Week 4

Contents

Question 1

```
x=[1.2,1.29,1.3,1.31,1.40];
fx=3*x.*exp(x)-cos(x); %Creates the f(x) for the x points. This is the same
                      % as the table that was provided in the question
dxfx=3*exp(x)+3*x.*exp(x)+sin(x); %First dirivative w.r.t. x
dxfx=dxfx(3); %Only keep the third value. ie we just needed this at x=1.3
disp("f'(1.3)=") %I use disp to output my results
disp(dxfx)       % Prints the result
h=0.1;  %Sets h for the first approximation.
```

```
f'(1.3)=
  26.281705191989978
```

First way to calculate \frac{d}{dx} f(x) with h=0.1

This requires knowing the index of the values we want to take, in this case we are using the 5th value in the array and the 1st value.

```
dxfx_h1=(fx(5)-fx(1))/(2*h);
```

Find index of a x value

The following define a function x_i(x,v,e) that finds the index of the x value within an epsilon of the value v. (ie equal within some error term)

```
e=0.0001;
x_i=@(x,v,e)find(abs(x - v)<=e);
```

Second way to calculate \frac{d}{dx} f(x) with h=0.1

This line to calculate the dxfx_h1 returns the same values as above. having to specify the index, it finds the index of the x value specified. The previous function worked. But this will make this easier to run again for any value of x.

```
dxFun_h=@(h)(fx(x_i(x,1.3+h,e))- fx(x_i(x,1.3-h,e)))/(2*h);
```

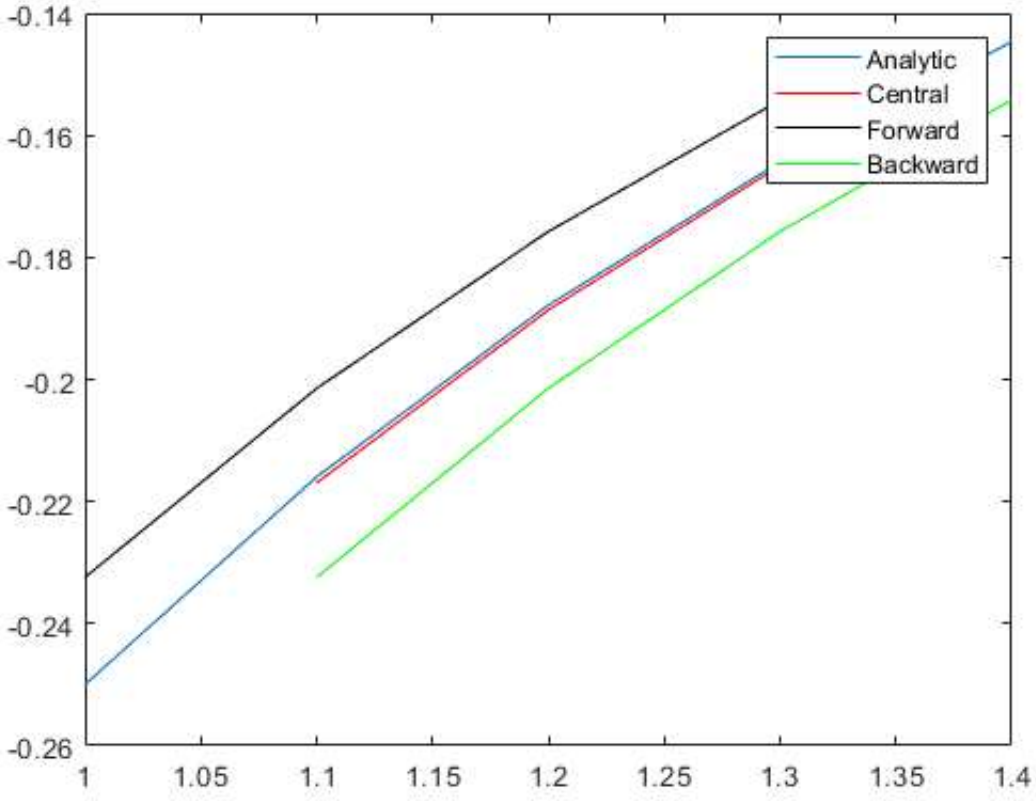
```
dxfx_h1 = dxFun_h(h);

disp("f'(1.3) is approx.    (h=0.1)")
disp(dxfx_h1)
disp("Which has an error of:")
disp(abs(dxfx_h1-dxfx))
h=0.01;
dxfx_h2 = dxFun_h(h); %Easy to rerun this since we defined a function above
disp("f'(1.3) is approx.    (h=0.01)")
disp(dxfx_h2)
disp("Which has an error of:")
disp(abs(dxfx_h2-dxfx))
```

```
f'(1.3) is approx.    (h=0.1)
  26.359047752362475

Which has an error of:
   0.077342560372497

f'(1.3) is approx.    (h=0.01)
  26.282478037329859

Which has an error of:
     7.728453398812007e-04
```

Internet is your friend.

If you use someones code online in your assignment, please cite your source and have changed it enough that you have demonstrated you understand its use. https://www.mathworks.com/matlabcentral/answers/213823-forward-backward-and-central-differences

```
Fun = @(x)((x)+1).^-2;
dFun = @(x) -2*(1 + x).^-3;
x=[1 1.1 1.2 1.3 1.4];
F=Fun(x);
h=x(2)-x(1);
xCentral=x(2:end-1);
dFCenteral=(F(3:end)-F(1:end-2))/(2*h);
xForward=x(1:end-1);
dFForward=(F(2:end)-F(1:end-1))/h;
xBackward=x(2:end);
dFBackward=(F(2:end)-F(1:end-1))/h;
plot(x,dFun(x));
hold on
plot(xCentral,dFCenteral,'r')
plot(xForward,dFForward,'k');
plot(xBackward,dFBackward,'g');
legend("Analytic",'Central','Forward','Backward')
```



Published with MATLAB® R2018a