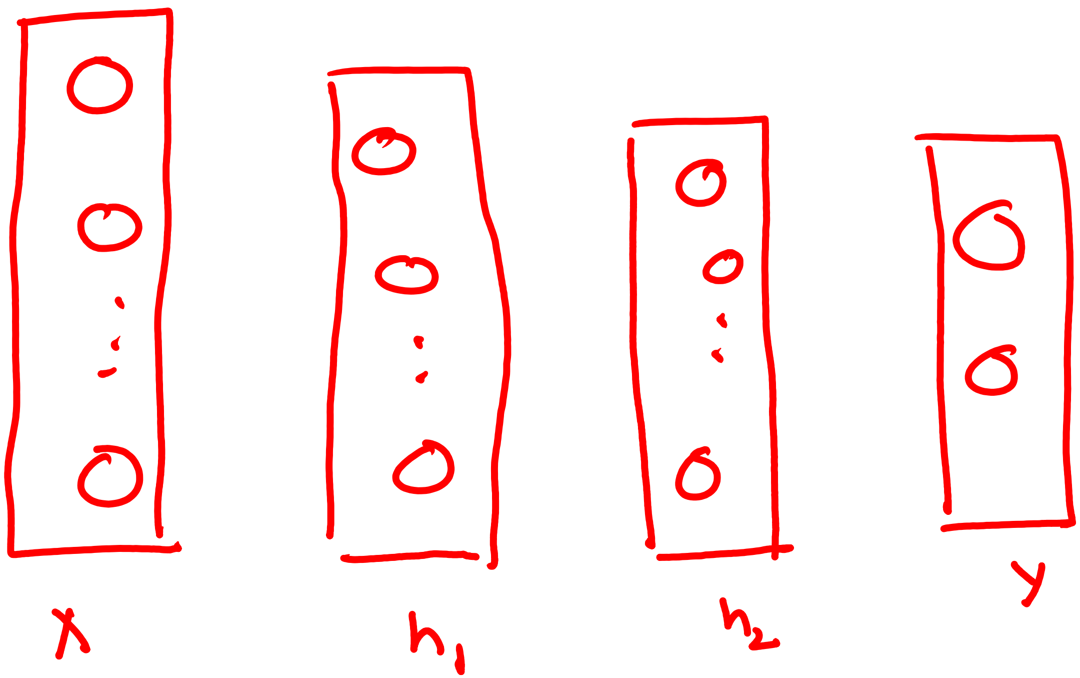
Neural operator

Dr. Souvik Chakraborty
Department of Applied Mechanics
Indian Institute of Technology Delhi
Hauz Khas – 110016, Delhi, India.

E-mail: souvik@am.iitd.ac.in
Website: https://www.csccm.in/

operator : M

$$M: f(x) \longmapsto u(x)$$



$$h_1 = \sigma_1(\omega_1^T x + b_1)$$

$$h_2 = \sigma_2(\omega_2^T h_1 + b_2)$$

$$y = \sigma_3(\omega_3^T h_2 + b_3)$$

* In OL, σ denotes activation operator

$$y = (\sigma_2 \cdot K_2 \cdot \sigma_1 \cdot K_1)\, x$$

$$-\nabla^2 \phi = \frac{\rho}{\epsilon_0} \qquad\qquad m\frac{d^2 x}{dt^2} + kx = f(t)$$

In a general form,

$$L\, \underbrace{y(x)}_{\text{variable of interest}} = \underbrace{f(x)}_{\text{forcing function}}$$

$L \leftarrow$ Linear diff. operator.

