Outline

Lecture Contents

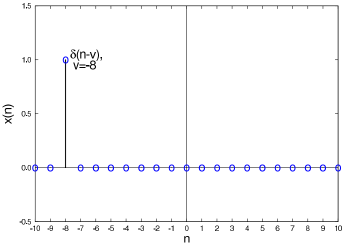
Outline

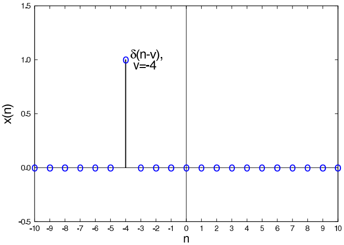
Unit Impulse Function

- Unit impulse function is a fundamental function in Digital Signal Processing (DSP)
- Symbol of Unit impulse function is the *Greek* delta: δ
- $\delta(n) = 1$ if $n = 0$, so that,

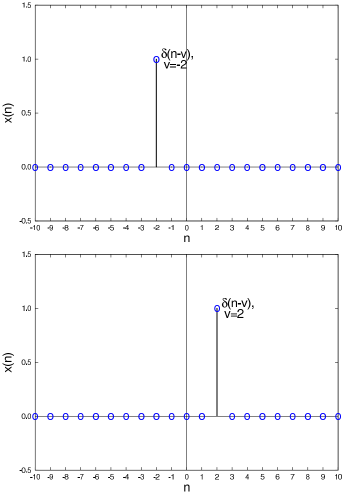
$$\delta(n-v) = \begin{cases} 1 & \text{if} \quad (n-v) = 0, \\ 0 & \quad\quad \text{otherwise.} \end{cases}$$

- *Examples*





Unit Impulse Function *examples cont'd.*



Scaling Unit Impulse Function

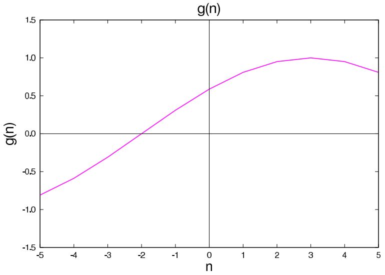
- Can scale unit implpulse with any value, *i.e.*

$$g \times \delta(n-v) = \begin{cases} g & \text{if} \quad n-v=0, \\ 0 & \quad\;\; \text{otherwise.} \end{cases}$$

- So if g is a function, such as $g(n)$ then

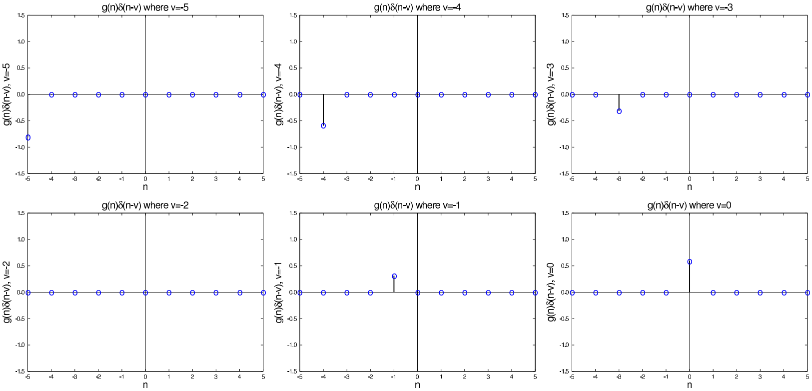
$$g(n)\delta(n-v) = \begin{cases} g(n) & \text{if} \quad n-v=0, \\ 0 & \quad\;\; \text{otherwise.} \end{cases}$$

- This is useful for something called *sifting*
- Given a signal $g(n)$:

