

# Convolution and Linear Time-Invariant Systems

Signal Processing Templates

November 24, 2025

## 1 Introduction

Convolution is the fundamental operation for analyzing linear time-invariant (LTI) systems. This template covers continuous and discrete convolution, impulse response characterization, and deconvolution techniques.

## 2 Mathematical Framework

### 2.1 Continuous Convolution

The convolution of two continuous signals:

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau) d\tau \quad (1)$$

### 2.2 Discrete Convolution

For discrete signals:

$$(x * h)[n] = \sum_{k=-\infty}^{\infty} x[k]h[n - k] \quad (2)$$

### 2.3 LTI System Response

Output of an LTI system with impulse response  $h[n]$ :

$$y[n] = x[n] * h[n] = \sum_{k=0}^{M-1} h[k]x[n - k] \quad (3)$$

### 2.4 Convolution Theorem

Convolution in time domain equals multiplication in frequency domain:

$$\mathcal{F}\{f * g\} = \mathcal{F}\{f\} \cdot \mathcal{F}\{g\} \quad (4)$$

## 2.5 Deconvolution

Recovering input from output using inverse filtering:

$$X(\omega) = \frac{Y(\omega)}{H(\omega)} \quad (5)$$

## 3 Environment Setup

## 4 Discrete Convolution Implementation

## 5 LTI System Analysis

## 6 Convolution Theorem Demonstration

## 7 Deconvolution

## 8 System Identification

## 9 Results Summary

### 9.1 Convolution Properties

### 9.2 LTI System Parameters

### 9.3 Deconvolution Performance

### 9.4 Statistical Summary

- Convolution theorem numerical error: ??
- Wiener deconvolution MSE: ??
- System identification error: ??

## 10 Conclusion

This template demonstrates convolution operations fundamental to signal processing. The convolution theorem enables efficient computation via FFT, while deconvolution recovers signals from observed data. Wiener filtering provides robust deconvolution in the presence of noise, significantly outperforming naive inverse filtering. System identification via least squares achieves ??