For electrostatics case,

$$L = -\nabla^2, \quad Y(x) = \phi(x)$$

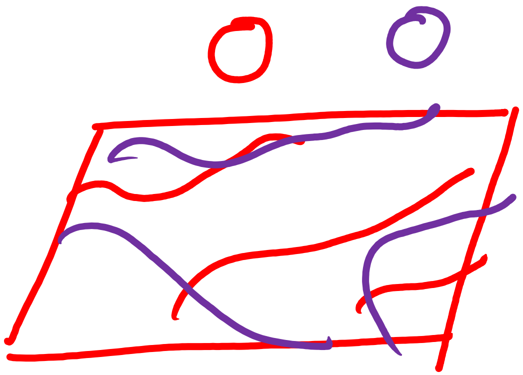
For vibration,

$$L = \left(m\frac{d^2}{dt^2} + K\right), \quad Y(t) = u(t)$$

L is a Linear Differential operator

$$L(\lambda u_1 + \mu u_2) = \lambda L(u_1) + \mu L(u_2)$$

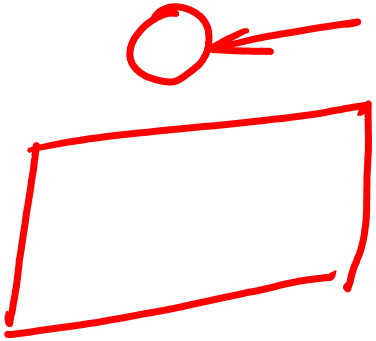
$$-\nabla^2\phi = \frac{\rho}{\epsilon_0}$$

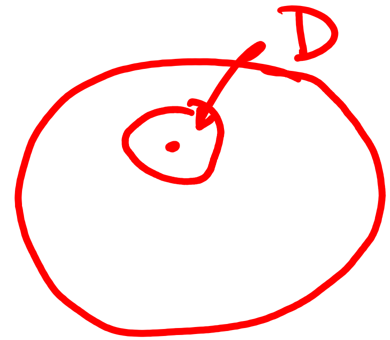


Since $L = -\nabla^2$ is a linear differential operator, we can solve of ϕ_1 corresponding to ρ_1 and ϕ_2 corresponding to ρ_2 and add it to obtain the combined effect.

* Dirac Delta function

charged body.





$\rho(x) \leftarrow$ charge density.

$$\rho(x) \approx \frac{1}{V}\int_D \rho(\xi)\, d\xi = \frac{1}{V}\int_{\mathbb{R}^3} \rho(\xi)\, I_D(\xi)\, d\xi$$

$I_D(\xi) \leftarrow$ indicator function

$$I_D(\xi) = \begin{cases} 1 & \text{if } \xi \in D \\ 0 & \text{if } \xi \notin D \end{cases}$$

$$\rho(x) \approx \int_{\mathbb{R}^3} \rho(\xi) \left(\frac{I_D(\xi)}{V} \right) d\xi$$

As $V \to 0$, $\frac{I_D(\xi)}{V} = \underbrace{\delta(x-\xi)}_{\downarrow}$

Dirac delta function

As $V \to 0$

$$\rho(x) = \lim_{V \to 0} \int \rho(\xi) \frac{I_D(\xi)}{V} \, d\xi$$

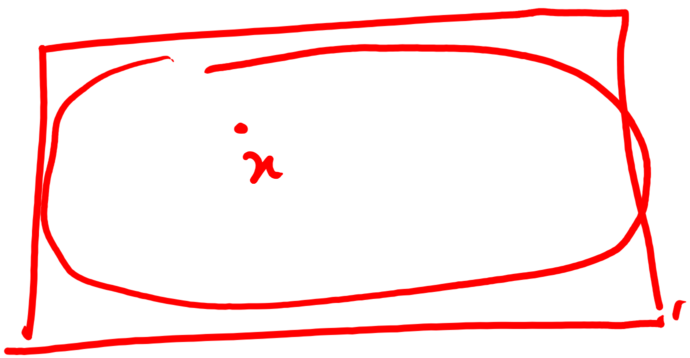
$$\rho(x) = \int \rho(\xi) \, \delta(x - \xi) \, d\xi$$

* $L \leftarrow$ linear diff. operator

$\rho(x) \leftarrow$ collection of particles.

