Brain
Code
Camp

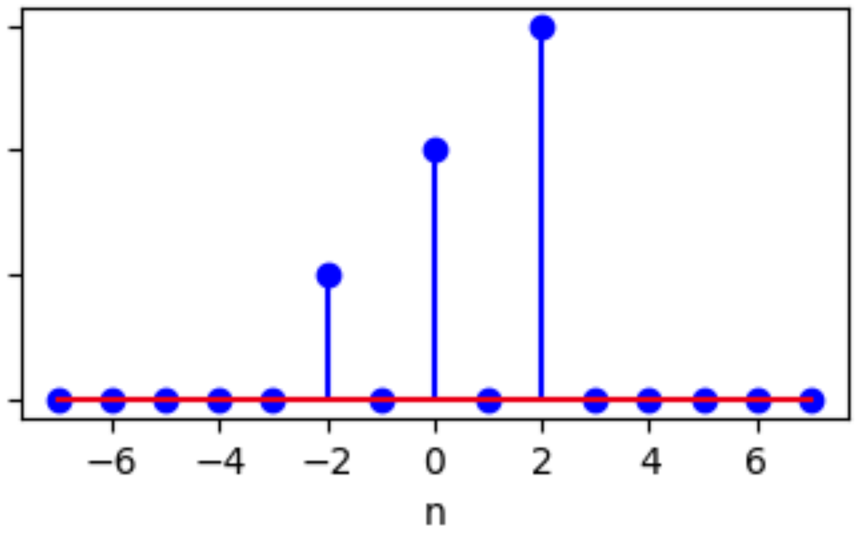
Convolution

Itthi Chatnuntawech

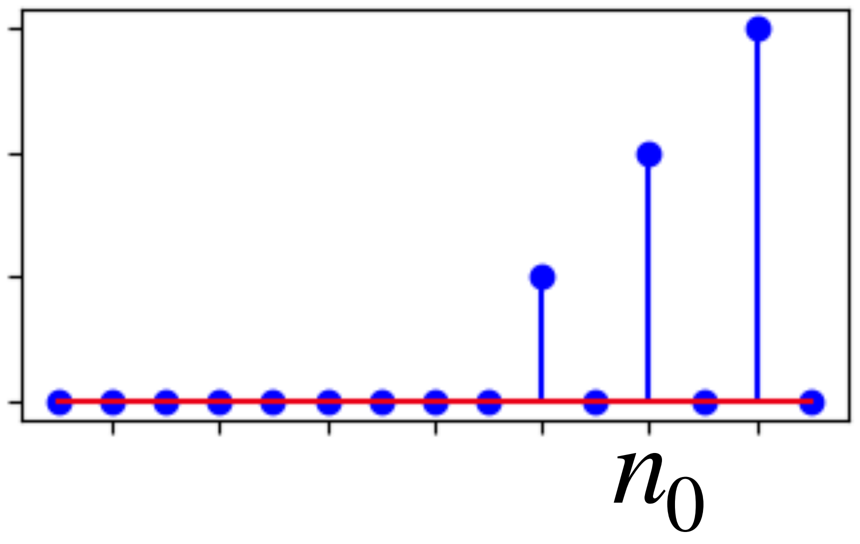
Basic Signal Transformations

Time shift

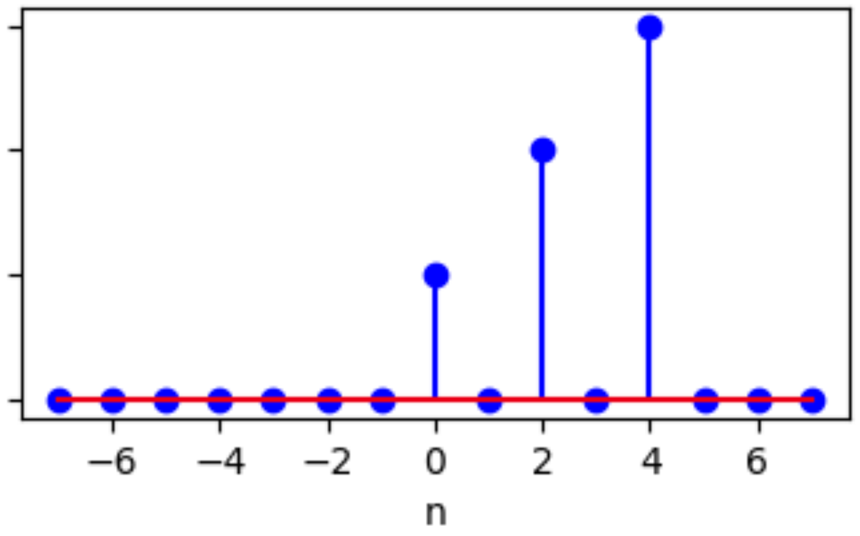
$x[n]$



$x[n - n_0]$

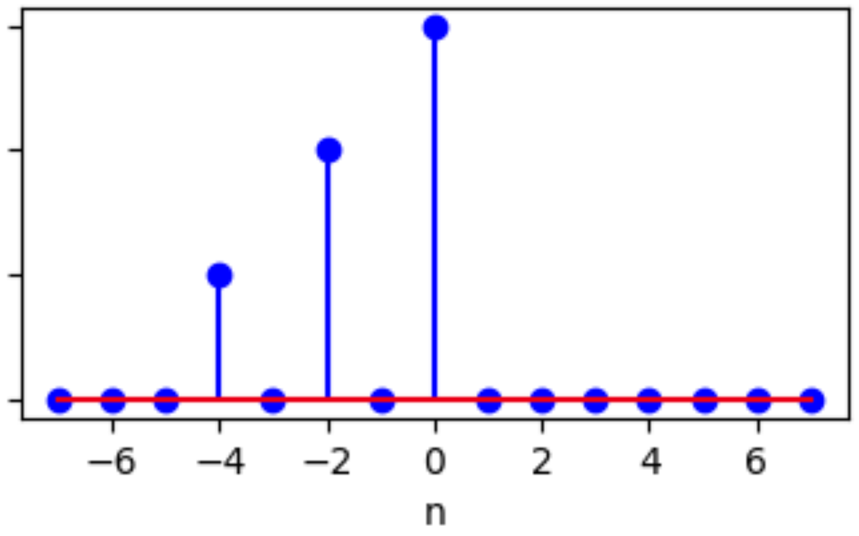


$x[n - 2]$



A delayed version of $x[n]$

$x[n + 2]$

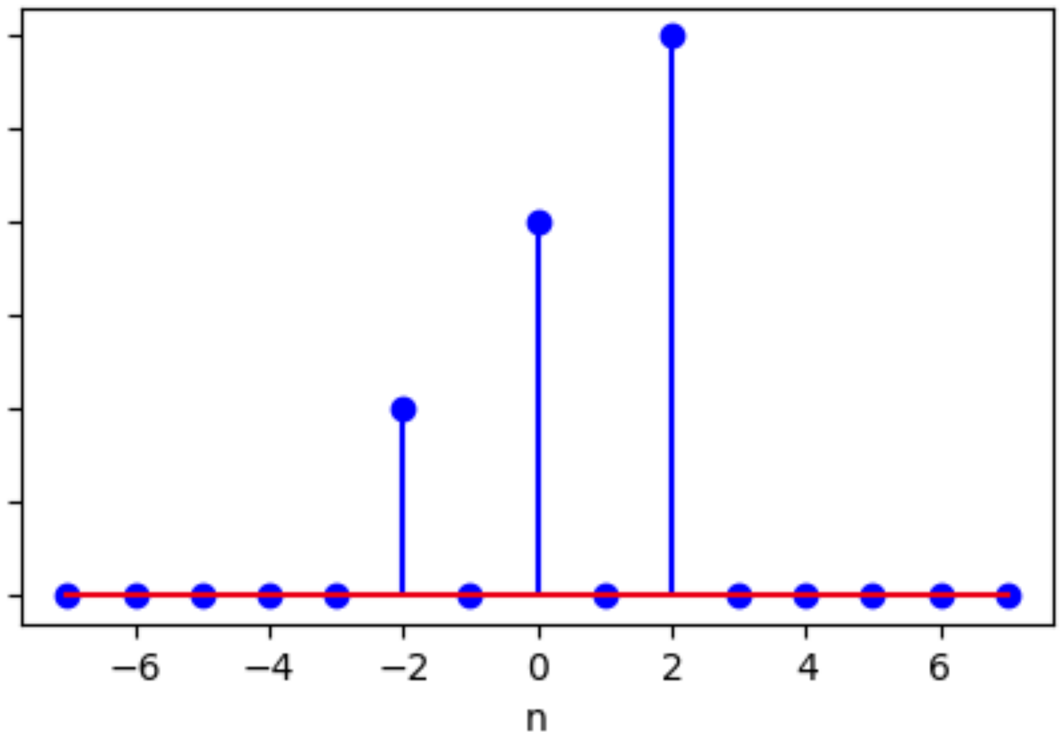


An advanced version of $x[n]$

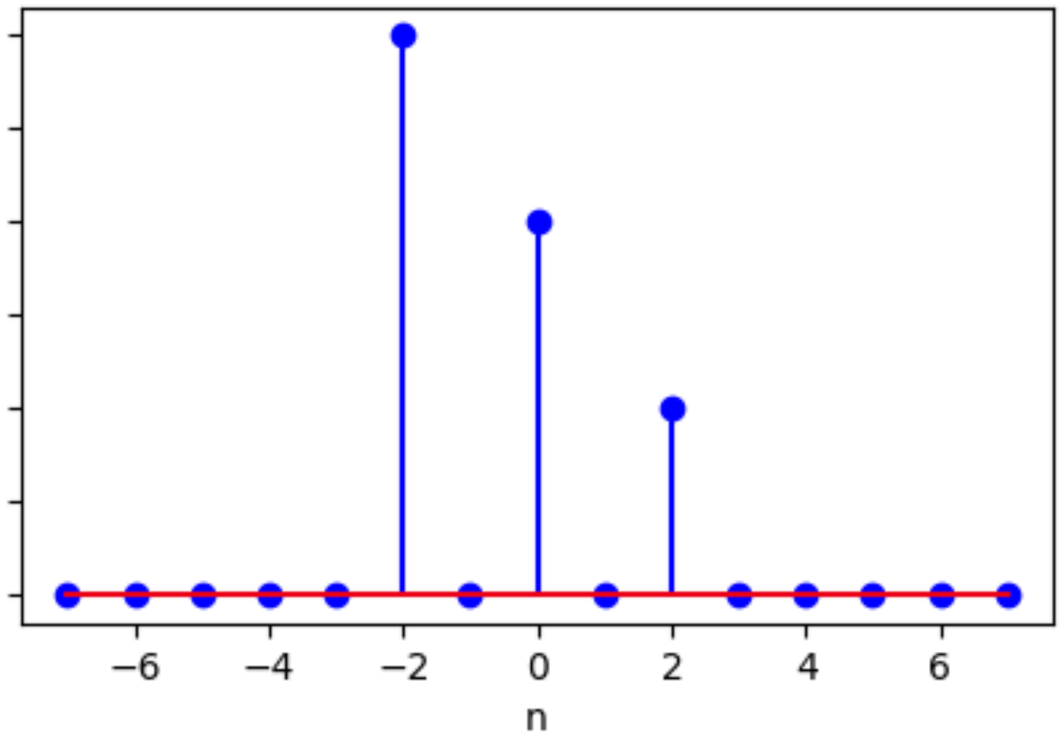
Basic Signal Transformations

Time reversal

$x[n]$



$x[-n]$



Reflection in the x-axis

Linear Time-Invariant (LTI) System

Many practical systems can be successfully modeled as LTI systems

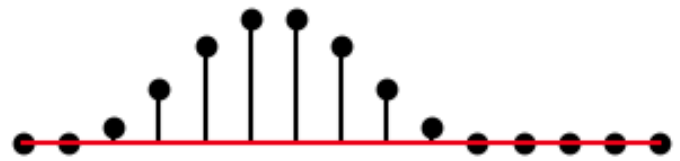
x → LTI System h → $y = x * h$

