
Table of Contents

Mayank Vanjani	1
1: Convolution of Non-Causal Signals	1
2: Smoothing Data	2
3: Difference Equations	6
a: Linearity => Linear	6
b: Time-Invariance => Time Invariant	7
c: Impulse Response	8
d: Convolution and Function	9
e & f: Different Cases	10

Mayank Vanjani

Lab 3 9/28/18

```
% KEY POINTS
% length conv: Ly=Lx+Lh-1 => support Matlab: Ly-1
% supopr for not starting zero signal: Nx+Nh <= n <= Nx+Lx-1+Nh+Lh-1
```

1: Convolution of Non-Causal Signals

```
n = -10:10;
figure(1);
clf;

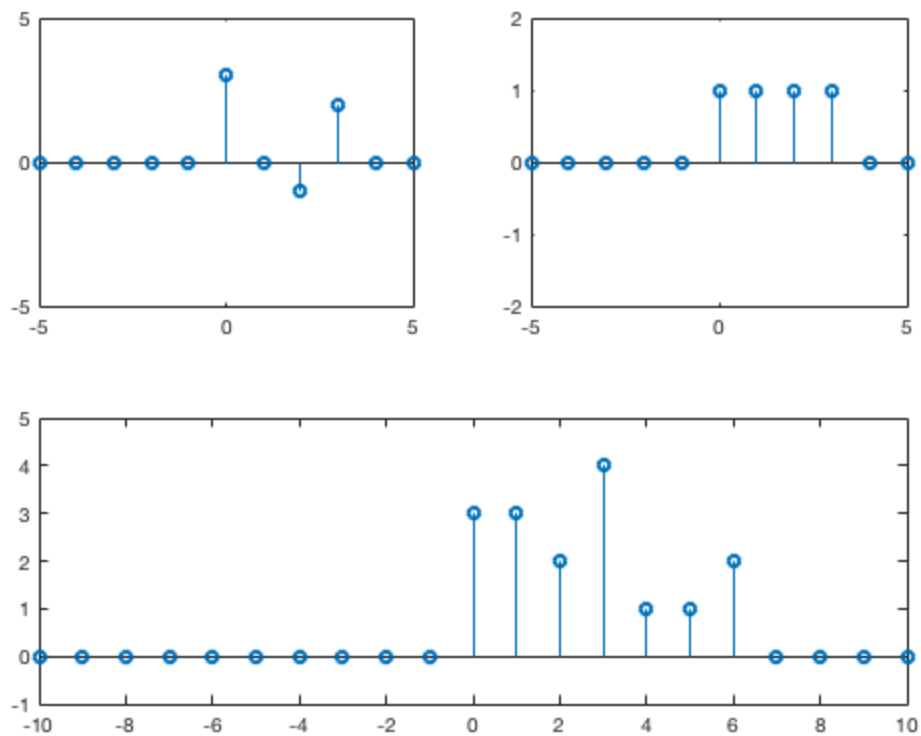
step = @(n, t) n >= t;
delta = @(n,t) n == t;
ramp = @(n,t) (n-t).*(n >= t);

f = 3*delta(n,0) - delta(n,2) + 2*delta(n,3);
Nf = length(f);
g = step(n,0) - step(n,4);
Ng = length(g);
x = conv(f,g);
Nx = Nf + Ng - 1; % or length(x);

subplot(2,2,1);
stem(n,f);
axis([-5 5 -5 5]);

subplot(2,2,2);
stem(n,g);
axis([-5 5 -2 2]);

subplot(2,2,[3,4]);
stem(-(Nx-1)/2:(Nx-1)/2,x);
axis([-10 10 -1 5]);
```



2: Smoothing Data

Delay of filter (shift back $(N-1)/2 == 5$ bc len is 11, change for higher)

```
load DataEOG.txt;
x = DataEOG;
Nx = length(x);

figure(2);
clf;
subplot(2,1,1);
stem(x);
subplot(2,1,2);
plot(x);

figure(3);
clf;
Nh = 11; % Using Nh = 11, 31, 67
h = ones(1,Nh)/Nh;
stem(0:Nh-1,h);
axis([-5 15 -0.2 0.2]);

figure(4);
clf;
y = conv(h,x);
Ny = Nh+Nx-1;
```