* This means, we can solve for each particle and then do a weighted sum ← this is the fundamental behind Green's function

ξ, $\rho(\xi) =$



$$G(x;\xi)$$

$$\delta(x-\xi)$$

$$-\nabla^2 \phi = \frac{\rho}{\epsilon_0}$$

$$Ly = f$$

$$\rho(x) = \int \rho(\xi)\,\delta(x-\xi)\,d\xi$$

$$f(x) = \int f(\xi)\,\delta(x-\xi)\,d\xi$$

$$L\,\underline{\underline{G(x,\xi)}} = \underline{\underline{\delta(x-\xi)}}$$

Multiplying LHS and RHS with $f(x)$

$$f(x)\,L\,G(x,\xi) = f(x)\,\delta(x-\xi)$$

$$\Rightarrow L f(\xi) G(x,\xi) = f(\xi)\,\delta(x-\xi)$$

Integrating both sides,

$$\int_\Omega L f(\xi) G(x,\xi)\, d\xi = \int_\Omega f(\xi)\,\delta(x-\xi)\, d\xi$$

Since L is linear operator

$$L\int f(\xi) G(x,\xi)\, d\xi = f(x)$$

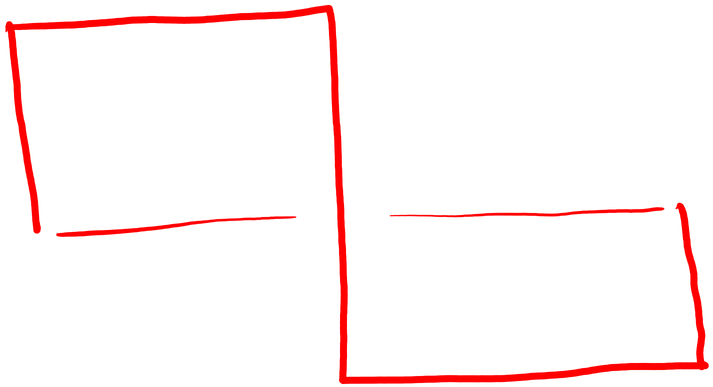
$$\therefore \; Y = \boxed{\int f(\xi)\, G(x,\xi)\, d\xi}$$

GF

$$y_0 = \sigma_N \cdot K_N \cdot \sigma_{N-1} \cdot K_{N-1} \cdot \ldots \cdot K_1 u$$

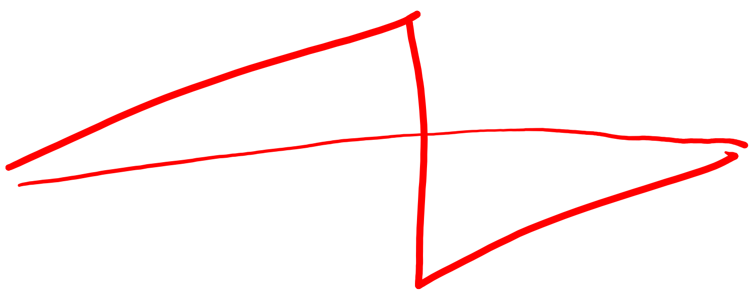
$$K = \int f(\xi)\, G(x,\xi)\, d\xi + b(x)$$

* We use integral kernel instead of local kernels.