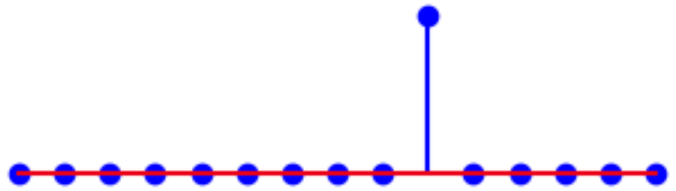
convolution

Linear + Time-Invariant

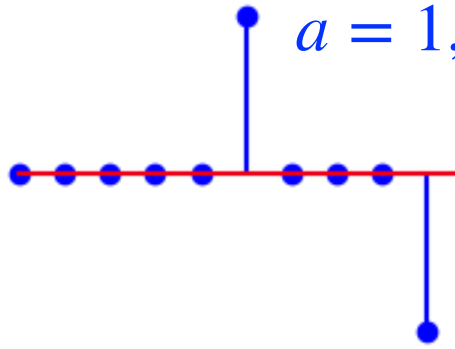
Linear System

$x_1[n]$ → Linear System A → $y_1[n]$



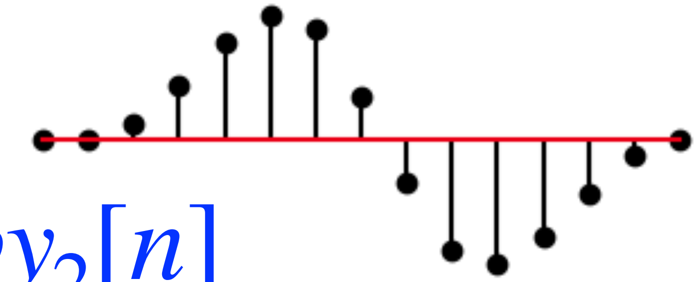


$x_2[n]$ → Linear System A → $y_2[n]$



$a = 1,\ b = -1$

$ax_1[n] + bx_2[n]$ → Linear System A → $ay_1[n] + by_2[n]$

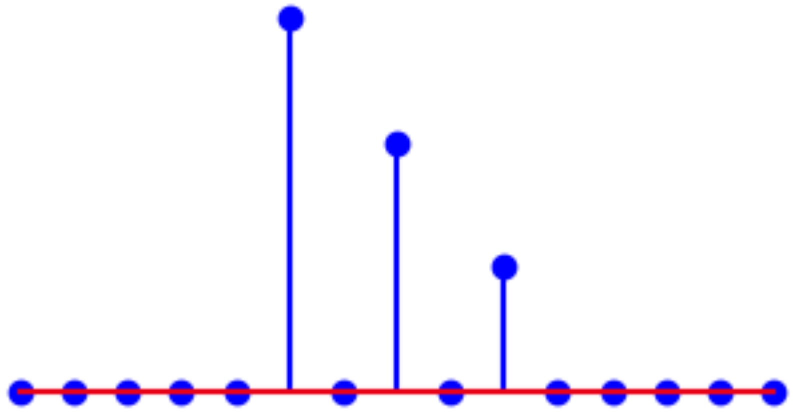


a and b are any complex constants

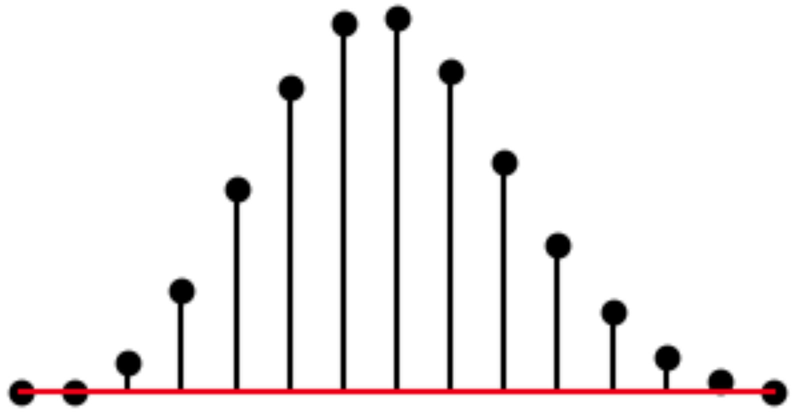
Brain Code Camp

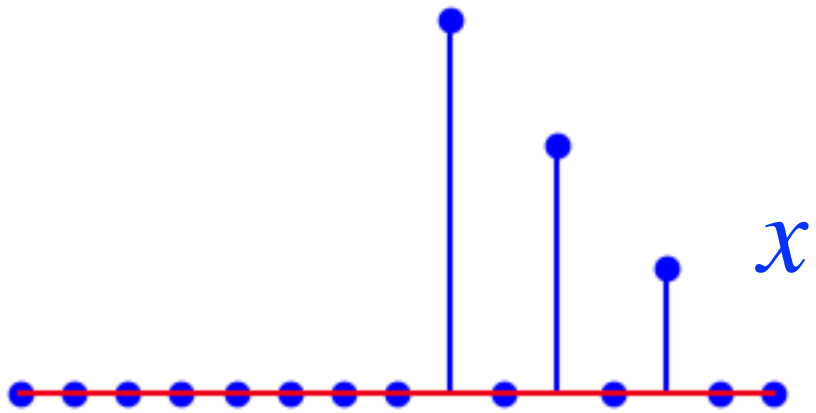
Time-Invariant System

Also referred to as a shift-invariant system

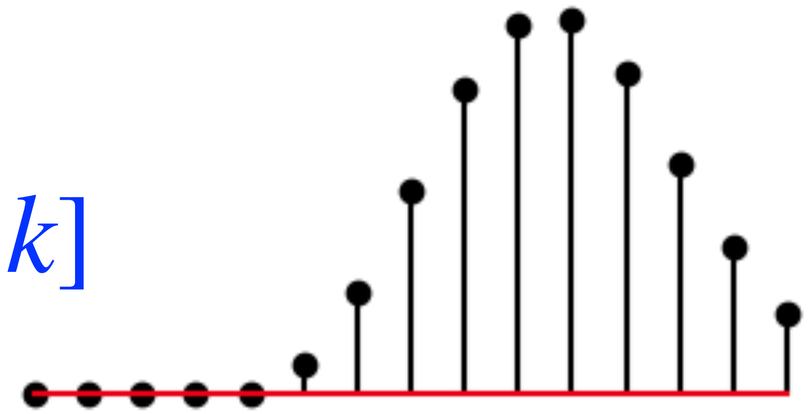


$x[n]$ → Time Invariant System B → $y[n]$





$x[n-k]$ → Time Invariant System B → $y[n-k]$



