HW 4 Key

Table of Contents

Problem 21.13

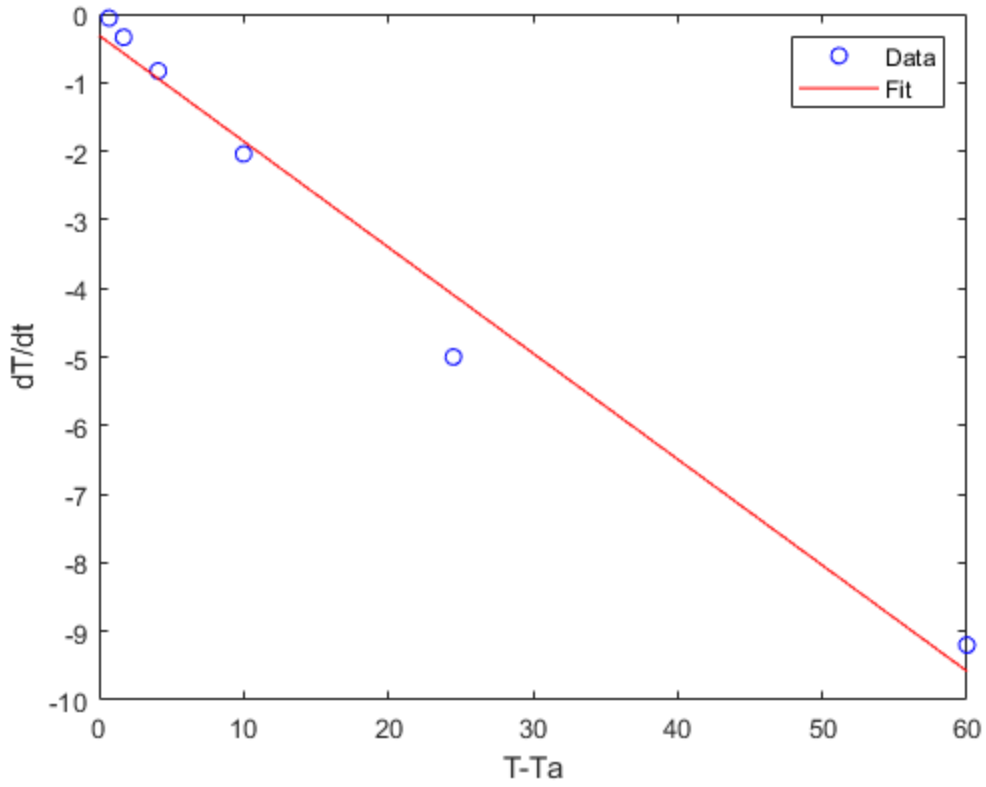
```
clc; clear all; close all; format compact;

%Declaring data
t = [0 2 4 6 8 10 12 14 16];
x = [0 0.7 1.8 3.4 5.1 6.3 7.3 8.0 8.4];

% (a) Centered finite-difference
fprintf('Part a (centered)\n')
fprintf('Velocity: %.4fm/s\n',nonzeros(deriv1O2(t(5:7),x(5:7),'c')))
fprintf('Acceleration: %.4fm/
s^2\n',nonzeros(deriv2O2(t(5:7),x(5:7),'c')))

% (b) Forward finite-difference
fprintf('\nPart b (forward)\n')
fprintf('Velocity: %.4fm/2\n',deriv1O2(t(6:8),x(6:8),'f'))
fprintf('Acceleration: %.4fm/s^2\n',deriv2O2(t(6:9),x(6:9),'f'))

% (c) Backward finite difference
fprintf('\nPart c (backward)\n')
fprintf('Velocity: %.4fm/2\n',nonzeros(deriv1O2(t(4:6),x(4:6),'b')))
fprintf('Acceleration: %.4fm/
s^2\n',nonzeros(deriv2O2(t(3:6),x(3:6),'b')))
```

```
Part a (centered)
Velocity: 0.5500m/s
Acceleration: -0.0500m/s^2

Part b (forward)
Velocity: 0.5750m/2
Acceleration: -0.0750m/s^2

Part c (backward)
Velocity: 0.4750m/2
Acceleration: -0.2750m/s^2
```

Problem 21.28

```
clc; clear all; close all; format compact;

%Declaring data
t = [0 5 10 15 20 25];
T = [80 44.5 30 24.1 21.7 20.7];
h = 5;
Ta = 20;

%Putting together composite data using forward, centered, then
 backward
%methods. Note the nonzeros() commands are to remove the extraneous
 zeros
%that are generated when you start indexing an array at i > 1.
dT = [deriv1O2(t(1:3),T(1:3),'f')
      nonzeros(deriv1O2(t,T,'c'))
      nonzeros(deriv1O2(t(4:6),T(4:6),'b'))]';


%Plotting data
plot(T-Ta,dT,'bo')
hold on
RgFit = polyfit(T-Ta,dT,1); %fitting data
fplot(@(x) RgFit(2)+RgFit(1)*x,[0,60],'r')
legend('Data','Fit')
xlabel('T-Ta')
ylabel('dT/dt')

%Calculating and displaying k
k = -RgFit(1)
```

```
k =
   0.154504746324175
```