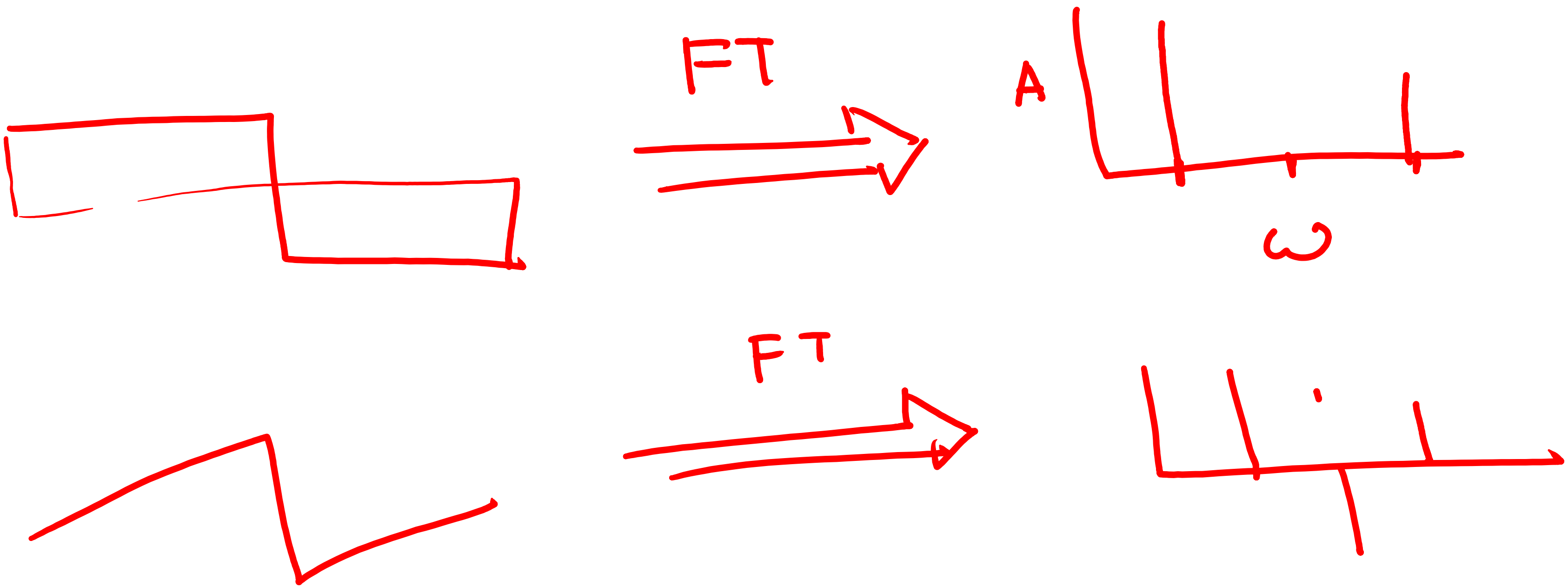


$$\sin(x) + 0\sin(2x) + 1/3\sin(3x) + \cdots$$

Sawtooth $\sin(x) - \frac{1}{2}\sin(2x) + 1/3\sin(3x) + \cdots$



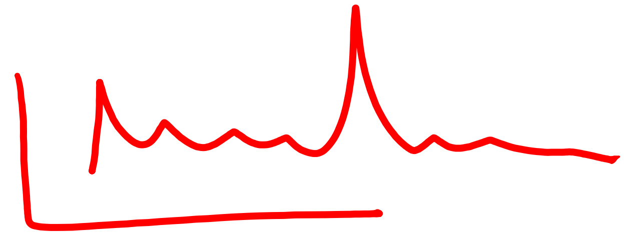
Fourier transform takes as input time series and provides frequency vs. amplitude.



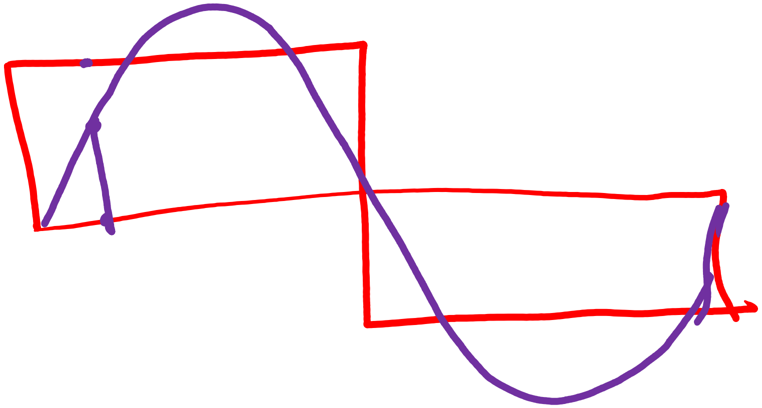
Fourier transform

$$F(\omega) = \int_{-\infty}^{\infty} f(\xi)\, e^{-i\omega\xi}\, d\xi$$

$f(t)$



$$= \int f(\;)\ \cos + \int f(\;)\ \sin$$



Operator

- An operator is defined as a mapping from a space of functions into another space of functions.

