# FOSSEE Summer Fellowship Report

On

**Mathematics using python**

Submitted by

**G Sri Harsha**
IIIT Bangalore
g.sriharsha@iiitb.org

Under the guidance of

**Prof. Prabhu Ramachandran**
Department of Aerospace Engineering
Indian Institute of Technology, Bombay

Mentors

**Mr. Purusharth Saxena**
**Ms. Sharanya Achut**

July 19, 2020

# Acknowledgment

I am overwhelmed in all gratefulness to acknowledge all those who have helped me to put my best efforts for the assigned work.

I express my deepest thanks to **Prof.Prabhu Ramachandran** for providing this wonderful opportunity to work under this special project and also for giving necessary advice and guidance, making our task easier.

I take this moment to acknowledge my project mentors **Mr. Purusharth Saxena** and **Ms. Sharanya Achut** for spending most of their valuable time with me by guiding through every step of the internship. I am very grateful to them for giving us suitable recommendations and suggestions in my work, which were extremely valuable to improve my knowledge and imagination skills.

I would also take this opportunity to thank all my fellow members for helping me in every aspect.

I thank my parents for constant support and motivation throughout the project. I consider this opportunity as one of the biggest milestones of my career.

# Contents

# Introduction

## 1.1 FOSSEE Animations and the Summer Fellowship

**FOSSEE** is a special project at **IIT-B** which aims to improve the quality of education and research. It encourages students and faculty to use **FLOSS** tools to visualize data and concepts in a better way. The FLOSS tools supported by FOSSEE are *Scilab*, *python*, *eSim*, *OpenFOAM*, *DWSIM*, *R*, *OpenModelica*. The FOSSEE project is funded by the National Mission on Education through ICT, MHRD, Government of India.

**FOSSEE-Animations** are a part of the FOSSEE project which also aims for the same. It makes seemingly complex math and science topics very approachable through animations using certain open-source toolkits. The videos themselves are also made open-source and can be contributed by anyone.

Under FOSSEE summer FELLOWSHIP I was assigned "**Series and Transformations**" as my topic. Under the assigned topic, I have worked over *Taylor Series*, *Power Series*, *Fourier Series and Transform*, *Laplace Transform*, and *Z-Transform*. For each sub-topic, I have made lecture notes which cover everything from history, motivation for the topic to various technical terms, and theorems involved in them. Although there are notes available from various sources, there aren't any visualizations provided along with them. Hence it is always difficult to explore the beauty of these topics. To deal with this, we are asked to insert appropriate animations at appropriate places to visualize the topic in a better way. For this, we used certain open-source software. For rendering animations, I have used "**Manim**". I have also used modules like "*Matplotlib*" and "*Numpy*" for visualizing and handling the data.

To store the source code for all the work done using these open-source software, we have used **Github**

# Topic for the Fellowship - Series And Transformations

A **Series** is just a summation of some set of terms of a sequence. If we sum up an infinite number of terms of a sequence we get an infinite series. There are many kinds of series such as *arithmetic series*, *geometric series*, *divergent series*, *harmonic series*, *p-series* to mention a few. In my lecture notes, I have worked on the following series:

- Taylor Series
- Power Series
- Fourier Series

Apart from the regular formulae, there always exists some hidden beauty in any given math topic, which was the reason I got well connected to the assigned topics. The beauty in these particular sets of series lies in their wide range of applications. They are used for making calculations easier, used in solving differential equations, solving integrals and have vital applications in signal processing, etc.

Following the series, we get a few limitations in solving things. Hence to solve these, we try to represent and study data in different domains. Hence **Transformations** comes into existence. Under transforms, I have worked on:

- Fourier Transform
- Laplace Transform
- Z-Transform

It also highlights its wide range of applications and also uses certain useful analogies to get basic intuition of the topics. They are used in image processing, filtering sound signals and many more.

Under each topic, I have included definition, motivation, bird's eye view, the context of the definition, applications, history and finally some further readings. To bring down the level of difficulty in understanding these topics, a total of 21 animations were included (at appropriate places) along with few useful images.

The source code to all these animations can be accessed through the below link.

**GitHub :** **https://github.com/FOSSEE/FSF-mathematics-python-code-archive/tree/master/FSF-2020/calculus/series-and-transformations**

## 2.1 Taylor Series

I have included the geometric interpretation of the Taylor series using suitable examples. Using the same, I have introduced the "**Maclaurin Series**". Initially introduced the topic using a few special functions which can't be integrated with their original form, rather can be integrated if they are represented in polynomial form.
Then I have explained all the technical terms. Following them, I have mentioned a few exceptions and hence proceeded with introducing the "*remainder term*" and basic logic involved behind it. Then proceeded with some of the theorems linked to it.
Finally mentioned a few applications related to Taylor series, followed by its history.