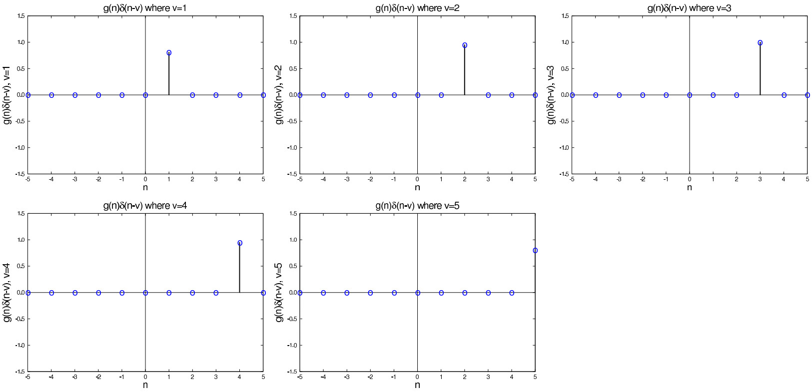


Sifting

- Calculate $g(n)\delta(n-v)$ for all values of v, *i.e.*



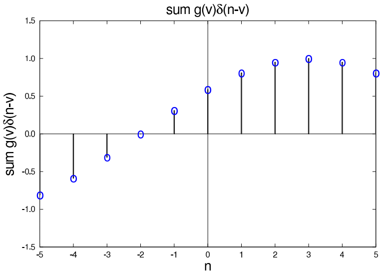
Sifting *cont'd.*



- We can now add all these together...

Sifting *cont'd.*

- Adding all the delta values together we get



- which is a discrete (*sifted*) representation of the original signal, $g(n)$.
- This process can be represented by

$$x[n] = \ldots + g(-5)\delta(n+5) + g(-4)\delta(n+4) + \ldots + g(4)\delta(n-4) + g(5)\delta(n-5) + \ldots$$

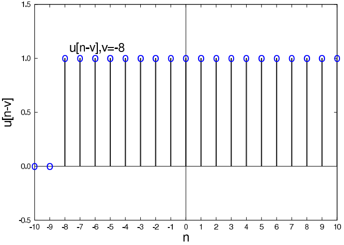
- where $[\cdot]$ signifies a discrete formulation. This can be shortened to $x[n] = \sum_{k=-\infty}^{\infty} g(k)\delta(n-k)$. For our case $x[n] = \sum_{k=-5}^{5} g(k)\delta(n-k)$.

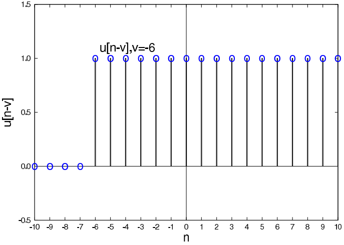
Unit Step Function

- The unit step function:

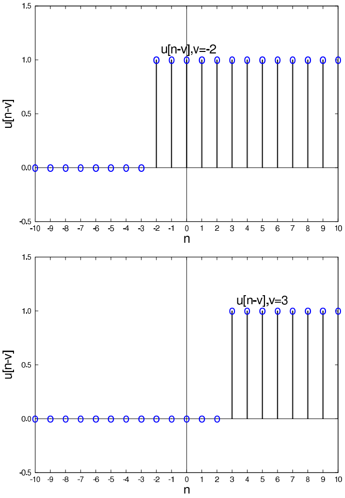
$$u[n-v] = \begin{cases} 1 & \text{if} \quad n - v \geq 0, \\ 0 & \quad\;\; \text{otherwise.} \end{cases}$$

- switches from zero to unit value.
- *Examples*





Unit Step Function *Examples cont'd.*



Unit Step Function

- It can be defined using the unit impulse function ($\delta[n-v]$):

$$u[n-v] = \sum_{m=\infty}^{\infty} \delta[m-v]$$

- Also

$$\delta[n-v] = u[n-v] - u[n-1-v].$$

- These are known as *recurrence* formula, where the current signal value is dependent on previous signal values:

  "to recur"

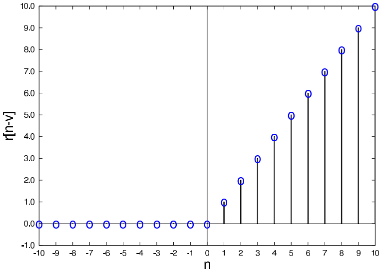
  Meaning: to repeat.