- Just like neural networks consist of linear transformations and non-linear activation functions, neural operators consist of linear operators and non-linear activation operators.

- Let v be the input vector, u be the output vector. A standard deep neural network can be written in the form:

$$u = (K_l \circ \sigma_l \circ \cdots \circ \sigma_1 \circ K_0)v$$

- where K is a linear layer or convolutional layer and σ is the activation function.

- The neural operator shares a similar framework. However, v and u are now functions with different discretizations

Green's function

- In mechanics, we deal with differential equations.

Maxwells's equation:

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\epsilon_0}$$

$$\nabla \cdot \mathbf{B} = 0$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}$$

$$\nabla \times \mathbf{B} = \mu_0 \mathbf{J} + \frac{1}{c^2}\frac{\partial \mathbf{E}}{\partial t}$$

Heat equation:

$$\frac{\partial \theta}{\partial t} = D\nabla^2\theta$$

Oscillator:

$$m\frac{d^2x}{dt^2} + \omega^2 x = F(t)$$

Green's function

- Let's consider electrostatics and forced oscillators for illustration purpose

Electrostatics:

$$-\nabla^2\phi = \frac{\rho}{\epsilon_0}$$

Oscillator:

$$m\frac{d^2x}{dt^2} + \omega^2 x = F(t)$$

- Note that both this equations can be written in a general form

$$Ly(x) = f(x)$$

- For electrostatics, $L = -\nabla^2$ and for oscillator, $L = m\frac{d^2}{dt^2} + \omega^2$

- L is a linear differential operator that maps $y(x)$ (to be solved) to $f(x)$ (forcing function).

- Green's function can deal with any function of this form.

Green's function

- Linear operator:

$$\boxed{L\big(\lambda y_1(x) + \mu y_2(x)\big) = \lambda L y_1(x) + \mu L y_2(x)}$$

- To better understand, consider,

$$-\nabla^2 \phi = \frac{\rho}{\epsilon_0}$$

$\phi \leftarrow$ electric potential, $\rho \leftarrow$ charge density (function of position),
$\epsilon_0 \leftarrow$ Electric constant

- Essentially, if you have a charged particle with charge density ρ, it generates some potential.

- An object with different charge density will produce a different potential.

- If two objects are brought together, we can simply add up the potential generated by the individual objects – this is because the operator is linear.

Dirac delta function

$$-\nabla^2\phi = \frac{\rho}{\epsilon_0}$$

$$\rho(\mathbf{x}) = \frac{charge}{volume} \approx \frac{1}{V}\int_D \rho(\xi)d\xi = \frac{1}{V}\int_{\mathbb{R}^3}\rho(\xi)\mathbf{1}_D(\xi)d\xi$$

- Here

$$\mathbf{1}_D(\xi) = \begin{cases} 1 & \text{if } \xi \in D \\ 0 & \text{elsewhere} \end{cases}$$

- Finally,

$$\rho(\mathbf{x}) \approx \int_{\mathbb{R}^3}\rho(\xi)\left(\frac{\mathbf{1}_D(\xi)}{V}\right)d\xi$$

- In the limiting condition, the above expression becomes exact. Also, $\left(\frac{\mathbf{1}_D(\xi)}{V}\right)$ tends to the dirac delta function $\delta(x-\xi)$ as volume shrinks to zero.

Dirac delta function

- Therefore,

$$\rho(\mathbf{x}) \approx \int_{\mathbb{R}^3} \rho(\xi)\delta(x-\xi)d\xi$$

- Note that here the limit and integration has been swapped. This is mathematically incorrect and is only used here for providing intuitive understanding. Dirac delta function can be more formally defined using distribution theory.