Number of animations included : 4

**Link to the Note : https://math.animations.fossee.in/contents/series-and-transformations/series/taylor-series**

**GitHub link to the animations : https://github.com/FOSSEE/FSF-mathematics-python-code-archive/tree/master/FSF-2020/calculus/series-and-transformations/Taylor Series**

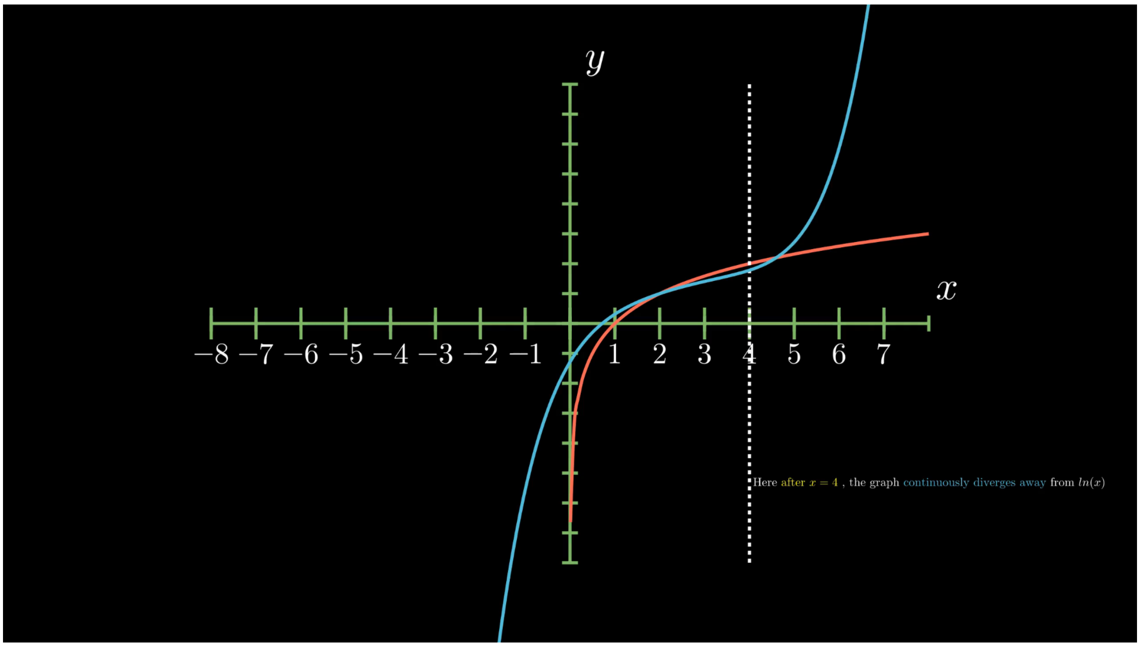


Figure 2.1: A frame of an animation depicting how a function diverges after a certain value

## 2.2 Power Series

Started with the normal definition. Then used the term "**Taylor Series**" and treated Power Series as a generalization (a superset) of Taylor Series. Then took all the technical terms used in Taylor Series and defined their general form and explained each of them with a suitable example. Then followed with a few sets of animations explaining the convergence and few other terms related to it, such as the radius of convergence, the interval of convergence along with a suitable example. Then proceeded with explaining a few theorems such as "*Cauchy Hadamard Theorem*", "*Uniqueness Theorem*" etc. These covered many other terms like "**Uniform Convergence**" etc.

Number of animations included : 4

**Link to the Note : https://math.animations.fossee.in/contents/series-and-transformations/series/power-series**

**GitHub link to the animations : https://github.com/FOSSEE/FSF-mathematics-python-code-archive/tree/master/FSF-2020/calculus/series-and-transformations/Power Series**