The system doesn't change over time → Would get the same results running an experiment now or later

Linear Time-Invariant (LTI) System

Many practical systems can be successfully modeled as LTI systems

$x \rightarrow \boxed{\text{LTI System } h} \rightarrow$

convolution

$$y = x * h$$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$

Oppenheim, Alan V. Discrete-time signal processing. Pearson Education India, 1999.
Oppenheim, Alan V., et al. Signals and systems. Vol. 2. Upper Saddle River, NJ: Prentice hall, 1997.

Brain Code Camp

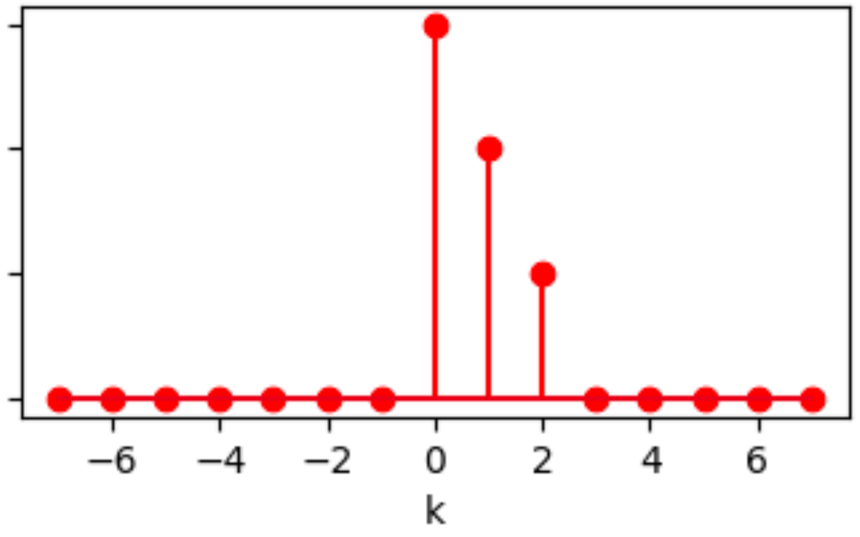
Convolution

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$

A time-reversed and shifted version of $h[k]$ (viewed as a function of k with n fixed)

