# Fourier Series and its applications

Janak Subedi and Baibhav Barwal

**Project Description:**

The Fourier series is a mathematical representation of a periodic waveform as a sum of sine and cosine waves of varying amplitudes and frequencies. The Fourier series is used to decompose complex signals into simpler components that can be analyzed and understood more easily.

The Fourier transformation is a mathematical technique used to transform a time-domain signal into a frequency-domain representation. In other words, it converts a signal that changes over time into a signal that is characterized by its frequency components. The Fourier transformation is used to analyze the frequency content of a signal and to identify the individual frequency components that make up the signal.

Our project focuses on using Fourier Series in music analysis. We want to understand how periodic waveforms can be represented mathematically and analyze the frequencies they contain. We also apply the Discrete Fourier Transform to uncover patterns in musical compositions.

Firstly, we will examine how different sine waves can be combined to approximate a square wave. A square wave is a type of periodic waveform that alternates between two discrete levels. It has a specific pattern where the signal stays at one level for a certain period and then switches to the other level for an equal period. Square waves are commonly encountered in digital systems, electronics, and communication signals. The Fourier Series breaks down complex signals into simpler parts like sine and cosine waves with different amplitudes and frequencies. By using Fourier Series, we can approximate square waves by combining these simpler waves and capturing their unique frequencies.

We developed a Python code that demonstrates the approximation of a square wave using the Fourier Series. As we increase the number of sine waves used in the approximation, the resulting waveform becomes more similar to the original square wave. Adding more sine waves helps us capture the higher-frequency parts of the square wave, making the approximation more accurate.

Additionally, we have also explored the use of the Discrete Fourier Transform to analyze music signals. We developed another Python script that reads an audio file and applies the DFT to break down the signal into its frequencies. By comparing these decomposed frequencies with the original waveform, we gain insights into the structure of the music. The code takes the original signal and decomposes it into the first five Fourier coefficients (which represent the five most dominant frequencies in the audio signal). The decomposed frequencies are plotted alongside the original waveform.

In conclusion, our project has demonstrated the effectiveness of the Fourier Series in approximating square waves and the application of the Discrete Fourier Transform (DFT) in analyzing audio signals. By utilizing Fourier Series, we were able to break down complex square waveforms into a combination of sine waves, resulting in increasingly accurate approximations as we added more sine components to capture higher-frequency details. Furthermore, through the application of the DFT, we successfully decomposed Aaron's solo into the first five coefficients of the Fourier series, revealing the dominant frequencies present in the audio. These findings highlight the power of Fourier analysis in understanding the structure and frequency content of both simple waveforms and complex musical compositions, providing valuable insights into the hidden patterns and enhancing our overall knowledge in this fascinating field.