputs and see the results immediately in the notebook. Once the students had reviewed the notebook, they were asked to fill out a questionnaire based on their experiences. Table 1 shows the six questions in the questionnaire.

#	Question
1	I was able to access and read the Juypter Notebook describing Digital Convolution.
2	It is not necessary to execute the notebook. You may read the text and see the results that were previously obtained. If you did attempt to run the code from Google Colab, were you successful?
3	The notebook gave me a better understanding of the application and properties of digital convolution
4	The application of digital convolution to the pharmaceutical and probability areas has stimulated my interest and curiosity on applications to other areas.
5	In addition to areas presented in this notebook, if you can think of other areas where digital convolution would be applicable, please enter them below.
6	After reviewing this application of digital convolution, I feel more connected to other disciplines and areas of study.

Table 1- Survey Questions

Survey Results

Sixteen students responded to the survey; however, only fourteen were able to open the Jupyter notebook (question 1 from the survey). The results from the fourteen who viewed the notebook are presented below. Related to question 2, four students opened the notebook and executed the code. Five students opened it but did not attempt to execute the code since it was not necessary to see the results, and five students attempted to execute the code but were unsuccessful. The survey results for questions 3, 4, and 6 are shown in Figure 4 for the students who could access the notebook. One student did not answer question 6.

Selected Answers to Question 5: What other areas are convolution applicable?

- Digital convolution could be used in image processing
- I believe digital convolution may be able to be used for signal processing as well as with signal filters.
- It can be applicable in digital image processing or image manipulation.
- You can use it for audio processing, signal filtering, artificial intelligence, optics or many other things.

Students received credit for viewing the notebook and answering the survey questions anonymously. Although it was not necessary to execute the Jupyter notebook to see the results, nine students attempted to execute the code and four did so successfully. The students were given comprehensive instruction on how to access and execute the code prior to the assignment. There are several reasons why several students may have had a problem with the code execution. For example, they may have attempted to execute the code segments out of sequence.

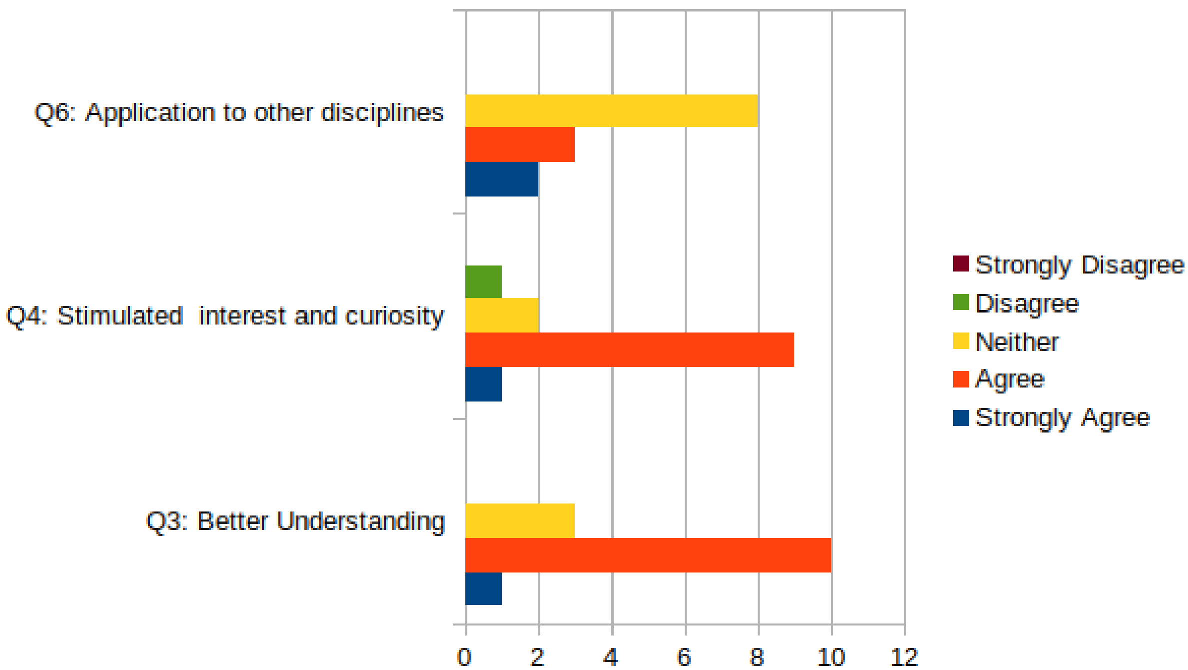


Figure 4- Student survey results for questions 3, 4, and 6.

Conclusions

An active learning environment for discrete convolution, a foundation of digital signal processing, was provided utilizing a Jupyter notebook. The discrete convolution operation and its properties were described along with two general examples, medication administration and the probability density distributions of independent random variables. Examples used in the notebook, namely medication administration and dice probability density, are not the typical signal processing textbook examples. This was done to stimulate the students' curiosity and to show that convolution is more universal and useful in many different applications. The mathematical explanation as well as visual demonstration were provided in the Jupyter notebook to allow the student to explore their understanding the convolution operator by making changes and observing the results. Visual demonstration and validation of the convolution and its properties were developed using the Python codes and graphs. The input data could be changed and executed inside the notebook without additional software or knowledge of Python. Viewing and executing the code in the notebook only required a web browser installed on the student's computer. The notebook was used by students and the survey questions were answered anonymously for credit. The results indicate that students were receptive to this learning modality to enhance their understanding of discrete convolution. The presentation of discrete-time convolution as an interactive Jupyter notebook was done to evaluate the students acceptance of this instructional modality. Based on the survey results the majority of students agreed or strongly agreed that the Jupyter notebook presentation was acceptable and facilitated their understanding of the convolutional operation in signal processing. With this established, a study can now be designed to validate the students' understanding of this and other discrete-time signal concepts compared to more traditional methods.

References

1- A.V. Oppenheim and R.W. Schafer, *Discrete-Time Signal Processing*, 3rd ed, Pearson, 2010.

2- S. W. Smith, *Digital Signal Processing*. San Diego, CA: California Technical Publishing, 1999.

3- “Welcome to Colaboratory”,[online]. Available: https://colab.research.google.com/?utm_source=scs-index [Accessed July 12, 2023].

4- A. B. Downey. “Think DSP: Digital Signal Processing in Python”. [online]. Available: https://github.com/AllenDowney/ThinkDSP. [Accessed July 12, 2023].

5- “Discrete-Time Convolution Visualization”. [online]. Available: https://colab.research.google.com/github/Domenikos/Digital_Convolution/blob/main/Digital_Convolution.ipynb [Accessed July 12, 2023].

6- Proakis, JG and Manolakis, *DG. Digital Signal Processing: Principles, Algorithms, and Applications.* 4th ed. Pearson Prentice Hall, 2007.

7- Azad, Kalid, “Intuitive Guide to Convolution, Better Explained”. [online]. Available: https://betterexplained.com/articles/intuitive-convolution/, [Accessed July 12, 2023].

Edward Carl Greco Jr.

Dr. Greco is a Professor of Electrical and Computer Engineering with research interest in biomedical signal processing. He teaches courses in digital systems, signals and systems, communications and biomedical signal processing.

Zahra Zamanipour

Zahra Zamanipour has a PhD in Electrical Engineering and has taught engineering courses for five years. She currently works on Thermoelectric generators and its applications. Her research interest is energy harvesting systems, thermoelectric generators, solar cell systems and machine learning applications in electronics.