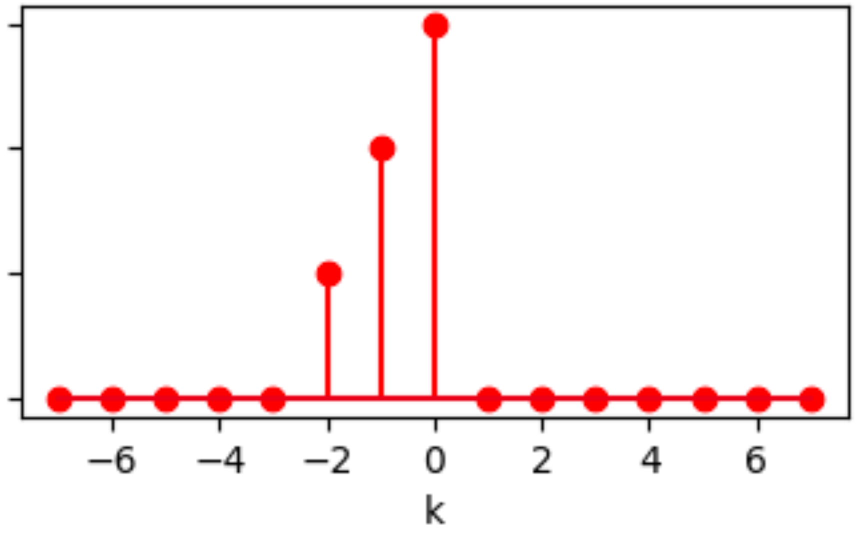
$h[-(k-n)] = h[n-k]$

$h[k]$



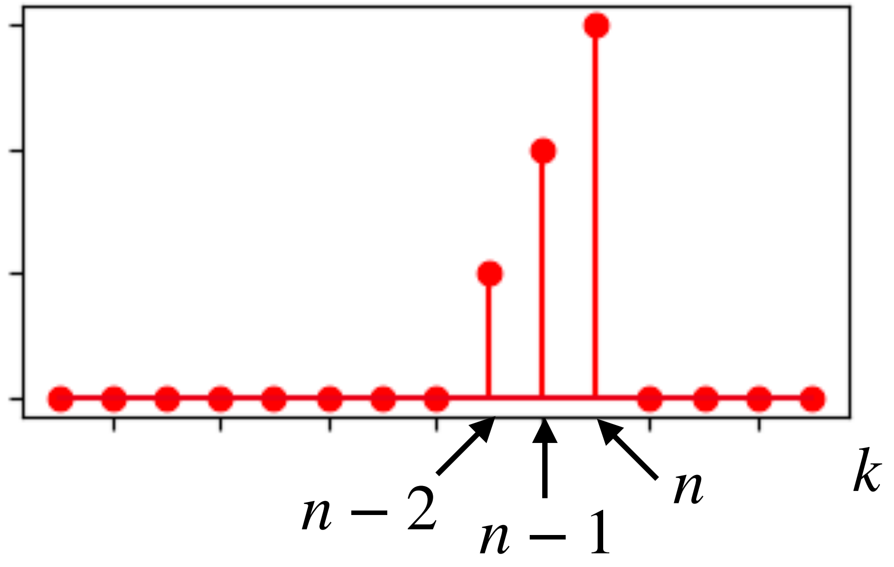
replace k by $-k$

$h[-k]$



Time-reversed version of $h[k]$

replace k by $k-n$



Time-shifted version of $h[-k]$

Convolution

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]\boxed{h[n-k]}$$

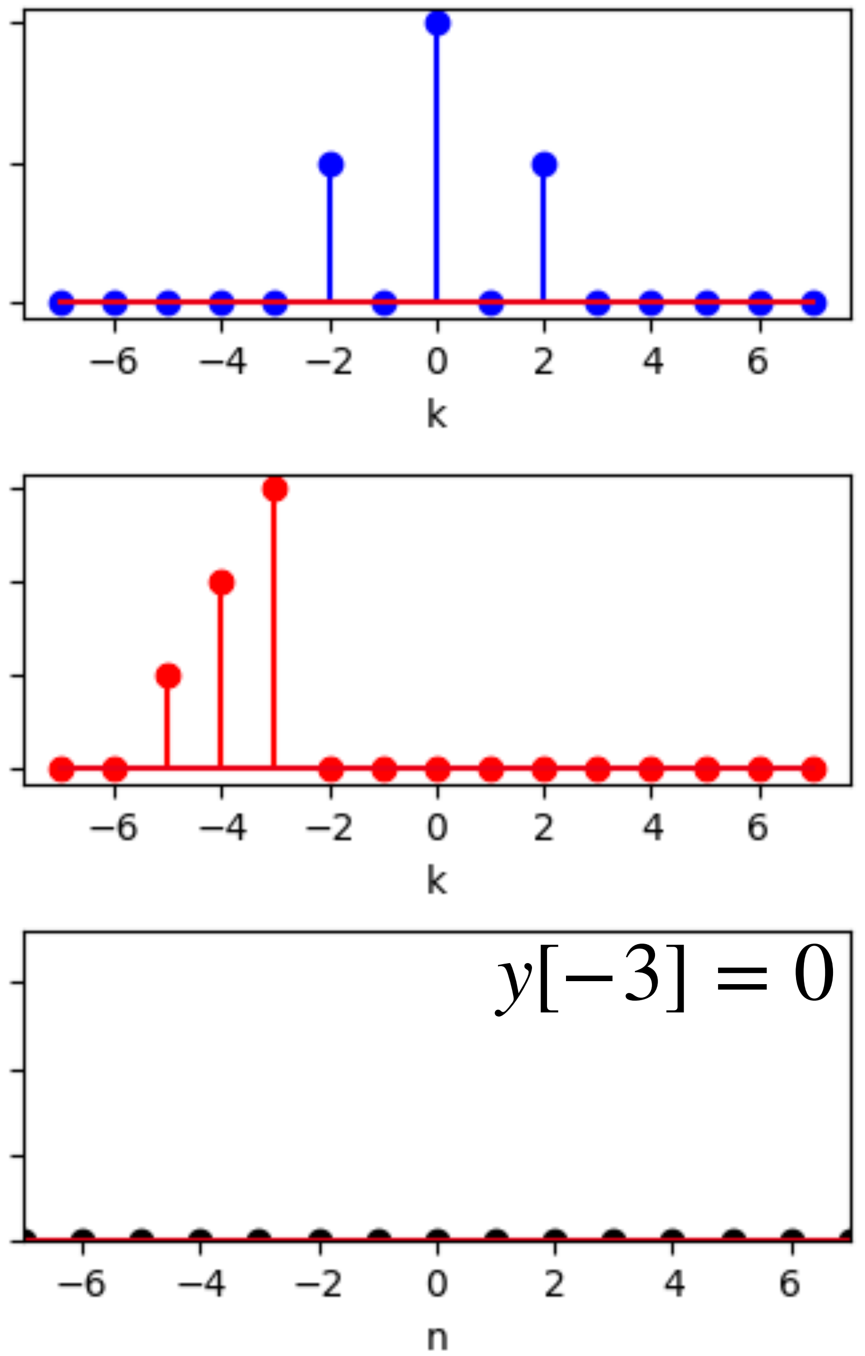
A time-reversed and shifted version of $h[k]$
(viewed as a function of k with n fixed)

$n = -3$

$x[k]$

$h[-3-k]$

$y[n]$



index =-2 | $x[k]$ | index =2

1	0	2	0	1

$h[-3-k]$

1	2	3

$y[n]$

0								

Brain Code Camp

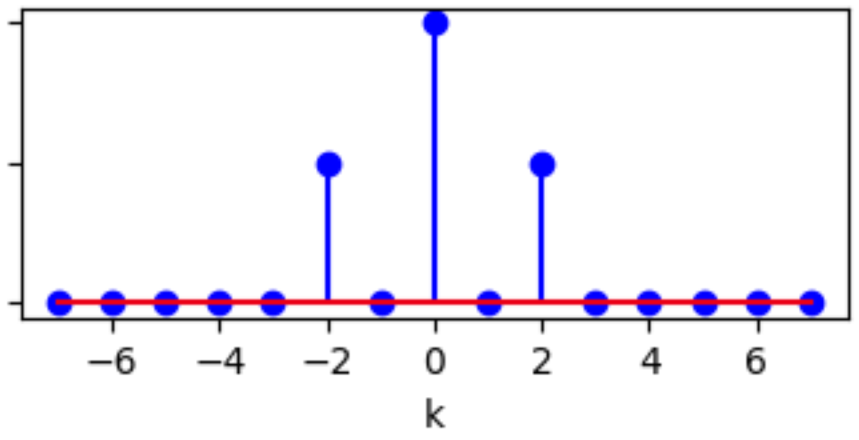
Convolution

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$

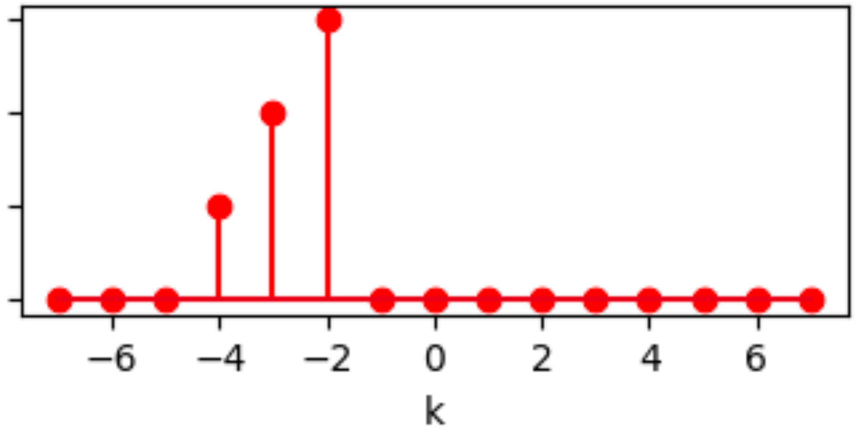
A time-reversed and shifted version of $h[k]$ (viewed as a function of k with n fixed)

