Lecture 12: Linearity and Shift-Invariance

Mark Hasegawa-Johnson

ECE 401: Signal and Image Analysis, Fall 2021
1 Systems
2 Linearity
3 Shift Invariance
4 Convolution
5 Written Example
6 Summary
What is a System?

A **system** is anything that takes one signal as input, and generates another signal as output. We can write

\[ x[n] \xrightarrow{\mathcal{H}} y[n] \]

which means

\[ x[n] \rightarrow \mathcal{H} \rightarrow y[n] \]
Example: Averager

For example, a weighted local averager is a system. Let’s call it system $A$.

$$x[n] \xrightarrow{A} y[n] = \sum_{m=0}^{6} g[m]x[n - m]$$
Example: Time-Shift

A time-shift is a system. Let’s call it system $\mathcal{T}$.

$$x[n] \xrightarrow{\mathcal{T}} y[n] = x[n - 1]$$
If you calculate the square of a signal, that’s also a system. Let’s call it system $S$:

$$x[n] \xrightarrow{S} y[n] = x^2[n]$$
Example: Add a Constant

If you add a constant to a signal, that’s also a system. Let’s call it system $C$:

$$x[n] \xrightarrow{C} y[n] = x[n] + 1$$
Example: Window

If you chop off all elements of a signal that are before time 0 or after time \(N - 1\) (for example, because you want to put it into an image), that is a system:

\[
x[n] \xrightarrow{W} y[n] = \begin{cases} 
x[n] & 0 \leq n \leq N - 1 \\
0 & \text{otherwise}
\end{cases}
\]
Outline

1. Systems
2. Linearity
3. Shift Invariance
4. Convolution
5. Written Example
6. Summary
A system is **linear** if these two algorithms compute the same thing:
A system $\mathcal{H}$ is said to be **linear** if and only if, for any $x_1[n]$ and $x_2[n]$, 

\begin{align*}
x_1[n] \xrightarrow{\mathcal{H}} y_1[n] \\
x_2[n] \xrightarrow{\mathcal{H}} y_2[n]
\end{align*}

implies that

\[ x[n] = x_1[n] + x_2[n] \xrightarrow{\mathcal{H}} y[n] = y_1[n] + y_2[n] \]

In words: a system is **linear** if and only if, for every pair of inputs $x_1[n]$ and $x_2[n]$, (1) adding the inputs and then passing them through the system gives exactly the same effect as (2) passing both inputs through the system, and then adding them.
Notice, a special case of linearity is the case when $x_1[n] = x_2[n]$: 

\[
\begin{align*}
    x_1[n] & \xrightarrow{\mathcal{H}} y_1[n] \\
    x_1[n] & \xrightarrow{\mathcal{H}} y_1[n]
\end{align*}
\]

implies that 

\[
    x[n] = 2x_1[n] \xrightarrow{\mathcal{H}} y[n] = 2y_1[n]
\]

So if a system is linear, then scaling the input also scales the output.
Example: Averager

Let’s try it with the weighted averager.

\[
x_1[n] \xrightarrow{A} y_1[n] = \sum_{m=0}^{6} g[m]x_1[n - m]
\]

\[
x_2[n] \xrightarrow{A} y_2[n] = \sum_{m=0}^{6} g[m]x_2[n - m]
\]

Then:

\[
x[n] = x_1[n] + x_2[n] = \sum_{m=0}^{6} g[m] (x_1[n - m] + x_2[n - m])
\]

\[
= \left( \sum_{m=0}^{6} g[m]x_1[n - m] \right) + \left( \sum_{m=0}^{6} g[m]x_2[n - m] \right)
\]

\[
= y_1[n] + y_2[n]
\]

...so a weighted averager is a **linear system**.
Example: Square

A squarer is just obviously nonlinear, right? Let’s see if that’s true:

\[ x_1[n] \xrightarrow{S} y_1[n] = x_1^2[n] \]
\[ x_2[n] \xrightarrow{S} y_2[n] = x_2^2[n] \]