Ramp Function

- Another interesting function type is the ramp function.
- Given by

$$r[n-v] = (n-v)u[n].$$

Example



Digital Sine and Cosine Functions

- Digital sine wave:

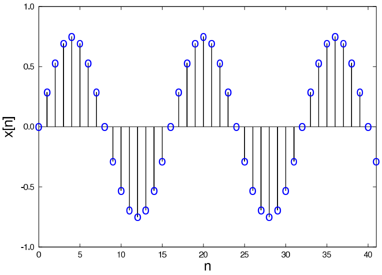
$$x[n] = a\sin(n\Omega + \theta)$$

- Digital cosine wave:

$$x[n] = a\cos(n\Omega + \theta)$$

- Ω is the *digital "frequency"* measured in *radians*
- 1 cycle every N samples. Also $\Omega = 2\pi/N$ so that $N = 2\pi/\Omega$

- *Example* $a = 0.75$, $\theta = 0$ and $\Omega = \pi/8$, therefore $N = 2 \times 8 = 16$ *i.e.* $x[n] = 0.75\sin(n\pi/8)$:



Comparison with Analog Sine Function

- Compare to a continuous analog sine wave:

$$x(t) = a\sin(t\omega + \theta)$$

  where t could be time in seconds and $\omega = 2\pi f$ is the angular frequency, therefore in *radians per second*.
- The interval between each sample n is T_s seconds, so there is a sample at every $t = nT_s$ seconds
- The continuous sine wave can then be written as

$$x(n) = a\sin(nT_s 2\pi f + \theta)$$

- If we equate the continuous and digital versions, then

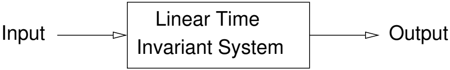
$$x[n] = x(n)$$

$$a\sin(n\Omega + \theta) = a\sin(nT_s 2\pi f + \theta)$$

- Therefore $\Omega = T_s 2\pi f$ *or* if sampling frequency is $f_s = 1/T_s$ then $\Omega = 2\pi f / f_s$.

Outline

Linear Time Invariant Systems

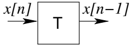


- **Time Invariance**:
  - The same response to the same input at any time. *e.g.*
    If $y[n+v] = x[n+v]$ for any value of v.
- **Linear System**:
  - *Principle of Superposition*:
    - If the input consists of a sum of signals then the output is the sum of the responses to those signals, *e.g.*

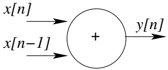
If the output of a system is $y_1[n]$ and $y_2[n]$ in response to two different inputs $x_1[n]$ and $x_2[n]$ respectively then the output of the same system for the two inputs weighted and combined *i.e.* $ax_1[n] + bx_2[n]$ will be $ay_1[n] + by_2[n]$ where a and b are constants.

Linear Time Invariant Systems

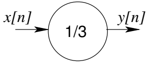
- A Linear Time Invariant (LTI) system consists of:
  - Storage / Delay:



  - Addition / Subtraction: *e.g.* $y[n] = x[n] + x[n-1]$

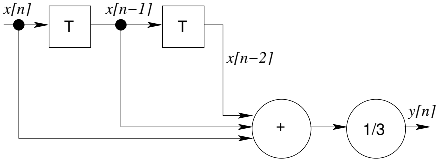


  - Multiplication by Constants: *e.g.* $y[n] = \frac{1}{3}x[n]$



- *Example*

  Moving average filter, $y[n] = \frac{x[n]+x[n-1]+x[n-2]}{3}$



Other System Properties

An LTI system is

- Associative, where a system can be broken down into simpler subsystems for analysis or synthesis
- Commutative, where if a system is composed of a series of subsystems then the subsystems can be arranged in any order

