

```

%Lab 2, Part A

%first, condition the data from Lab 1 again
load('drop_data.mat')
d=(d1+d2+d4+d5)/4;
s=min(d);
d=d-s+.1;

%now make a sine wave for other test
t=linspace(0,6*pi,500);      %another way to create a vector.  this
                             %one has 500 points, whereas the
                             %t=[0:.1:6*pi] method creates an array with
                             %points that are separated by .1 intervals

g=sin(t);

%part a) First Derivative
[dt,dg] = derivative(t,g);
[dTime,dv] = derivative(Time,d);

%part b) Second Derivative
[ddt,ddg] = derivative(dt,dg);
[ddTime,da] = derivative(dTime,dv);

figure(1)
subplot(3,1,1)
plot(t,g), title('Sine Wave'),xlabel('t'),ylabel('sin(t)')
subplot(3,1,2)
plot(dt,dg),title('1st Derivative of Sine Wave'),xlabel('t'),
    ylabel('cos(t)')
subplot(3,1,3)
plot(ddt,ddg),title('2nd Derivative of Sine Wave'),xlabel('t'),
    ylabel('-sin(t)')

figure(2)
subplot(3,1,1)
plot(Time,d),title('Averaged, Normalized Ball-Drop Data'),
    xlabel('time'),ylabel('height')
subplot(3,1,2)
plot(dTime,dv),title('Velocity of Ball-Drop Data'),
    xlabel('time'),ylabel('height')
subplot(3,1,3)