Then:

\[ x[n] = x_1[n] + x_2[n] \xrightarrow{A} y[n] = x^2[n] \]
\[ = (x_1[n] + x_2[n])^2 \]
\[ = x_1^2[n] + 2x_1[n]x_2[n] + x_2^2[n] \]
\[ \neq y_1[n] + y_2[n] \]

\[ \ldots \text{so a squarer is a \textbf{nonlinear system}.} \]
Example: Add a Constant

This one is tricky. Adding a constant seems like it ought to be linear, but it’s actually **nonlinear**. Adding a constant is what’s called an **affine** system, which is not necessarily linear.

\[
x_1[n] \xrightarrow{c} y_1[n] = x_1[n] + 1
\]

\[
x_2[n] \xrightarrow{c} y_2[n] = x_2[n] + 1
\]

Then:

\[
x[n] = x_1[n] + x_2[n] \xrightarrow{A} y[n] = x[n] + 1 = x_1[n] + x_2[n] + 1 \neq y_1[n] + y_2[n]
\]

...so adding a constant is a **nonlinear** system.
What about the real world?

Suppose you’re showing people images $x[n]$, and measuring their brain activity $y[n]$ as a result. How can you tell if this system is linear?

- Show them one image, call it $x_1[n]$. Measure the resulting brain activity, $y_1[n]$.
- Show them another image, $x_2[n]$. Measure the brain activity, $y_2[n]$.
- Show them $x[n] = x_1[n] + x_2[n]$. Measure $y[n]$. Is it equal to $y_1[n] + y_2[n]$?
- Repeat this experiment with lots of different images, and their sums, until you are convinced that the system is linear (or not).
Outline

1. Systems
2. Linearity
3. Shift Invariance
4. Convolution
5. Written Example
6. Summary
A system $\mathcal{H}$ is **shift-invariant** if these two algorithms compute the same thing (here $\mathcal{T}$ means “time shift”):

\[
x[n] \xrightarrow{T} x[n-1] \xrightarrow{\mathcal{H}} y[n] \xrightarrow{T} y[n-1]
\]

\[
x[n] \xrightarrow{\mathcal{H}} y[n] \xrightarrow{T} y[n-1]
\]
A system $\mathcal{H}$ is said to be **shift-invariant** if and only if, for every $x_1[n]$,

$$x_1[n] \xrightarrow{\mathcal{H}} y_1[n]$$

implies that

$$x[n] = x_1[n - n_0] \xrightarrow{\mathcal{H}} y[n] = y_1[n - n_0]$$

In words: a system is **shift-invariant** if and only if, for any input $x_1[n]$, (1) shifting the input by some number of samples $n_0$, and then passing it through the system, gives exactly the same result as (2) passing it through the system, and then shifting it.
Example: Averager

Let’s try it with the weighted averager.

\[ x_1[n] \xrightarrow{A} y_1[n] = \sum_{m=0}^{6} g[m] x_1[n - m] \]

Then:

\[ x[n] = x_1[n - n_0] \xrightarrow{A} y[n] = \sum_{m=0}^{6} g[m] x[n - m] = \sum_{m=0}^{6} g[m] x_1[(n - m) - n_0] \]

\[ = \sum_{m=0}^{6} g[m] x_1[(n - n_0) - m] = y_1[n - n_0] \]

...so a weighted averager is a **shift-invariant system**.
Example: Square

Squaring the input is a nonlinear operation, but is it shift-invariant? Let’s find out:

\[ x_1[n] \xrightarrow{S} y_1[n] = x_1^2[n] \]

Then:

\[ x[n] = x_1[n - n_0] \xrightarrow{A} y[n] = x^2[n] \]

\[ = (x_1[n - n_0])^2 \]

\[ = x_1^2[n - n_0] \]

\[ = y_1[n - n_0] \]

\[ \ldots \text{so computing the square is a shift-invariant system.} \]
Example: Windowing

How about windowing, e.g., in order to create an image?

\[ x_1[n] \xrightarrow{\mathcal{W}} y_1[n] = \begin{cases} x_1[n] & 0 \leq n \leq N - 1 \\ 0 & \text{otherwise} \end{cases} \]