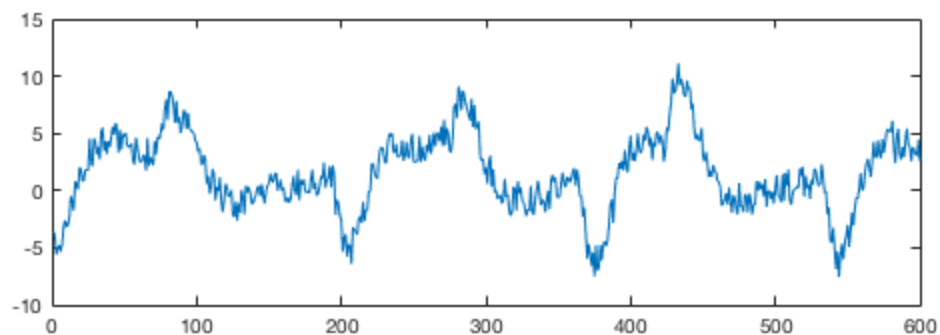
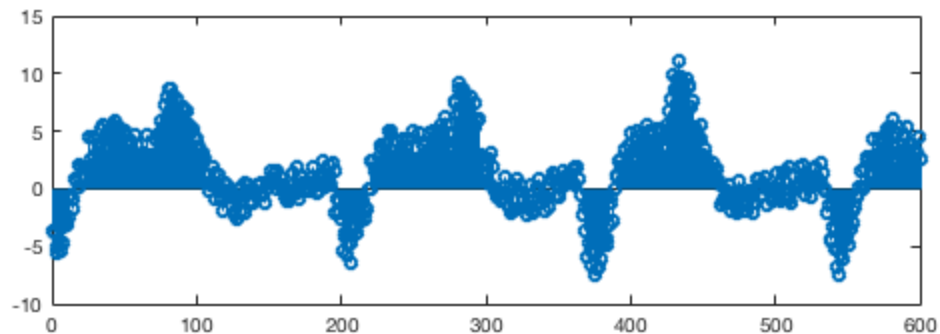
```

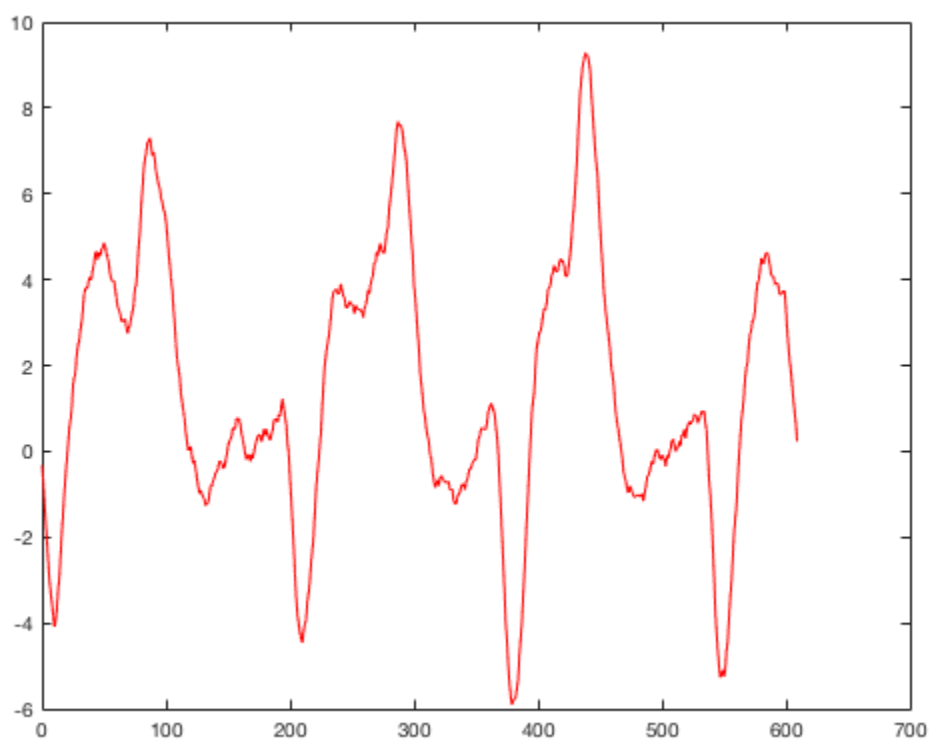
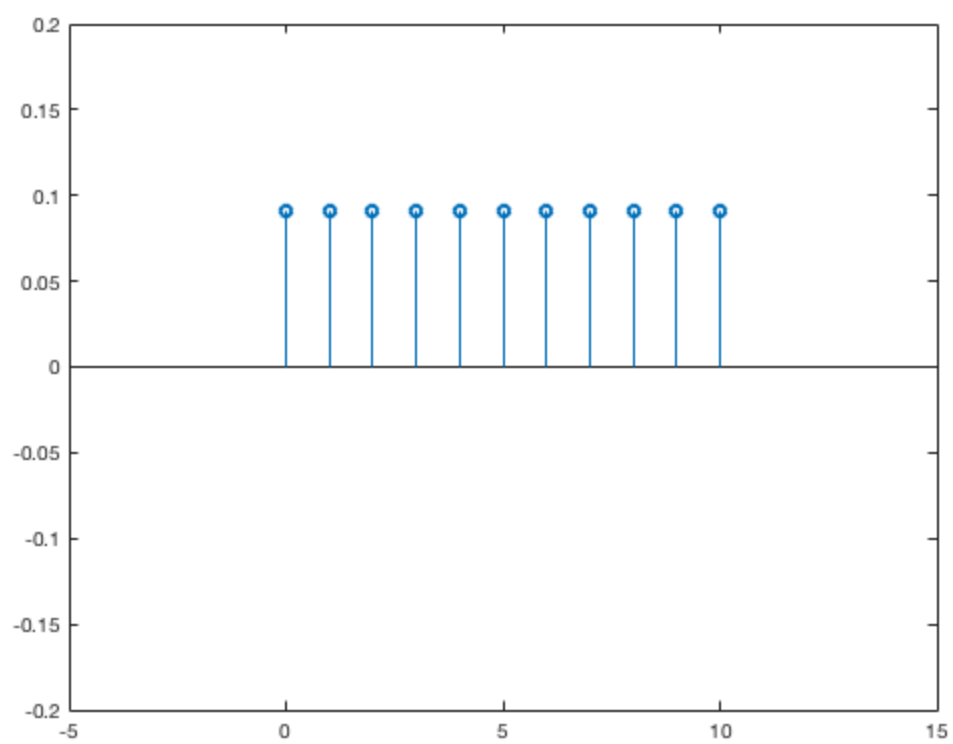
plot(0:Ny-1,y,'r');
%disp(length(x)); %600
%disp(length(h));
%disp(length(y)); %600+Nh-1

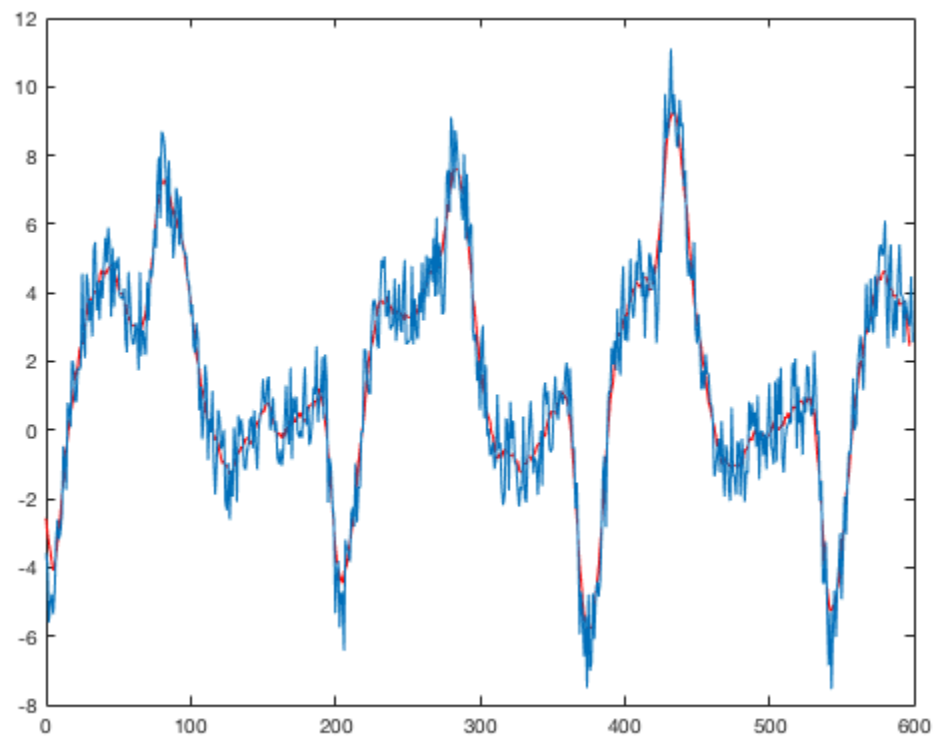