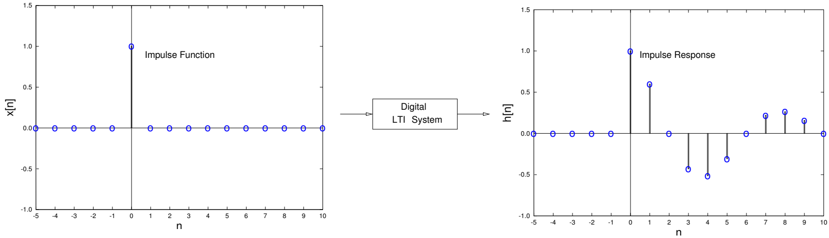
LTI systems may also have

- Causality: output does not depend on future input values
- Stability: output is bounded for a bounded input (see Lecture 04)
- Invertibility: input can be uniquely found from the input (*e.g.* the square of a number is not invertible)
- Memory: output depends on past input values

Impulse Response

An LTI system possesses an Impulse Response which characterizes the system's output if an impulse function is applied to the input.

Example Impulse Response



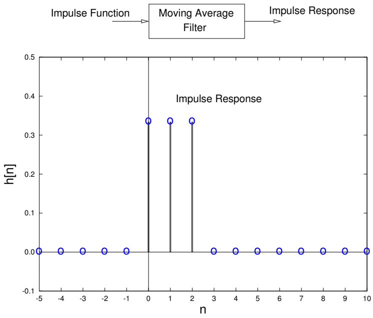
Impulse Response *Example*

Remember the moving average filter:

$$y[n] = \frac{1}{3}(x[n] + x[n-1] + x[n-2])$$

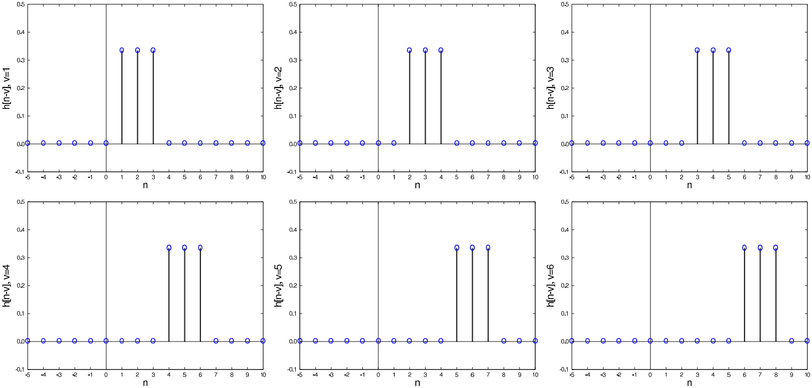
If the input is the impulse function: $x[n=0] = \delta(0)$, then $y[n]$ is the output in response to an impulse function, *i.e.* the **impulse response** hence $h[n] = y[n]$...
At time $n = -1$:
$x[n=-1] = \delta[-1] = 0$;
At time $n = 0$: $x[n=0] = \delta[0] = 1$;
At time $n = 1$: $x[n=1] = \delta[1] = 0$
etc.
Therefore

- $h[n < 0] = y[n < 0] = 0$
- $h[0] = y[0] = \frac{1}{3}(\delta[0] + \delta[-1] + \delta[-2]) = \frac{1}{3}$
- $h[1] = y[1] = \frac{1}{3}(\delta[+1] + \delta[0] + \delta[-1]) = \frac{1}{3}$
- $h[2] = y[2] = \frac{1}{3}(\delta[+2] + \delta[+1] + \delta[0]) = \frac{1}{3}$
- $h[n > 2] = y[n > 2] = 0$



Impulse Response *Examples - shifting*

The impulse response can also be determined for a shifted impulse function, *i.e.* $\delta[n-v]$



What will the system output ($y[n]$) be if the input consists of more than one impulse function shifted by different amounts?