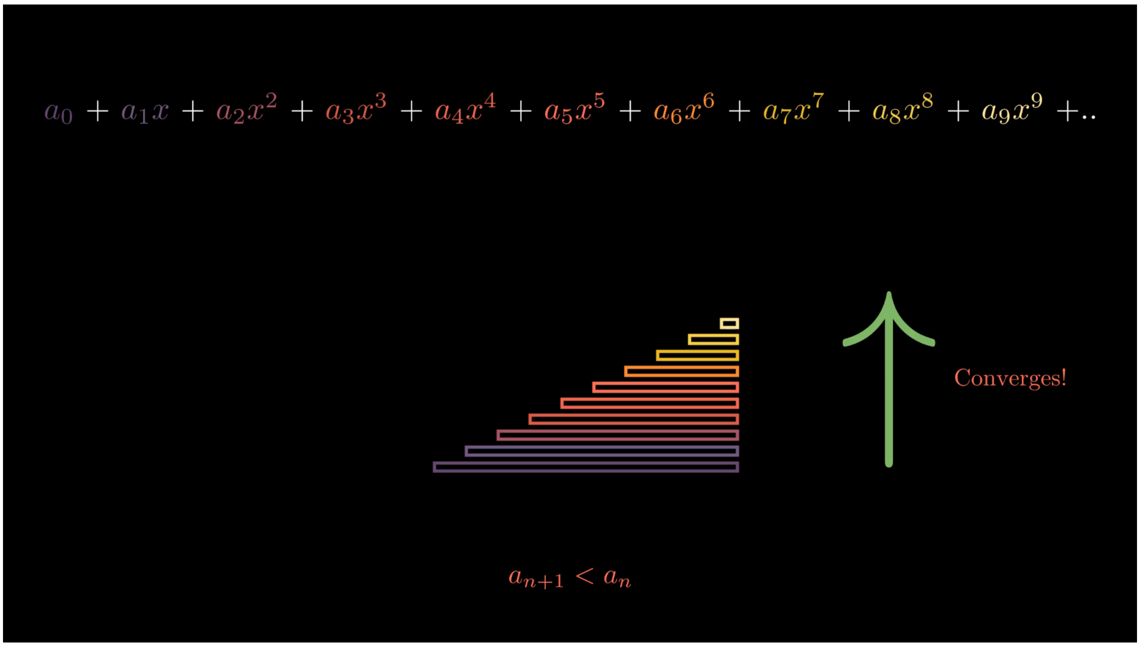


Figure 2.2: A frame of an animation depicting convergence of terms

## 2.3 Fourier Transform

Initially introduced **Fourier Series** and used a few animations to explain how a **periodic function** can be represented in terms of only sines and cosines.
As a motivation, I have used decomposing a tone into its constituent frequencies and hence representing the original wave as the sum of its constituents. As this can even be a non-periodic curve, using an animation, I have depicted how the same technique fails for non-periodic curves. For a better understanding, I have included an animation of applying Fourier Expansion on a non-periodic function to get its geometric view.
Taking these limitations as an advantage, I have introduced the concept and the logic behind changing the domain itself to represent and study the behavior of a function. Hence used it as a bridge between Series and Transformations.
Then proceeded with **Fourier Transformations** by taking the same non-periodic function used above. Then I gave an intuition on how the Fourier Series is different from Fourier transform. I have also used certain other examples, counterexamples and also used certain analogies to explain the concept.

Number of animations included : 5

**Link to the Note : https://math.animations.fossee.in/contents/series-and-transformations/transformations/fourier-transform**

**GitHub link to the animations : https://github.com/FOSSEE/FSF-mathematics-python-code-archive/tree/master/FSF-2020/calculus/series-and-transformations/Fourier Transform**

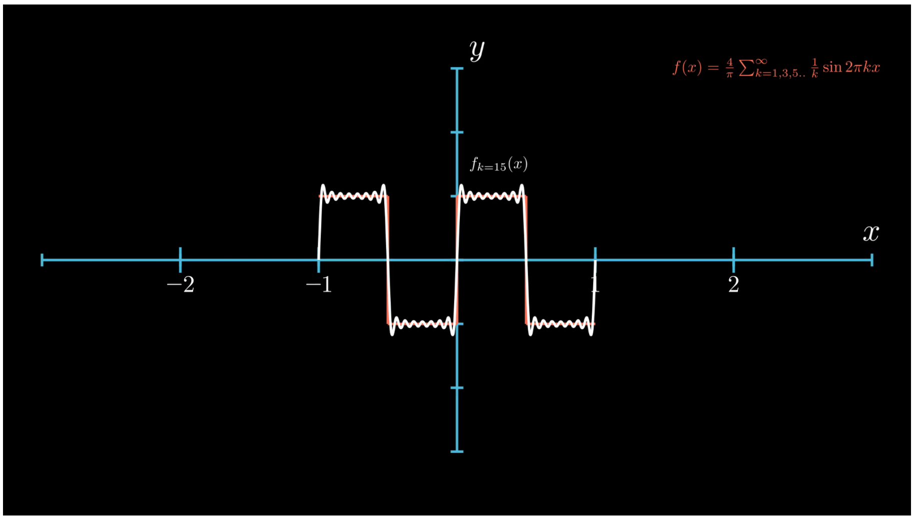


Figure 2.3: A frame of an animation depicting the Fourier expansion of a square pulse