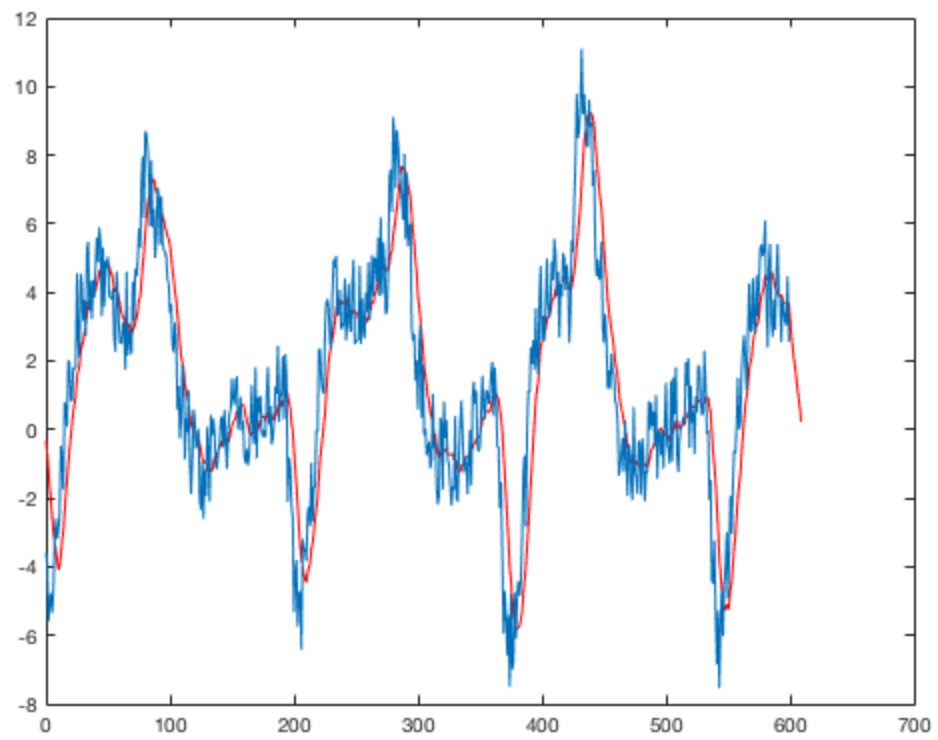
figure(5);
clf;
plot(0:Ny-1,y,'r', 0:Nx-1,x);

figure(6);
clf;
shift = (Nh-1)/2;
% The effect of these commands is to consider the shift factor when
% recreating a signal using convolution, this shift is dependent on
% impulse
% response lengths above Nh = 11, 31, 67... The higher this value is,
% the
% more samples are averaged per span of the input signal resulting in
% less
% deviation during spikes and more of an average of the signal.
y2 = y;
y2(1:shift) = [];
y2(end-shift-1:end) = [];
Ny2 = length(y2);
plot(0:Ny2-1,y2,'r', 0:Nx-1,x);