System Response to Multiple Shifted Impulse Responses

What will the system output ($y[n]$) be if the input consists of more than one impulse function shifted by different amounts?
Remember that all LTI systems obey the *"Principle of Superposition"*...
So, for the inputs

$$x_1[n] = a\delta[n] \text{ and } x_2[n] = b\delta[n-1],$$

where a and b are constants, the corresponding outputs will be

$$y_1[n] = ah[n] \text{ and } y_2[n] = bh[n-1],$$

i.e. impulse responses. Therefore if
$x[n] = x_1[n] + x_2[n] = a\delta[n] + b\delta[n-1]$ then

$$y[n] = ah[n] + bh[n-1].$$

Other System Inputs: Step Function

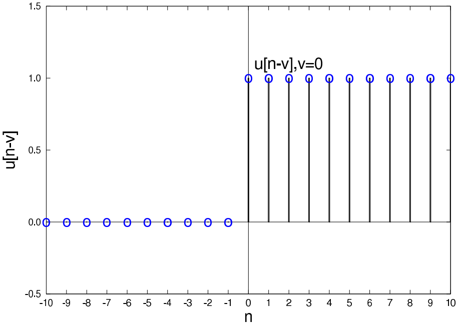
- The discrete step function can be thought of as a series of impulse functions (remember sifting).
- Each impulse function creates an impulse response.
- The output is then the joint response of all the impulse responses scaled by the inputs.

- A discretely sampled step input (starting at $n = 0$) is given by:

$$x[n] = \sum_{k=0}^{\infty} \delta(n-k).$$

- Therefore, using the *Principle of Superposition* we get

$$y[n] = \sum_{k=0}^{\infty} h(n-k).$$



Moving Average of a Step Function

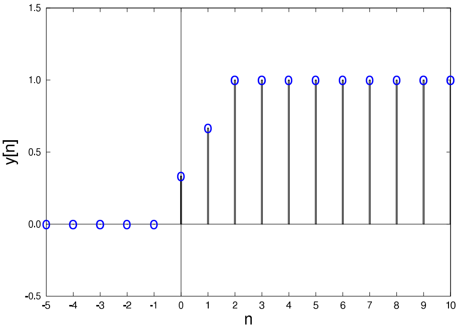
Moving average (with $k = 3$) has an impulse response:

- $h[n < 0] = y[n < 0] = 0$
- $h[0] = y[0] = \frac{1}{3}(\delta[0] + \delta[-1] + \delta[-2]) = \frac{1}{3}$
- $h[1] = y[1] = \frac{1}{3}(\delta[+1] + \delta[0] + \delta[-1]) = \frac{1}{3}$
- $h[2] = y[2] = \frac{1}{3}(\delta[+2] + \delta[+1] + \delta[0]) = \frac{1}{3}$
- $h[n > 2] = y[n > 2] = 0$

Moving average of a step function is then:

$$y[n] = \sum_{k=0}^{\infty} h(n-k)$$

$$= \begin{cases} 0 & \text{if} \quad n \leq 0 \\ 1/3 & \text{if} \quad n = 0 \\ 2/3 & \text{if} \quad n = 1 \\ 1 & \text{if} \quad n \geq 2 \end{cases}$$



Scaled Impulse Function Inputs

What happens when the step function is given by:

$$u[n-v] = \begin{cases} a & \text{if} \quad n - v \geq 0, \\ 0 & \quad\;\, \text{otherwise} \end{cases} ?$$

The discrete impulse function version is

$$x[n] = \sum_{k=0}^{\infty} a\delta[n-k].$$

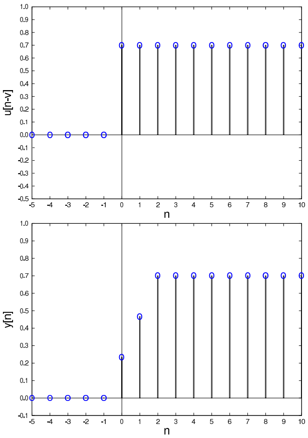
Using the *Principle of Superposition*:

$$y[n] = \sum_{k=0}^{\infty} ah[n-k].$$

Example Moving average filter, $k = 3$

$$y[n] = \begin{cases} 0 & \text{if} \quad n \leq 0 \\ a/3 & \text{if} \quad n = 0 \\ 2a/3 & \text{if} \quad n = 1 \\ a & \text{if} \quad n \geq 2 \end{cases}$$