$n = -2$

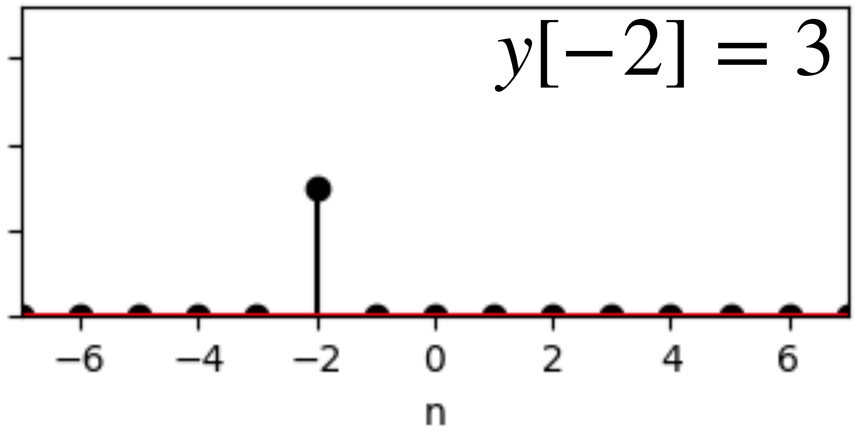
$x[k]$



$h[-2-k]$



$y[n]$



index =-2 | $x[k]$ | index =2

1	0	2	0	1

$h[-2-k]$

1	2	3

$y[n]$

0	3							

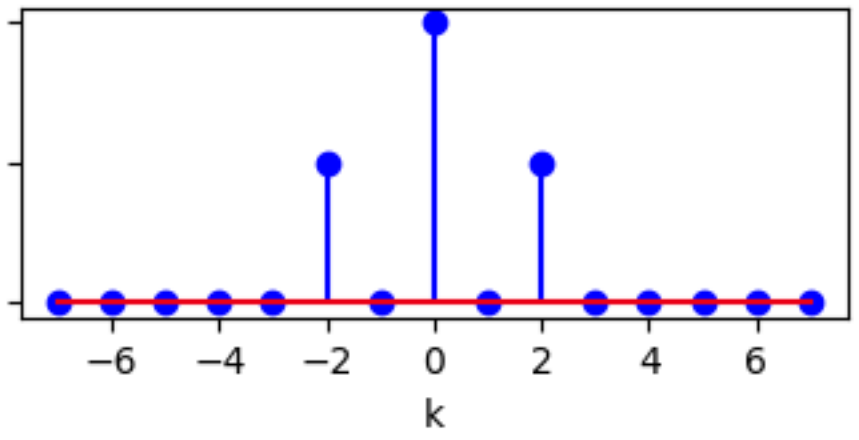
Convolution

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$

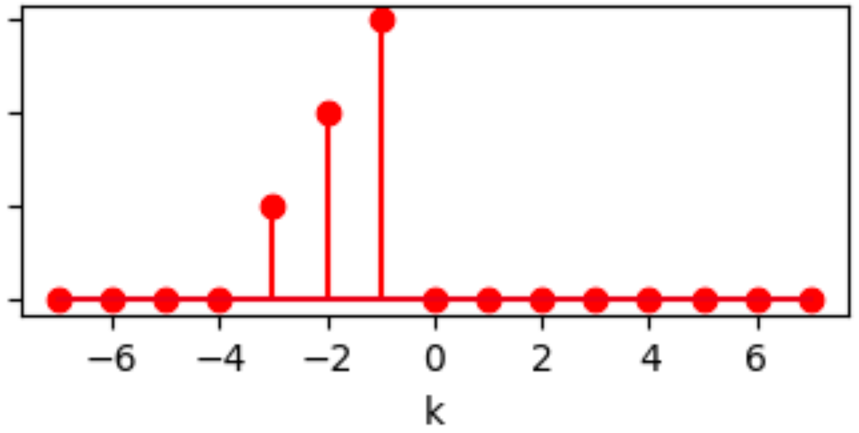
A time-reversed and shifted version of $h[k]$ (viewed as a function of k with n fixed)

$n = -1$

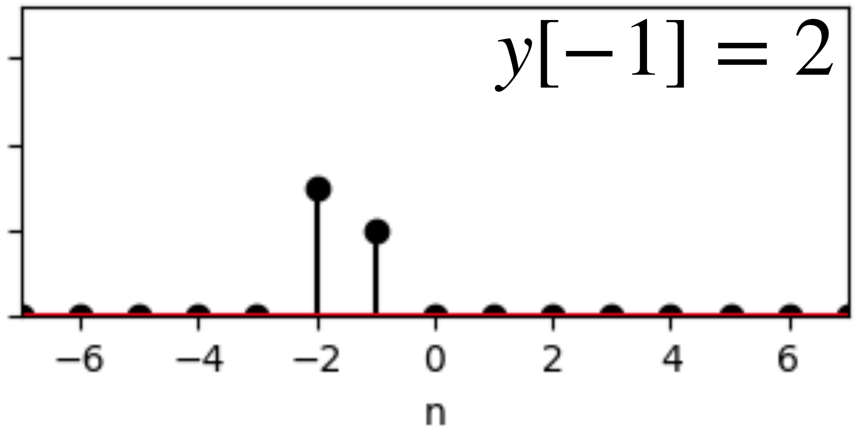
$x[k]$



$h[-1-k]$



$y[n]$



index =-2 $x[k]$ index =2

1	0	2	0	1

$h[-1-k]$

1	2	3

$y[n]$

0	3	2						

Brain Code Camp

Convolution

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]\boxed{h[n-k]}$$

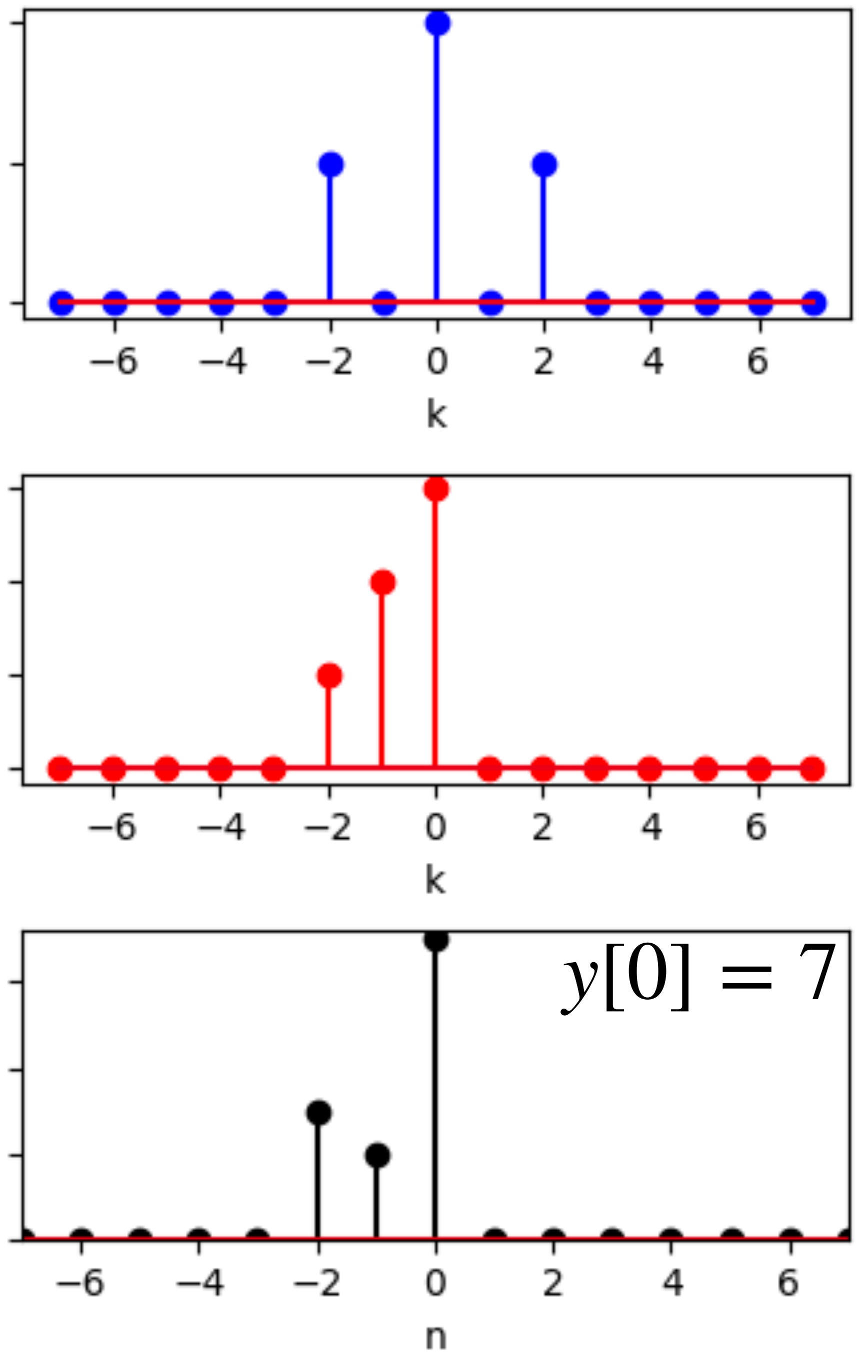
A time-reversed and shifted version of $h[k]$ (viewed as a function of k with n fixed)