```







3: Difference Equations

a: Linearity => Linear

```
n = 0:20;
step = @(n, t) double(n >= t);
delta = @(n,t) double(n == t);

x1 = delta(n,0) + delta(n,2);
len_x1 = length(x1);
y1 = myDiffeq(x1,1);
len_y1 = length(y1);

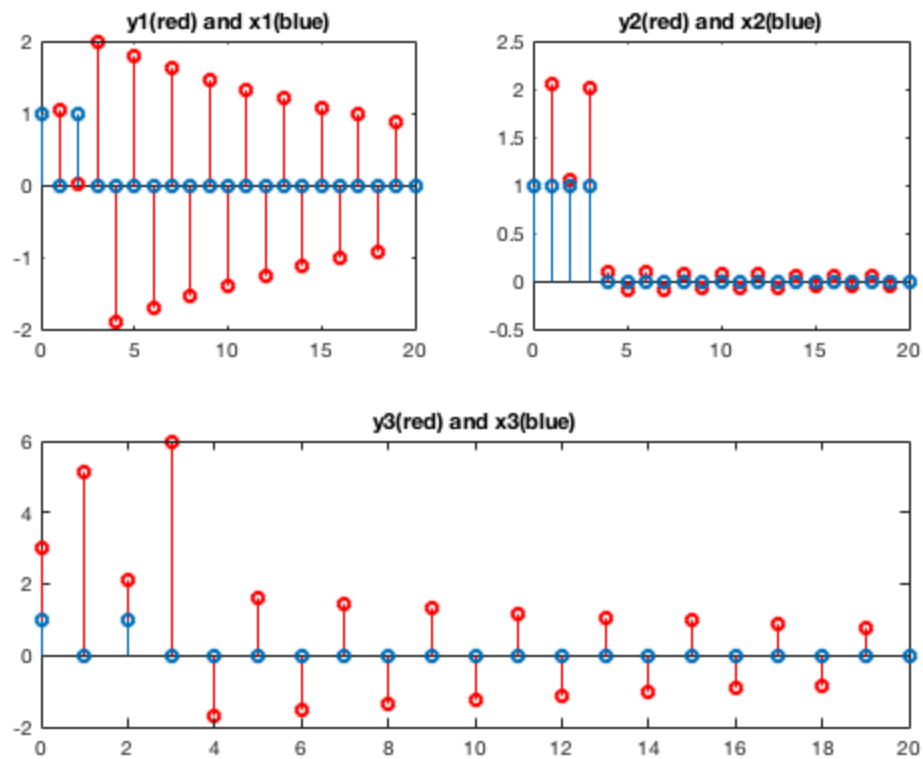
x2 = step(n,0) - step(n,4);
len_x2 = length(x2);
y2 = myDiffeq(x2,1);
len_y2 = length(y2);

x3 = x1 + 2*x2;
len_x3 = length(x3);
y3 = myDiffeq(x3,1);
len_y3 = length(y3);

figure(7);
clf;
subplot(2,2,1);
stem(0:len_y1-1,y1, 'r');
hold on;
stem(0:len_x1-1,x1);
hold off;
title("y1(red) and x1(blue)");

subplot(2,2,2);
stem(0:len_y2-1,y2, 'r');
hold on;
stem(0:len_x2-1,x2);
hold off;
title("y2(red) and x2(blue)");

subplot(2,2,[3,4]);
stem(0:len_y3-1,y3, 'r');
hold on;
stem(0:len_x3-1,x1);
hold off;
title("y3(red) and x3(blue)");
%Linear signal since we tested correctly for scaling and superposition
```



b: Time-Invariance => Time Invariant

```

x1 = delta(n,0) - delta(n,2);
len_x1 = length(x1);
y1 = myDiffeq(x1,1);
len_y1 = length(y1);

x2 = delta(n,3) - delta(n,5);
len_x2 = length(x2);
y2 = myDiffeq(x2,1);
len_y2 = length(y2);

figure(8);
clf;
subplot(2,1,1);
stem(0:len_x1-1,x1, 'r');
hold on;
stem(0:len_x2-1,x2);
hold off;
title("x1(red) and x2(blue)");
ylim([-2 2]);

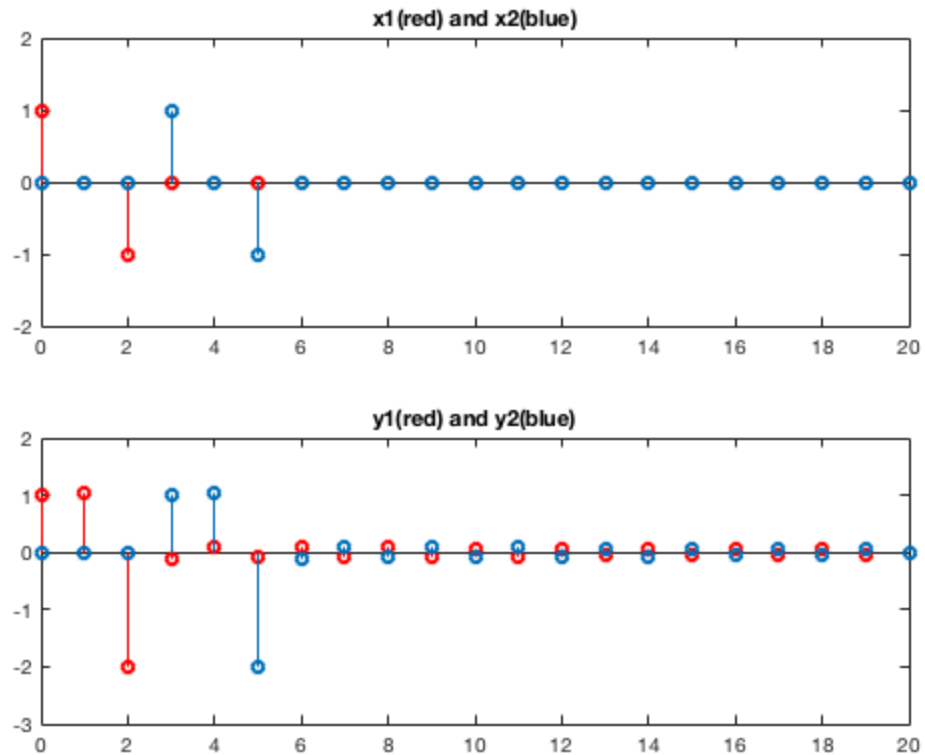
subplot(2,1,2);
stem(0:len_y1-1,y1, 'r');
hold on;