## 2.4 Laplace Transform

Since Laplace Transform is just a simple extension of Fourier Transform, I have considered it as a machine that does a particular operation on a given function, which even changes the domain itself. Then proceeded with their application in solving differential equations. And also intuitively showed how a differential equation is solved using these as a tool. Followed by this, I gave a glance on **Inverse Laplace transform**.

Since many of the properties of Laplace transform are the same as for Fourier Transform, so mentioned the references at the appropriate places. Then introduced **Unit step function** and **Dirac delta function** and their Laplace transforms, which we encounter usually while dealing with this topic. Apart from geometrically graphing these functions, I have also included a set of animations depicting how Unit step functions are used and how a Dirac delta function is formed.

Number of animations included : 5

**Link to the Note : https://math.animations.fossee.in/contents/series-and-transformations/transformations/laplace-transform**

**GitHub link to the animations : https://github.com/FOSSEE/FSF-mathematics-python-code-archive/tree/master/FSF-2020/calculus/series-and-transformations/Laplace Transformations**

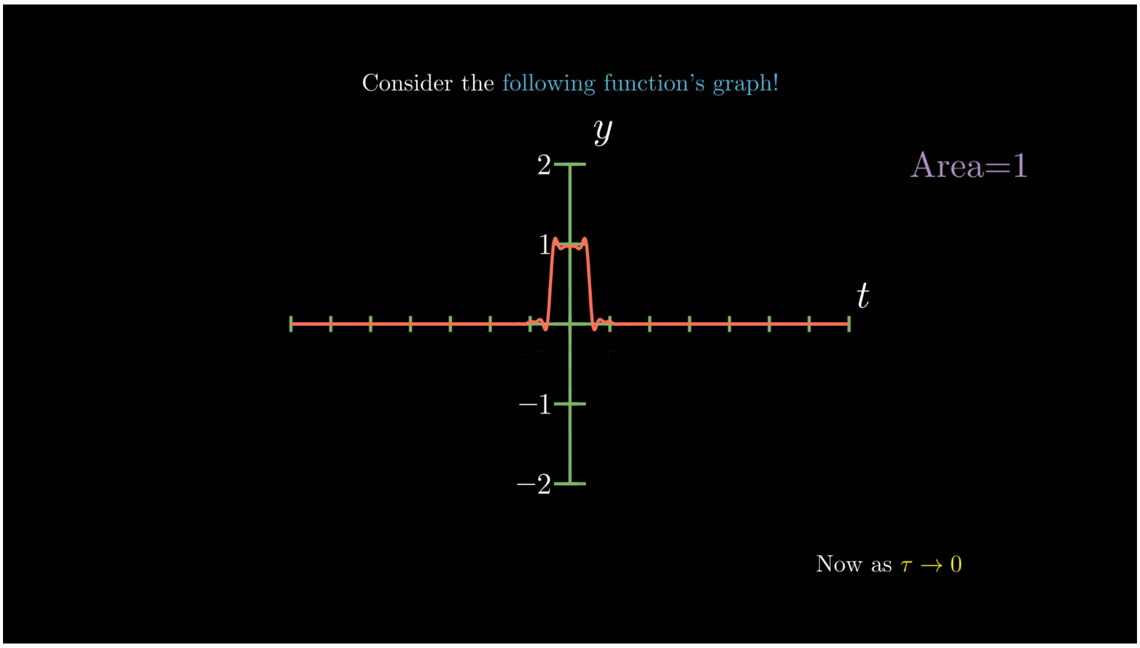


Figure 2.4: A frame of an animation depicting the formation of a Dirac Delta function

## 2.5 Z-Transform

Following the Fourier and Laplace Transforms, I have introduced Z-Transform. Since the Laplace transform doesn't deal with discrete functions, I have proceeded with the motivation of finding a method similar to the Laplace transform, which is applicable for discrete functions. Hence calling it a discrete equivalent of Laplace transforms, I have introduced few other terms such as "**sampling**" of a continuous function to get its discrete form. I have included an animation depicting how sampling works. For this, I have used copying data into a CD as an example.
I have taken certain examples and have also taken a counterexample to introduce the "**Region of convergence**" and have also included suitable animation for the same. Since I have already introduced the Delta function, I have used it as an example and found its Z-Transform geometrically. At appropriate places, I have also established the similarities between Z-Transform and Laplace Transform.