- Defining properties of delta function are as follows:

$$\delta(x-\xi) = \begin{cases} 0 & \text{if } x \neq \xi \\ +\infty & \text{if } x = \xi \end{cases}$$

$$\int \delta(x-\xi)\, d\xi = 1$$

- From definition, delta function denote charge density at point x.

Dirac delta function

- Therefore,

$$\rho(\mathbf{x}) \approx \int_{\mathbb{R}^3} \rho(\xi)\delta(x-\xi)d\xi$$

- Based on what we discussed till now, a charged object can be thought of as "continuous" sum of point charges.

- Now with this setup, we return to the electrostatics problem,

$$-\nabla^2\phi = \frac{\rho}{\epsilon_0}$$

- Remember the operator is linear and we can consider a charged object as continuous sum of point charges.

- Therefore, we can solve for each particle and then carry out a summation over it – This is the fundamental idea of Green's function.

Green's function

- The potential generated by a point charge is the rescaled Green's function.
- Green's function depends on both the location of point charge ξ and the location where we want to find the potential x.
- Therefore, we denote a function as $G(x,\xi)$ and a rescaled Green's function as $G(x,\xi)/\epsilon_0$.
- Green's function can be thought of as series of function of x, and which function to consider depends on position of the charged particle ξ.
- Therefore, the potential is defined as

$$\phi(x) = \int_{\mathbb{R}^3} \rho(\xi)\, G(x,\xi)/\epsilon_0 d\xi$$

- This means potential is a weighted sum of the Green's function.

Green's function for generalized linear differential operator

$$Ly(x) = f(x)$$

Step 1: Find Green's function:

$$LG(x,\xi) = \delta(x-\xi)$$

Step 2: Multiply both sides by $f(\xi)$

$$f(\xi)LG(x,\xi) = f(\xi)\delta(x-\xi) \Longrightarrow L\big(f(\xi)G(x,\xi)\big) = f(\xi)\delta(x-\xi)$$

Step 3: Carry out the so called continuous sum

$$\int_{\mathbb{R}^n} L\big(f(\xi)G(x,\xi)\big)d\xi = \int_{\mathbb{R}^n} f(\xi)\delta(x-\xi)d\xi$$

Note that the right hand side is precisely $f(x)$

- Since L is a linear operator and integration is a continuous sum

$$L\int_{\mathbb{R}^n} \big(f(\xi)G(x,\xi)\big)d\xi = f(x)$$

Green's function for generalized linear differential operator

$$L\int_{\mathbb{R}^n}\big(f(\xi)G(x,\xi)\big)d\xi = f(x)$$

- Therefore,

$$\int_{\mathbb{R}^n}\big(f(\xi)G(x,\xi)\big)d\xi = y(x)$$

- So, if we know $G(x,\xi)$, we can find $y(x)$.
- But how to find Green function?
- Use the properties discussed earlier.
- List of green function can be obtained by [clicking here].

- **Generalization: Hammerstein integral equation**

$$u(x) = \int_\Omega k(x,y)\,\psi(y)dy + g(x)$$

Motivation for FNO, GNO and WNO

- Just like neural networks consist of linear transformations and non-linear activation functions, neural operators consist of linear operators and non-linear activation operators.

- Let v be the input vector, u be the output vector. A standard deep neural network can be written in the form:

$$u = (K_l \circ \sigma_l \circ \cdots \circ \sigma_1 \circ K_0)v$$

where K is a linear layer or convolutional layer and σ is the activation function.

- The neural operator shares a similar framework. However, v and u are now functions.

- In neural operator, the linear transformation K is formulated using Green's function (i.e., it becomes an integral operator).

- Note that this is very similar to CNN where convolution kernel is used.

- However, instead using local kernels, we use the integral kernel which is better suited for mechanics problems.