```

```

stem(0:len_y2-1,y2);
hold off;
title("y1(red) and y2(blue)");
ylim([-3 2]);
% Time Invariant signal since we tested for the same signal only under
a
% time shift

```



c: Impulse Response

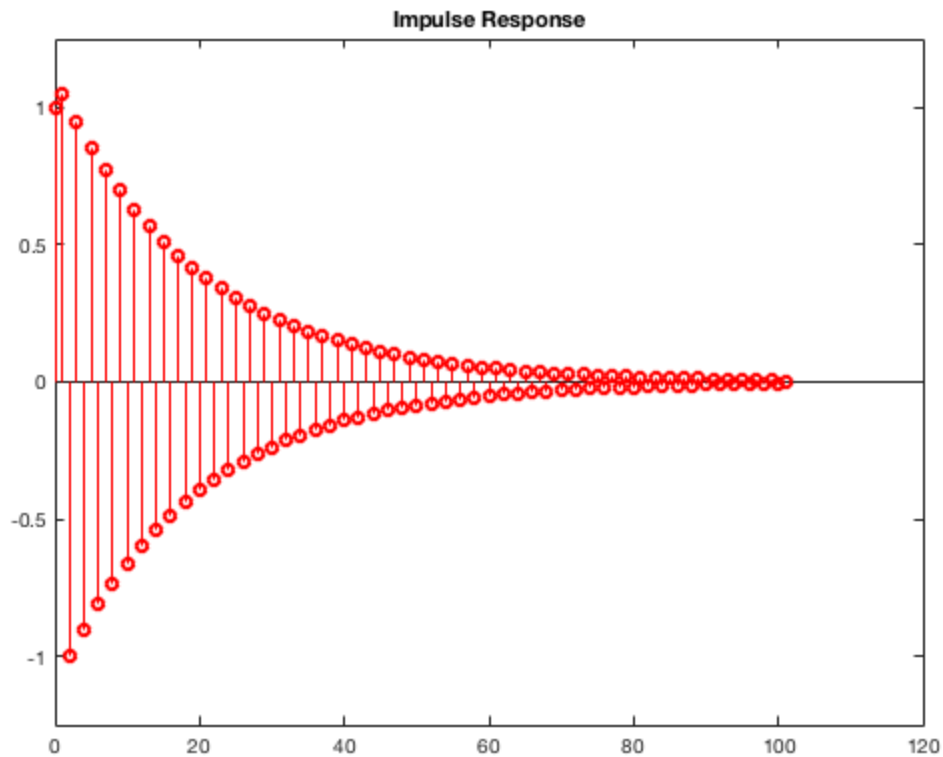
```

n = 0:101;

x1 = delta(n,0);
h = myDiffeq(x1,1);
len_h = length(h);

figure(9);
clf;
stem(0:len_h-1,h, 'r');
title("Impulse Response");
ylim([-1.25 1.25]);