$n = 0$

$x[k]$

$h[0-k]$

$y[n]$



index =-2 | $x[k]$ | index =2

1	0	2	0	1

$h[0-k]$

1	2	3

$y[n]$

0	3	2	7					

Convolution

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]\boxed{h[n-k]}$$

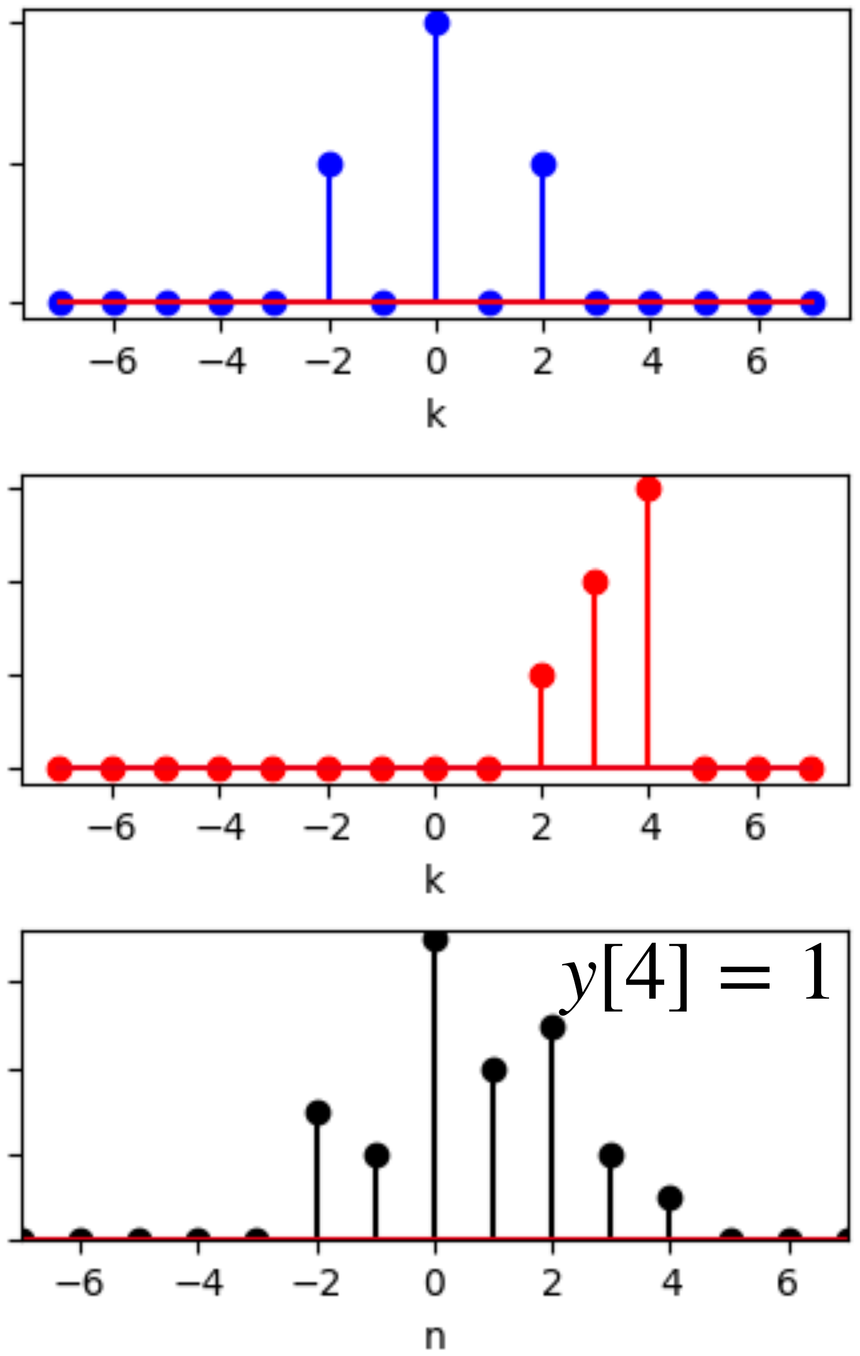
A time-reversed and shifted version of $h[k]$
(viewed as a function of k with n fixed)

$n = 4$

$x[k]$

$h[4-k]$

$y[n]$



index =-2 | $x[k]$ | index =2

1	0	2	0	1

$h[4-k]$

1	2	3

$y[n]$

0	3	2	7	4	5	2	1	

Brain Code Camp

Convolution

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]\boxed{h[n-k]}$$

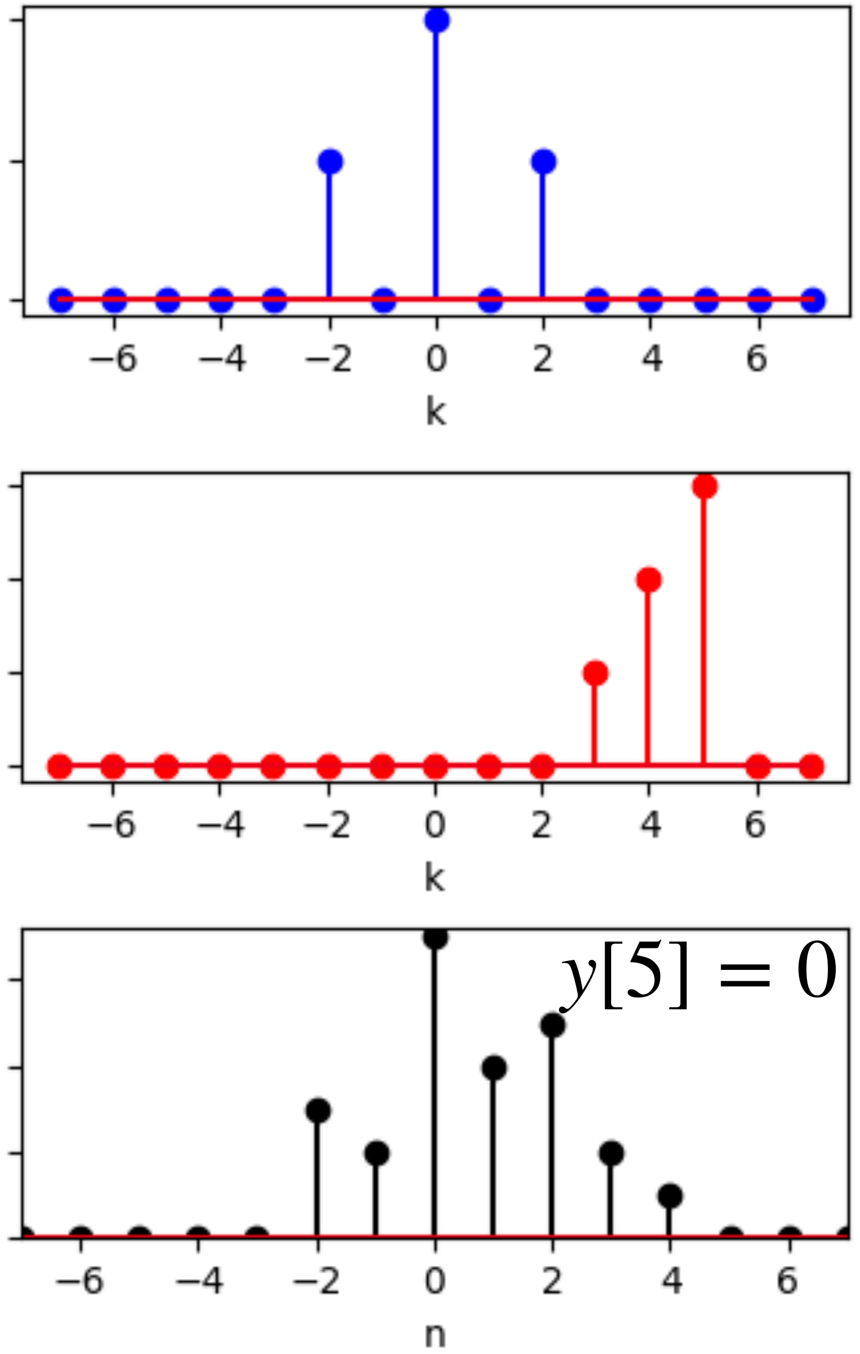
A time-reversed and shifted version of $h[k]$ (viewed as a function of k with n fixed)