Example Moving Average and $a = 0.7$



Outline

Digital Convolution

What happens if the scale of the input impulse functions (a) varies with n? *i.e.*

$$x[n] = a[n]\delta[n-k].$$

Using the *Principle of Superposition* we get

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

This is known as the **Convolution Sum**.
Example

$$x[n] = \begin{cases} 0 & \text{if} \quad n \leq 0 \\ a[0] & \text{if} \quad n = 1 \\ a[1] & \text{if} \quad n = 2 \\ 0 & \text{if} \quad n \geq 0 \end{cases},$$

which is the same as $x[n] = a[0]\delta[n] + a[1]\delta[n-1]$. Then

$$y[n] = a[0]h[n] + a[1]h[n-1].$$

Digital Convolution *Example*

Q. Find $y[n]$ if $a[0] = 1$ and $a[1] = 2$ using the impulse response of the moving average filter, $k = 3$.

A.

$$y[n] = a[0]h[n] + a[1]h[n-1] = h[n] + 2h[n-1]$$

$$y[-1] = h[-1] + 2h[-2] = 0 + 0 = 0$$

$$y[0] = h[0] + 2h[-1] = 1/3 + 0 = 1/3$$

$$y[1] = h[1] + 2h[0] = 1/3 + 2/3 = 1$$

$$y[2] = h[2] + 2h[1] = 1/3 + 2/3 = 1$$

$$y[3] = h[3] + 2h[2] = 0 + 2/3 = 2/3$$

$$y[4] = h[4] + 2h[3] = 0 + 0 = 0$$

Digital Convolution Trivia

Convolution is often represented by an asterik:

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

Convolution is commutative:

$$\begin{aligned} y[n] &= a[n] * h[n] = h[n] * a[n] \\ &= \sum_{-\infty}^{\infty} h[k]a[n-k]. \end{aligned}$$

Convolution is associative: *cascaded systems*

$$\{x[n] * h_1[n]\} * h_2[n] = x[n] * \{h_1[n] * h_2[n]\}$$

Convolution is distributive: *systems in parallel*

$$x[n] * \{h_1[n] + h_2[n]\} = x[n] * h_1[n] + x[n] * h_2[n]$$

Outline

Digital Cross-Correlation

Cross-correlation can be used to compare 2 signals.

- If $x_1[n]$ and $x_2[n]$ are two signals then digital cross-correlation is defined:

$$y[l] = \sum_{m=-\infty}^{\infty} x_1^*[m]x_2[n+m]$$

  where $x_1^*[n]$ is the complex conjugate of $x_1[n]$.
- For a real signal $x_1^*[n] = x_1[n]$.
- l is the *lag*.
- If $x_1[n]$ and $x_2[n]$ are the same signal but with a delay between them, then $y[l]$ is at a maximum when l is equal to this delay.

Digital Cross-Correlation *Example*

Q. Given $x_1 = (0\ 0\ 0.5\ 0.7\ 0)^{\mathrm{T}}$ and $x_2 = (0\ 0.5\ 0.7\ 0\ 0)^{\mathrm{T}}$.
Calculate the cross-correlation for these two real signals.
A. Cross correlation for a real signal is:

$$y[l] = \sum_{m=-\infty}^{\infty} x_1[m]x_2[n+m].$$

There are 5 elements in these vectors so (changing the limits):

$$y[l] = \sum_{m=0}^{4} x_1[m]x_2[n+m].$$

We can then calculate the results. Some example calculations:

$$\begin{aligned} y[l=0] =& \overbrace{x_1[0] \times x_2[0]}^{l=0,m=0} + \overbrace{x_1[1] \times x_2[1]}^{l=0,m=1} + x_1[2] \times x_2[2] + x_1[3] \times x_2[3] + \overbrace{x_1[4] \times x_2[4]}^{l=0,m=4} \\ =& 0 \times 0 + 0 \times 0.5 + \underline{0.5 \times 0.7} + 0.7 \times 0 + 0 \times 0 = \underline{0.5 \times 0.7} = 0.35 \end{aligned}$$