```

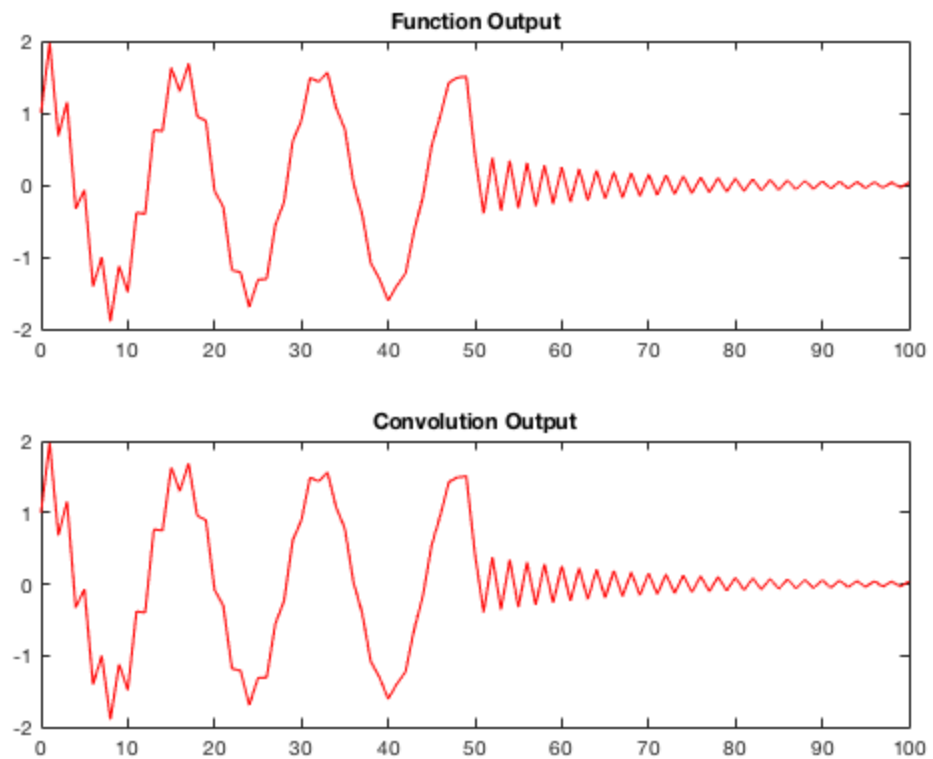



d: Convolution and Function

```
n = 0:101;

x = cos(pi*n/8) .* (step(n,0)-step(n,50));
len_x = length(x);
y = myDiffeq(x,1);
len_y = length(y);
y1 = conv(h,x);
len_y1 = length(y1);

figure(10);
clf;
subplot(2,1,1);
plot(0:len_y-1,y,'r');
title("Function Output");
axis([0 100 -2 2]);
subplot(2,1,2);
plot(0:len_y1-1,y1,'r');
title("Convolution Output");
axis([0 100 -2 2]);
% Same response using convolutions and and our function
% Characterize a system using convolution and difference equation
```