$n = 5$

$x[k]$

$h[5-k]$

$y[n]$



index =-2 $x[k]$ index =2

1	0	2	0	1

$h[5-k]$

1	2	3

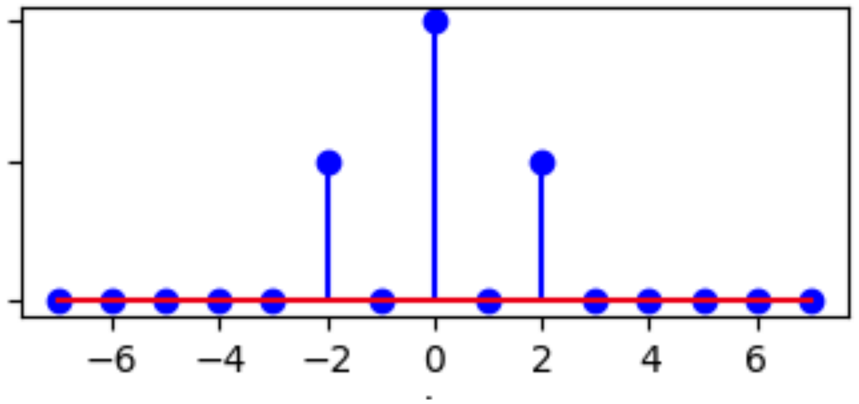
$y[n]$

0	3	2	7	4	5	2	1	0

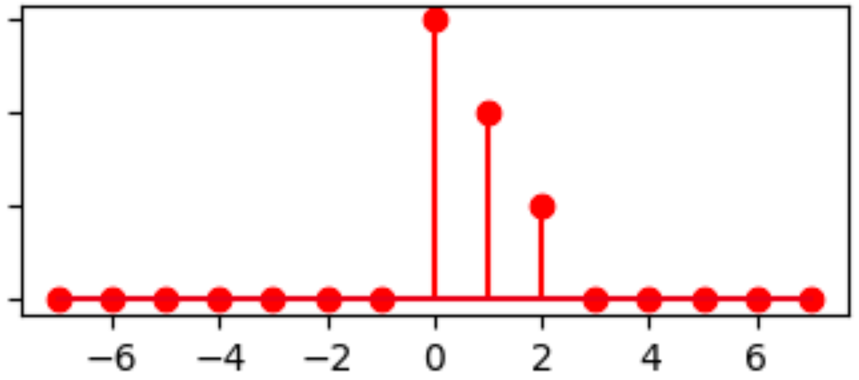
Convolution

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$

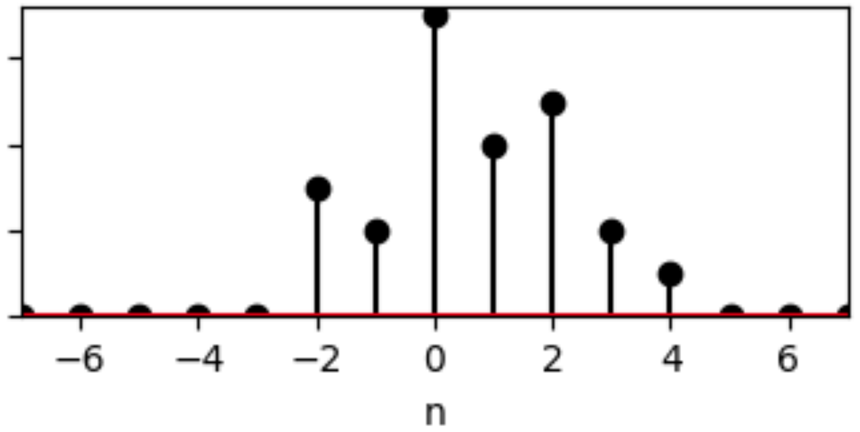
$x[n]$



$h[n]$



$y[n]$



$x[n]$

1	0	2	0	1

$h[n]$

3	2	1

$y[n]$

3	2	7	4	5	2	1

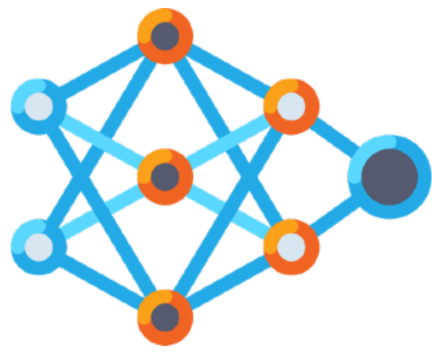
length of y = (length of x) + (length of h) -1

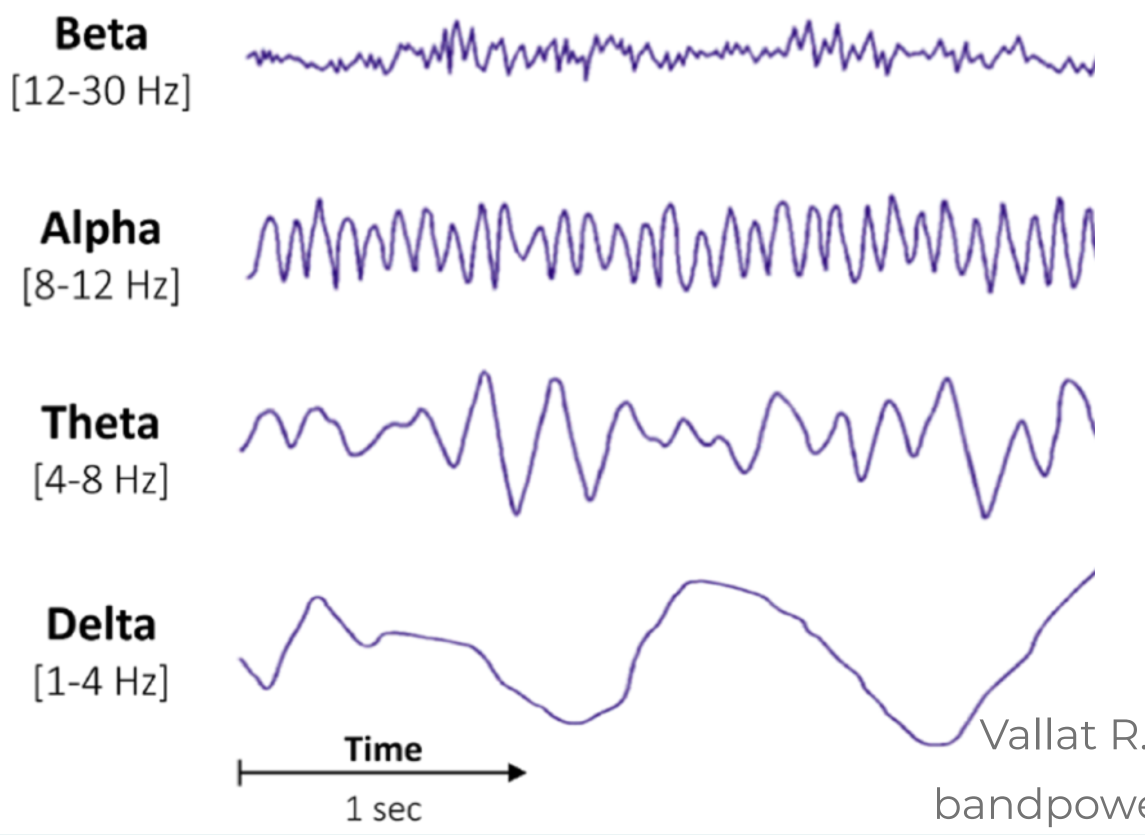
Applications of Convolution

Time series analysis - stock market averages

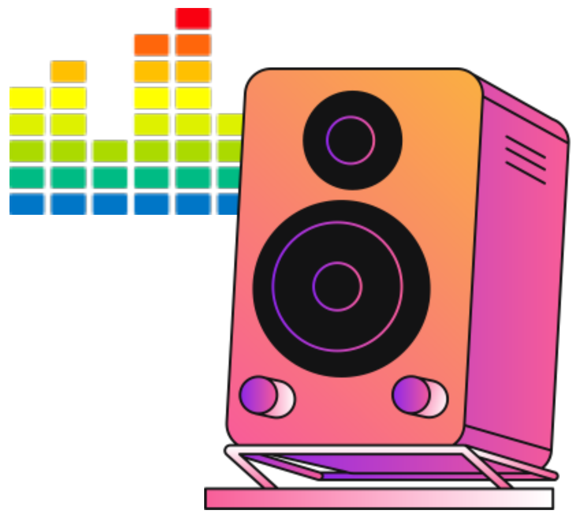
Communication systems - Amplitude Modulation (AM)