If we shift the output, we get

\[ y_1[n - n_0] = \begin{cases} x_1[n - n_0] & n_0 \leq n \leq N - 1 + n_0 \\ 0 & \text{otherwise} \end{cases} \]

... but if we shift the input \((x[n] = x_1[n - n_0])\), we get

\[ y[n] = \begin{cases} x[n] & 0 \leq n \leq N - 1 \\ 0 & \text{otherwise} \end{cases} = \begin{cases} x_1[n - n_0] & 0 \leq n \leq N - 1 \\ 0 & \text{otherwise} \end{cases} \]

\[ \neq y_1[n - n_0] \]

... so windowing is a **shift-varying system** (not shift-invariant).
How about the real world?

Suppose you’re showing images $x[n]$, and measuring the neural response $y[n]$. How do you determine if this system is shift-invariant?

- Show an image $x_1[n]$, and measure the neural response $y_1[n]$.
- Shift the image by $n_0$ columns to the right, to get the image $x[n] = x_1[n - n_0]$. Show people $x[n]$.
- Is the resulting neural response exactly the same, but shifted to a slightly different set of neurons (shifted “to the right?”) If so, then the system may be shift-invariant!
- Keep doing that, with many different images and many different shifts, until you’re convinced the system is shift-invariant.
We care about linearity and shift-invariance because of the following remarkable result:

Let $H$ be any system,

$$x[n] \xrightarrow{H} y[n]$$

If $H$ is linear and shift-invariant, then whatever processes it performs can be equivalently replaced by a convolution:

$$y[n] = \sum_{m=-\infty}^{\infty} h[m] x[n - m]$$
The weights $h[m]$ are called the “impulse response” of the system. We can measure them, in the real world, by putting the following signal into the system:

$$\delta[n] = \begin{cases} 1 & n = 0 \\ 0 & \text{otherwise} \end{cases}$$

and measuring the response:

$$\delta[n] \xrightarrow{H} h[n]$$
**Convolution: Proof**

1. \( h[n] \) is the impulse response.
   \[ \delta[n] \xrightarrow{H} h[n] \]

2. The system is **shift-invariant**, therefore
   \[ \delta[n - m] \xrightarrow{H} h[n - m] \]

3. The system is **linear**, therefore **scaling the input by a constant** results in **scaling the output by the same constant**:
   \[ x[m]\delta[n - m] \xrightarrow{H} x[m]h[n - m] \]

4. The system is **linear**, therefore **adding input signals** results in **adding the output signals**:
   \[ \sum_{m=-\infty}^{\infty} x[m]\delta[n - m] \xrightarrow{H} \sum_{m=-\infty}^{\infty} x[m]h[n - m] \]
Convolution: Proof (in Words)

- The input signal, $x[n]$, is just a bunch of samples.
- Each one of those samples is a scaled impulse, so each one of them produces a scaled impulse response at the output.
- Convolution $=$ add together those scaled impulse responses.
Convolution: Proof (in Pictures)
Prove that differentiation, \( y(t) = \frac{dx}{dt} \), is a linear shift-invariant system (in terms of \( t \) as the time index, instead of \( n \)).
Outline

1. Systems
2. Linearity
3. Shift Invariance
4. Convolution
5. Written Example
6. Summary
Summary

- A system is **linear** if and only if, for any two inputs $x_1[n]$ and $x_2[n]$ that produce outputs $y_1[n]$ and $y_2[n]$,

  \[ x[n] = x_1[n] + x_2[n] \xrightarrow{\mathcal{H}} y[n] = y_1[n] + y_2[n] \]

- A system is **shift-invariant** if and only if, for any input $x_1[n]$ that produces output $y_1[n]$,

  \[ x[n] = x_1[n - n_0] \xrightarrow{\mathcal{H}} y[n] = y_1[n - n_0] \]

- If a system is **linear and shift-invariant** (LSI), then it can be implemented using convolution:

  \[ y[n] = h[n] * x[n] \]

  where $h[n]$ is the impulse response:

  \[ \delta[n] \xrightarrow{\mathcal{H}} h[n] \]