Fourier series

- It is a mathematical tool used to study frequency analysis of signals.
- Fourier series: Any function can be represented using sine and cosine functions.
- This is an extremely powerful tool as it can provide information of the frequencies present in a signal.
- For example, consider the square wave and the sawtooth wave

$$\text{square wave: } \sin(x) + 0\sin(2x) + \frac{1}{3}\sin(3x) + \cdots$$

$$\text{sawtooth wave: } \sin(x) - \frac{1}{2}\sin(2x) + \frac{1}{3}\sin(3x) + \cdots$$

- Note that just by looking into the amplitude, it is possible to identify the pattern of the waves – this concept is often used for pattern recognition.
- The tricky part is to identify the Fourier series.

Fourier transform

- The mathematical equation for the Fourier transform is as follows:

$$F(\omega) = \int f(x)e^{-i\omega x}dx$$

- Here $f(x)$ is the time function for which we want to calculate the Fourier series (e.g., square wave or sawtooth wave).

- Note that $e^{-i\omega x} = \cos(\omega x) - i\sin(\omega x)$ from the Euler's formula.

- Therefore,

$$F(\omega) = \int f(x)(\cos(\omega x) - i\sin(\omega x))dx$$

- The above integral fundamentally calculates how correlated a sin or a cosine wave is with the given signal. In other words, it tells you whether a given frequency is actually present in a signal or not.

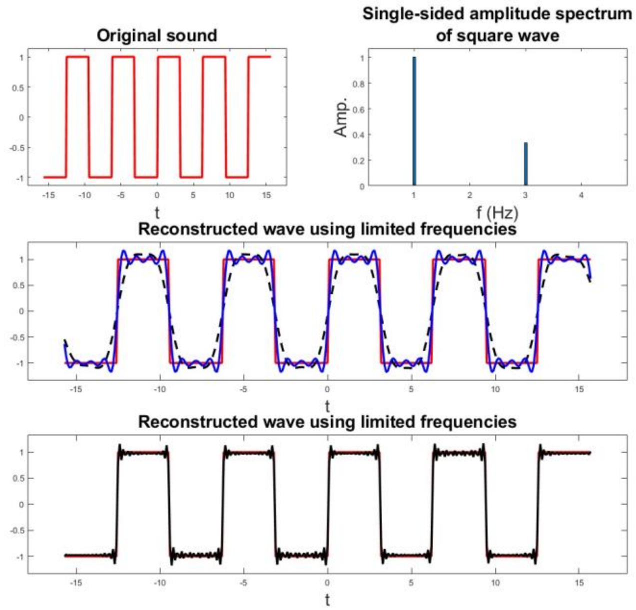
- This can also be thought of as changing the basis of the function in an infinite dimensional space.

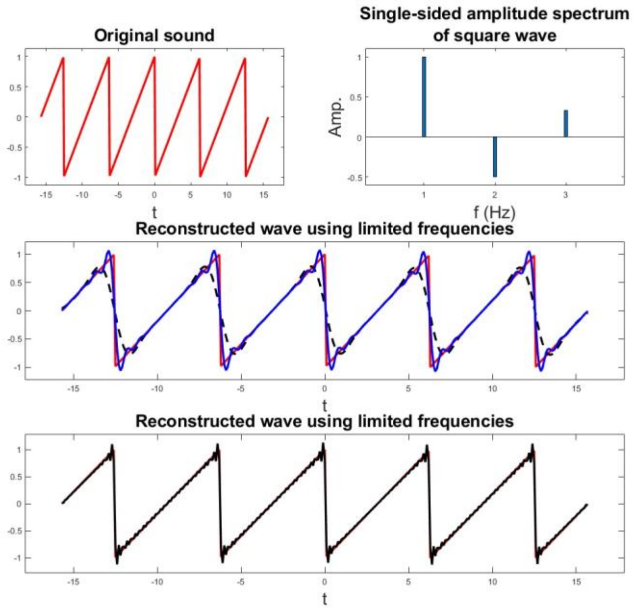
Fourier transform

- Fourier transform can be used to identify which frequencies (or which sines and cosines) are present in a signal.