$$\begin{aligned} y[l=1] =& \overbrace{x_1[0] \times x_2[0+1]}^{l=1,m=0} + \overbrace{x_1[1] \times x_2[1+1]}^{l=1,m=1} + x_1[2] \times x_2[2+1] + \\ & x_1[3] \times x_2[3+1] + \overbrace{x_1[4] \times x_2[4+1]}^{l=1,m=4} \\ =& 0 \times 0.5 + 0 \times 0.7 + 0 \times 0 + 0 \times 0 + 0 \times 0 = 0 \end{aligned}$$

Digital Cross-Correlation *Example cont'd.*

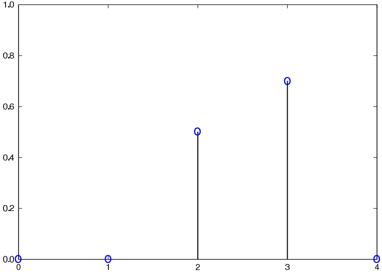
Here are the results for each combination of l and m values:

l	m **0**	**1**	**2**	**3**	**4**	$y[l]$
-5	0	0	0	0	0	0
-4	0	0	0	0	0	0
-3	0	0	0	0	0	0
-2	0	0	0	0.35	0	0.35
-1	0	0	0.25	0.49	0	$\underbrace{0.74}_{PEAK}$
0	0	0	0.35	0	0	0.35
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0

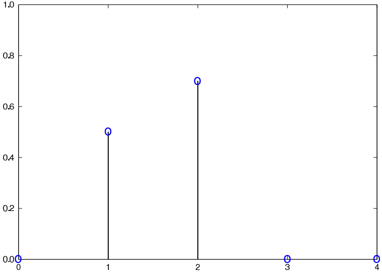
- A peak at $l = -1$.
- l is the *lag*, so there is a lag of -1.
- This means x_1 has some similar signal as x_2 but lagged by 1 step.
- We can also see from the signal definitions $x_1 = (0\ 0\ 0.5\ 0.7\ 0)^{\mathrm{T}}$ and $x_2 = (0\ 0.5\ 0.7\ 0\ 0)^{\mathrm{T}}$ that $x_1[n-1] = x_2[n]$.

Digital Cross-Correlation *Example cont'd.*

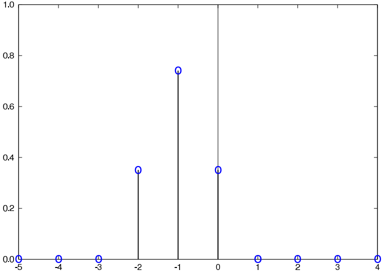
$$x_1[n] =$$



$$x_2[n] =$$



The result of the digital cross-correlation, $y[l] =$



Outline

Difference Equations

Difference equations are the name given to the equations that describe the digital signals and systems. For example the equation for the moving average filter with $k = 3$:

$$y[n] = \frac{x[n] + x[n-1] + x[n-2]}{3}$$

is known as a difference equation.
Difference equations for LTI systems can always be put in the form:

$$\sum_{m=0}^{N} a[m]y[n-m] = \sum_{m=0}^{M} b[m]x[n-m].$$

So for our moving average filter:

- $M = 2$ and $N = 0$.
- $a[m]$ and $b[m]$ are known as coefficients.
- For the moving average output y there is only one coefficient, $a[0] = 1$.
- For the moving average input x, there are three coefficients $b[0] = b[1] = b[2] = \frac{1}{3}$.

Outline

Lecture Summary

Today we have covered

- Types of digital signal, *e.g.* unit impulse function
- Sifting
- Digital sine and cosine functions
- Linear time invariant (LTI) systems
- Impulse response
- Moving average of a step function
- Digital convolution
- Digital cross-correlation
- Generalized difference equation for LTI systems