plot(ddTime,da),title('Acceleration of Ball-Drop Data'),
    xlabel('time'),ylabel('height')

-----

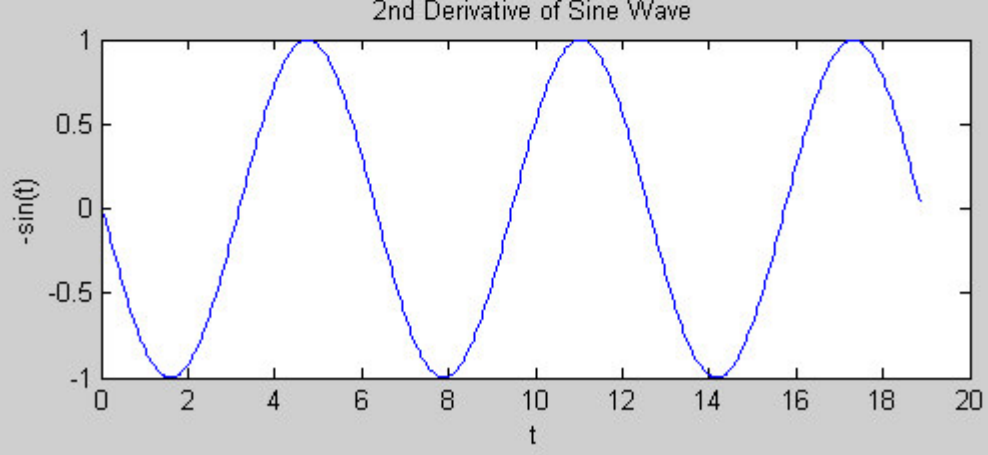
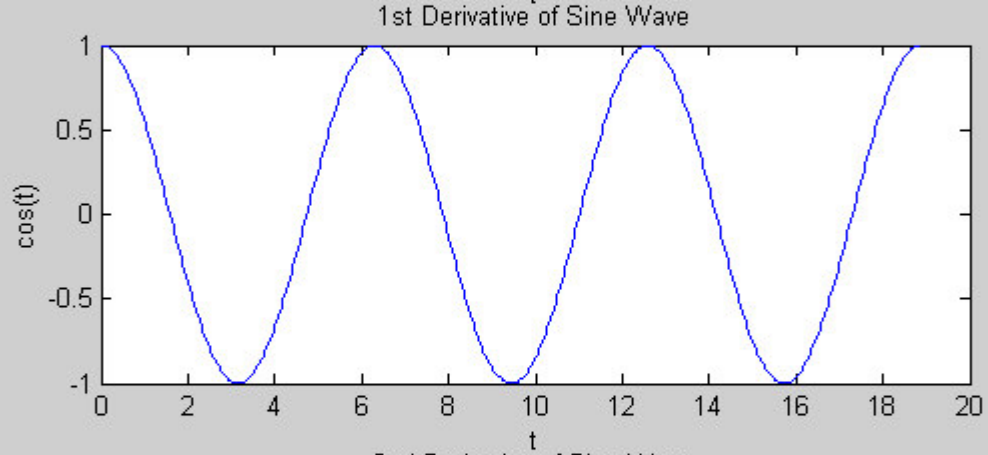
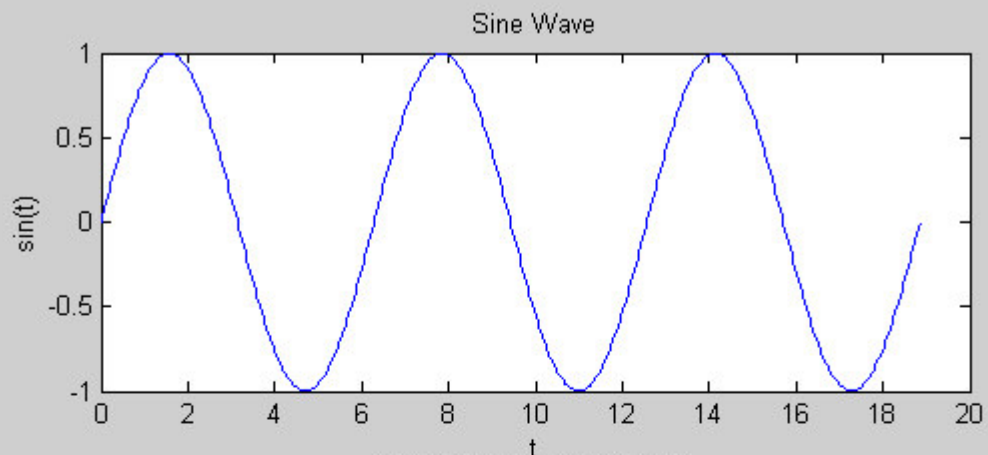
function [dervx, dervy] = derivative(x,y)

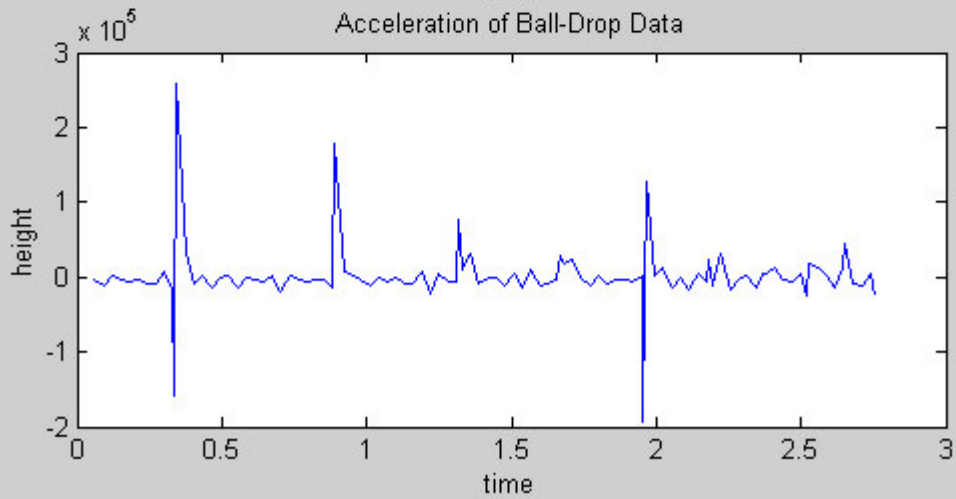
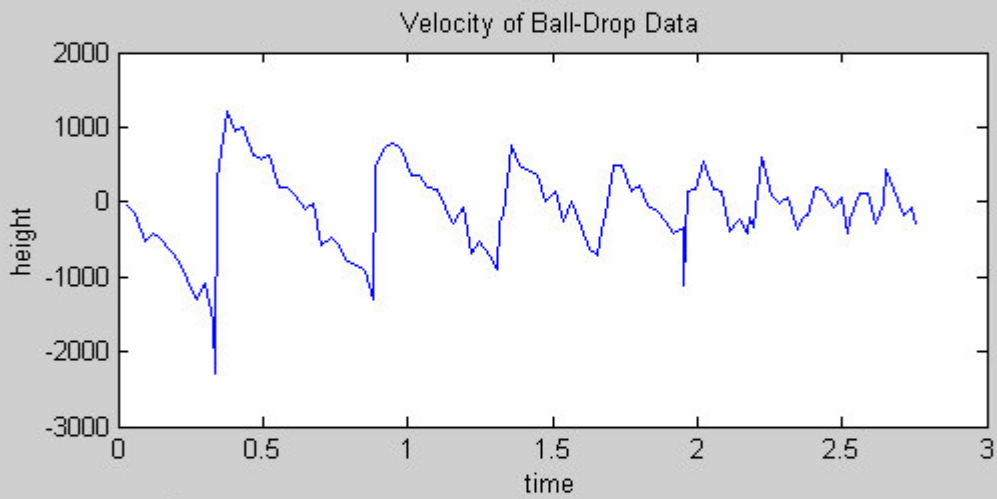
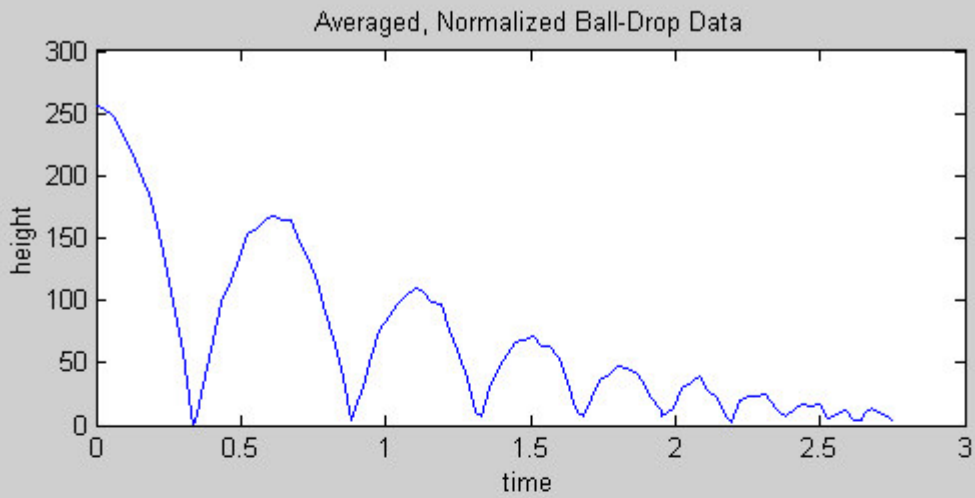
dervy = diff(y)./diff(x);      %could have also used a for loop here.
dervy = dervy';               %matrix gets tranposed for some reason.

dervx = x(2:length(x));      %shorten x vector accordingly
dervx = dervx';