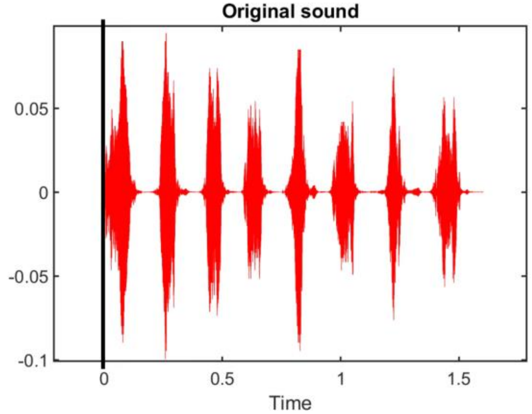
- The basic idea of Fourier transform is that we feed amplitude vs. time function and receives amplitude vs frequency as output – it fundamentally transforms from time domain to frequency domain.

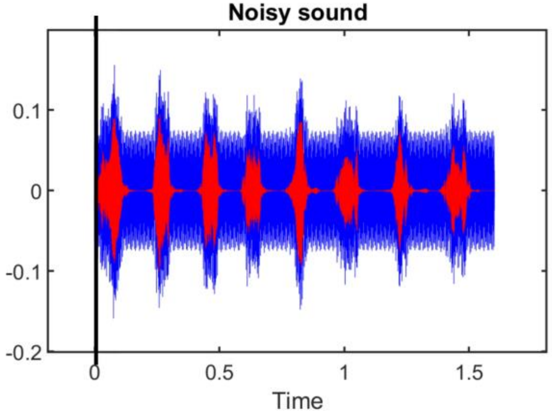
- Going back to the square and sawtooth wave, the Fourier transform will yield the following graphs.

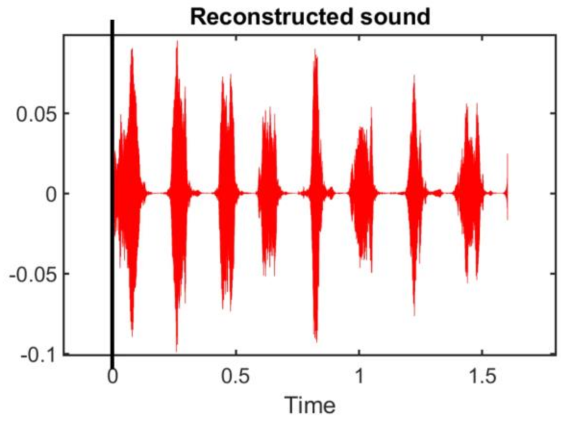


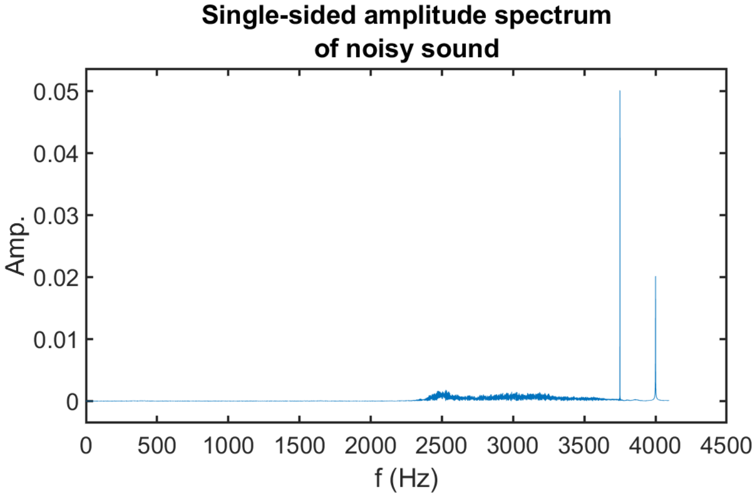


Fourier series: Application to high-pitch sound removal









[Matlab Code]()