e & f: Different Cases

```
n = 0:50;

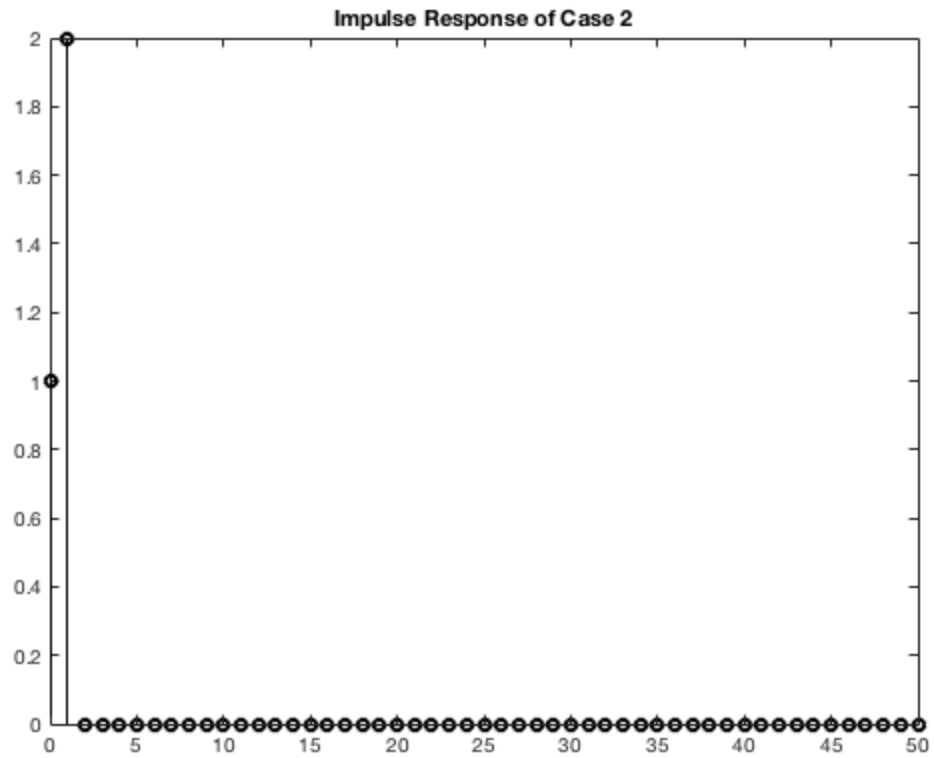
x = delta(n,0);
h = myDiffeq(x,2);
len_h = length(h);

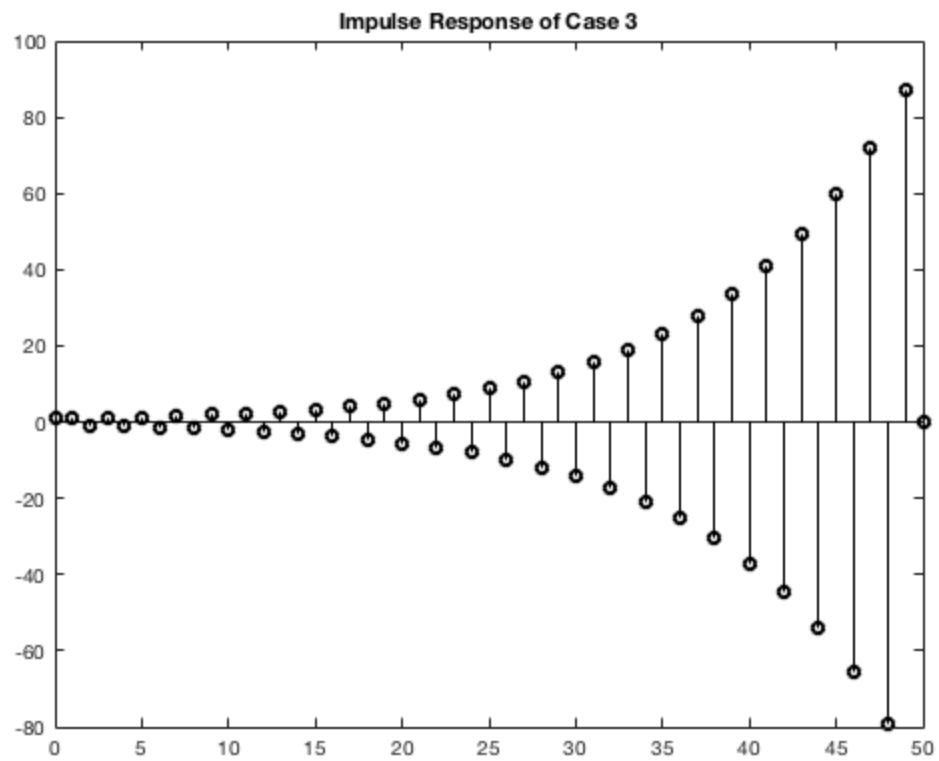
figure(11);
clf;
stem(0:len_h-1, h, 'k');
title("Impulse Response of Case 2");
% Not dependent on previous values of the output but only the current
% and
% previous values of the input

x1 = delta(n,0);
h1 = myDiffeq(x1,3);
len_h1 = length(h1);

figure(12);
clf;
stem(0:len_h1-1, h1, 'k');
title("Impulse Response of Case 3");
```

```
% Unstable signal since this is a function of previous inputs (which
starts
% with an impulse response) and keeps increasing since case 3 in our
% function is a multiplier >1 times our previous input
```





Published with MATLAB® R2018a