Number of animations included : 3

**Link to the Note : https://math.animations.fossee.in/contents/series-and-transformations/transformations/z-transform**

**GitHub link to the animations : https://github.com/FOSSEE/FSF-mathematics-python-code-archive/tree/master/FSF-2020/calculus/series-and-transformations/Z-Transform**

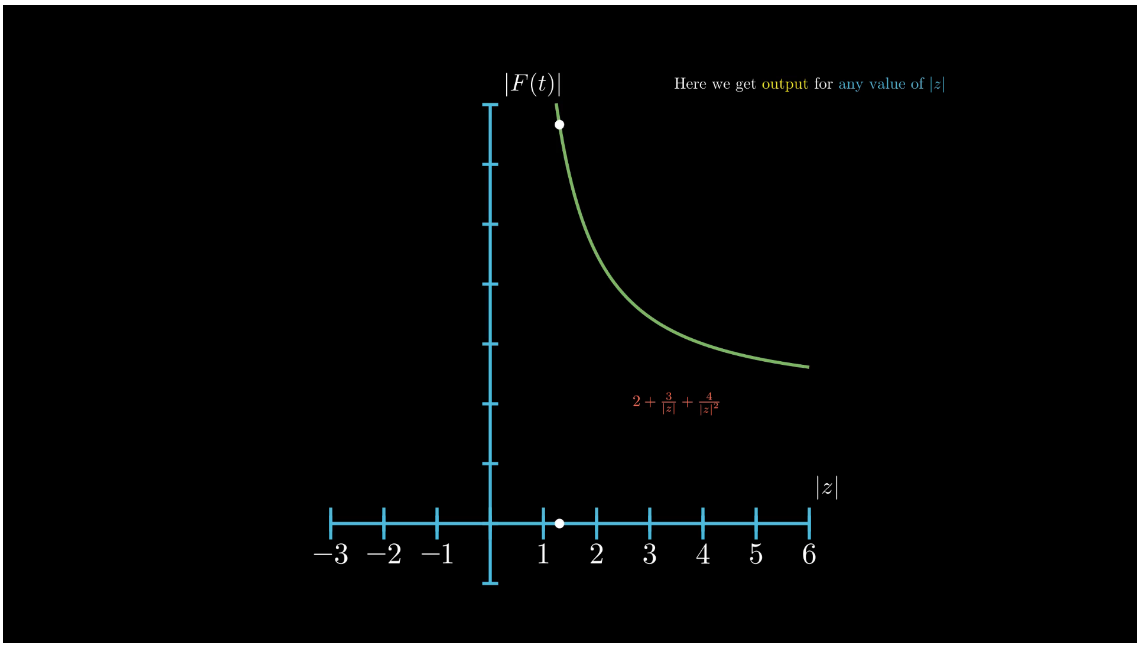


Figure 2.5: A frame of an animation showing Z-Transform of a Delta function

# Conclusion

I would consider this Fellowship as a great learning experience to work with people from various parts of the country with creative minds. There were some topics which I didn't have any idea about. Making lecture notes over those topics was one of the most challenging tasks. Apart from lecture notes, It was also challenging to curate animations in a way that everyone could understand, and also choosing appropriate examples was a bit difficult task.

The other main experience obtained from this Fellowship is to manage time. As in the given time, we need to learn the topic, make some analogies, get some intuitive ideas of explaining the topic, and finally select appropriate examples to curate the animations, It was very difficult to manage time properly and distribute it across various aspects.

Besides exploring new topics, I also learned to present and to teach a topic. I have also learned to explore the beauty behind a given math topic, rather than just taking down the formulae and solving a set of problems.

Since the Fellowship conducted this time was exceptional by making it online, It was also a bit challenging to coordinate with other fellows and mentors. Since I wasn't having any prior experience with "Manim", it initially took some time to get used to it. The video tutorials provided by the mentors were quite useful and helped me to explore the software. Though there were some challenges I faced while curating the videos, there was always constant support from the fellow people besides browsing the things.

I feel fortunate to work under this project.