```

```
>> lab2
```





## Part B

This part was also supposed to be related to a bouncing ball, but the way that the equation was given, it wasn't immediately obvious to show how to plot such a trajectory, showing when and where the ball hit the floor. It is perfectly all right to have plotted the parabolic trajectory that went down to -2000, like somebody falling off of a cliff or something, but here is an m-file which calculates the position and velocity and includes bouncing, and a coefficient of restitution equal to 0.8, as used later in the problem.

```
function [z,v,impact] = drop(z0,v0,tf)

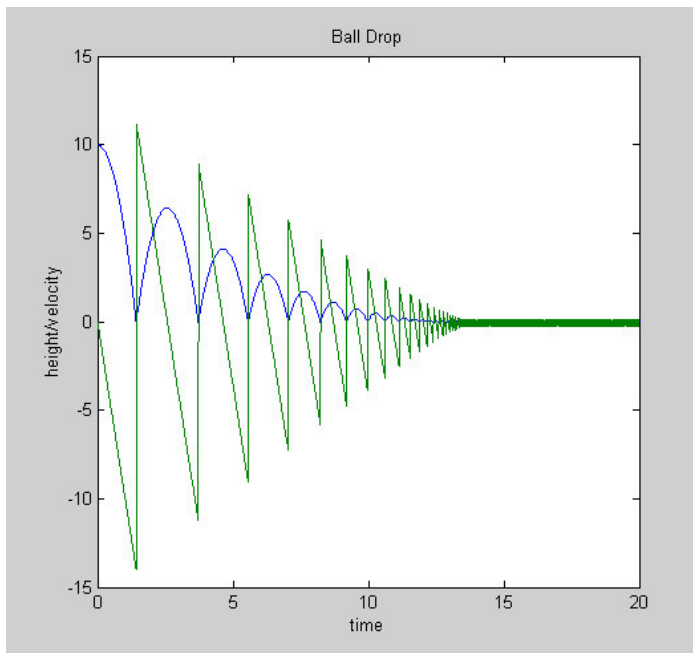
g=-9.8;
e=0.8;
t=[0:0.01:tf];

t0=0;
j=0;
for i=1:length(t)
    z(i) = 0.5*g*(t(i)-t0)^2 + v0*(t(i)-t0) + z0;
    v(i) = g*(t(i)-t0) + v0;

    if z(i) < 0           %when the ball goes below floor level,
        j = j+1;         %reverse the sign of the velocity and
        v0 = -e*v(i);    %reduce the velocity by the coefficient of
        z0 = 0;          %restitution.  add the impact times to a vector.
        t0 = t(i);       %t0 is now the last impact time.
        impact(j) = t(i);
    end
end

plot(t,z,t,v),title('Ball Drop'),xlabel('t'),ylabel('height/velocity')

>> [z,v,im] = drop(10,0,20);
```



```
>> im(1:10)'
```

```
ans =
```

```
1.4300  
3.7200  
5.5600  
7.0400  
8.2300  
9.1900  
9.9700  
10.6100  
11.1300  
11.5500
```

now I just found my first 10 approximations for the impact time. This comes in handy now for parts B3 and beyond.

B3) the first collision time is guessed at 1.43. at this time, the ball bounces back up from the floor. Now, I shall use the `fzero` function to find  $t_1$  more precisely. It should work with a function filename as argument, but here we will just have to input the dynamics as an inline function.

```
>> fzero(inline('-.5*9.8*t.^2+10'),1.43)
```

```
ans =
```

```
1.4286
```

B4) the velocity at  $t_1=1.4286$  is  $-9.8*1.4286 = -14\text{m/s}$ .

B5) the coefficient of restitution makes the outgoing velocity =  $14*.8 = 11.2\text{m/s}$ . this can be seen on the plot above, as well.

B6) running the program again with the new initial conditions of  $z_0 = 0$  and  $v_0 = 11.2$ , we find that the next impact time is at 3.7186 sec, as predicted before.

```
>> [z,v,im]=drop(0,11.2,20);
```

```
>> im(1)+1.4286
```

```
ans =
```

```
3.7186
```

and then done properly,

```
>> fzero(inline('-0.5*9.8*t.^2+11.2*t'),2.3)
```

```
ans =
```

```
2.2857
```

```
>> ans+1.4286
```

```
ans =
```

```
3.7143
```

the velocity at this time is  $-9.8 \cdot 2.2857 + 11.2 = -11.1999$  m/s. good to see that energy was conserved..... after the rebound,  $v = 0.8 \cdot 11.2 = 8.96$  m/s. and so on, through all of the bounces. Ta-da, we're done. Por fin, gracias a Dios.