Data representation and feature extraction





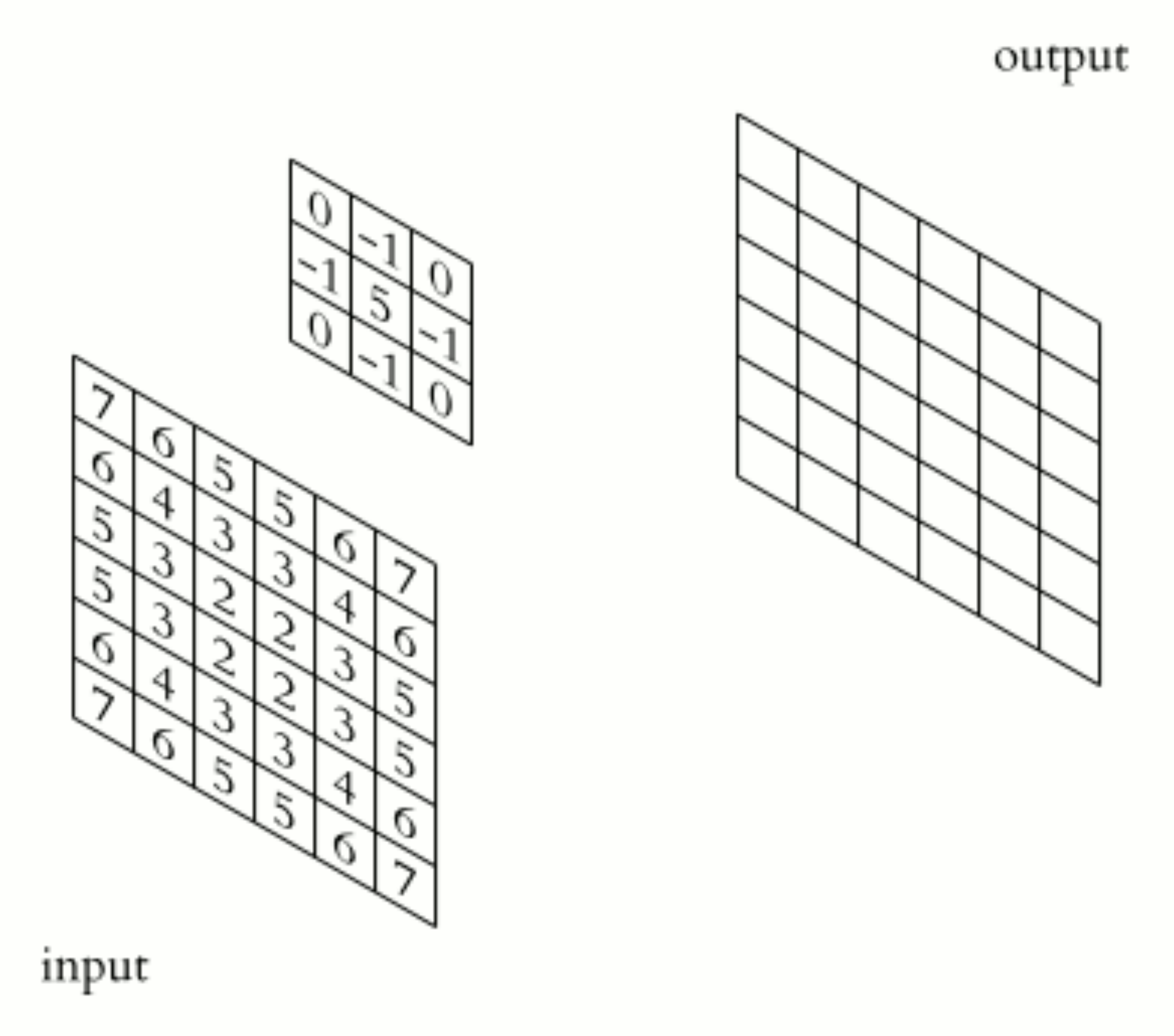
An equalizing filter compensates for the frequency characteristics of speakers



Oppenheim, Alan V., et al. Signals and systems. Vol. 2. Upper Saddle River, NJ: Prentice hall, 1997.

Vallat R. Compute the average bandpower of an EEG signal (2018)

2D Convolution

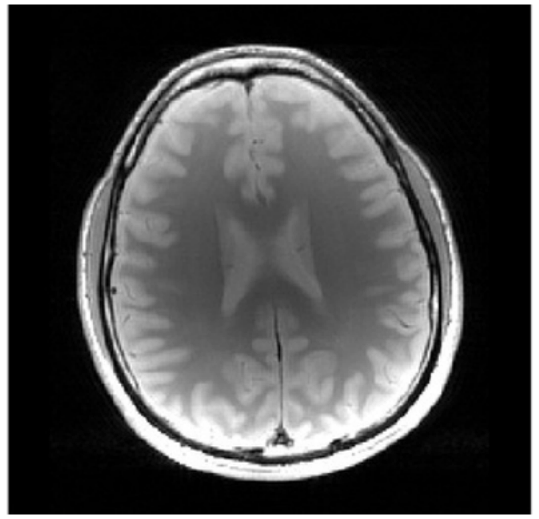


https://commons.wikimedia.org/wiki/File:2D_Convolution_Animation.gif

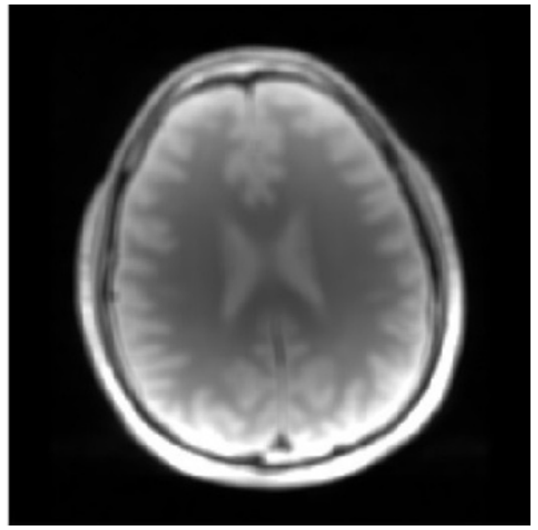
2D Convolution

Original image	Blurring	Highlighting large differences	Vertical edge detection

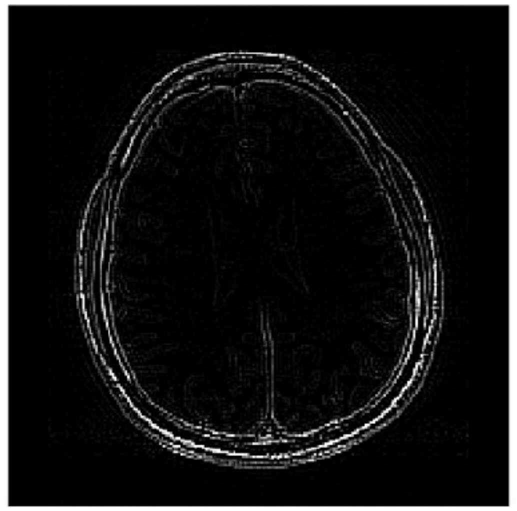
Original image



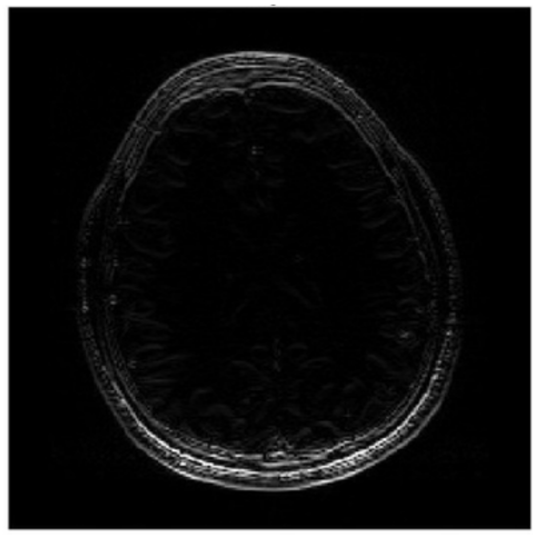
Blurring



Highlighting large differences



Vertical edge detection



Filters/ kernels

Blurring:

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Highlighting large differences:

−1	−1	−1
−1	8	−1
−1	−1	−1

Vertical edge detection:

1	1
-1	-1

Brain Code Camp

2D Convolution

Predefined filters/kernels

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Learnable filters/kernels