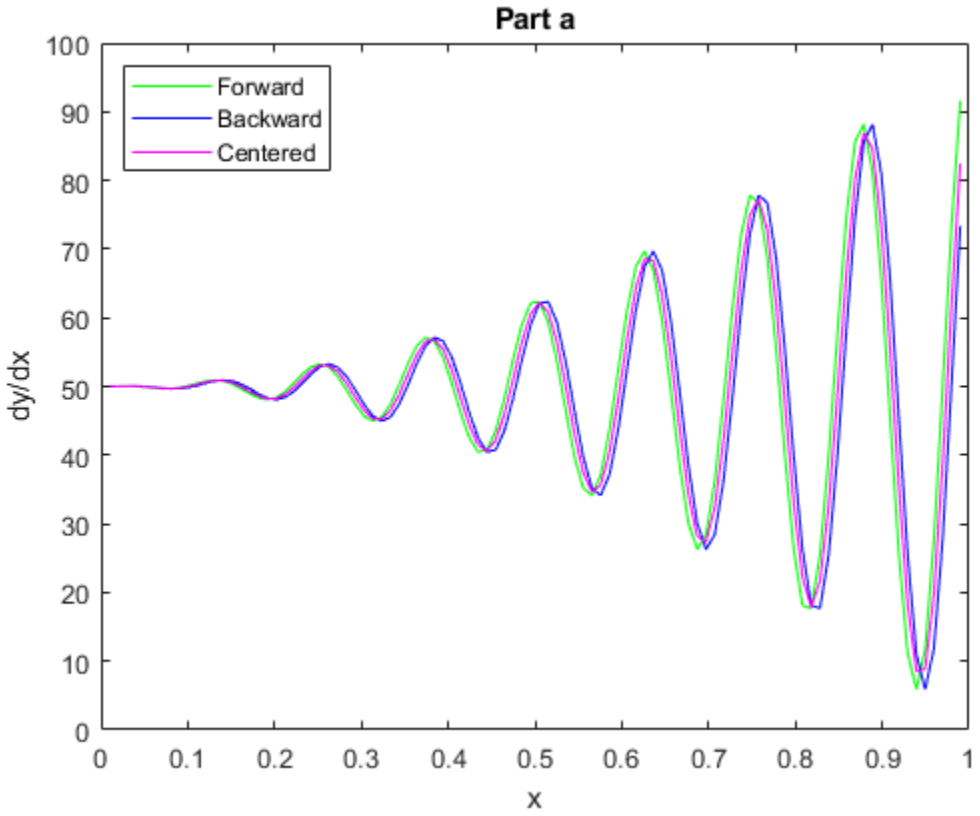
Additional Problem

```
clc; clear all; close all; format compact;

%Declaring functions MUST BE RUN BEFORE ANY FOLLOWING SECTIONS
f = @(x) sin(50.*x).*x.^2 + 50*x;
fprime = @(x) 50*x.^2.*cos(50.*x) + sin(50.*x).*x*2+50;
```

(a)

```
%Creating x array
x = linspace(0,1,100);
xx = x(2:end-1);
y = f(x);

%Solving derivatives
dyF = deriv1O1(x(2:end),y(2:end),'f');
dyB = deriv1O1(x(1:end-1),y(1:end-1),'b');
dyB(1) = [];
dyC = deriv1O1(x,y,'c');
dyC(1) = [];

%Plotting results
plot(xx,dyF,'g')
hold on
```

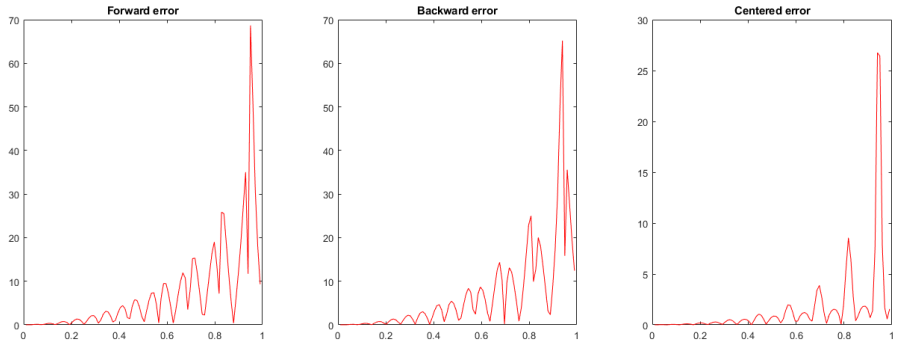
```
plot(xx,dyB,'b')
plot(xx,dyC,'m')
hold off
legend('Forward','Backward','Centered','Location','northwest')
title('Part a')
xlabel('x')
ylabel('dy/dx')
```



(b)

```
close all; clc;

%Solving errors
fP = fprime(xx);
errF = abs((fP-dyF)./fP)*100;
errB = abs((fP-dyB)./fP)*100;
errC = abs((fP-dyC)./fP)*100;

%Plotting results
figure('Position',[0 0 1500 500])
subplot(1,3,1)
plot(xx,errF,'r')
title('Forward error')

subplot(1,3,2)
plot(xx,errB,'r')
```

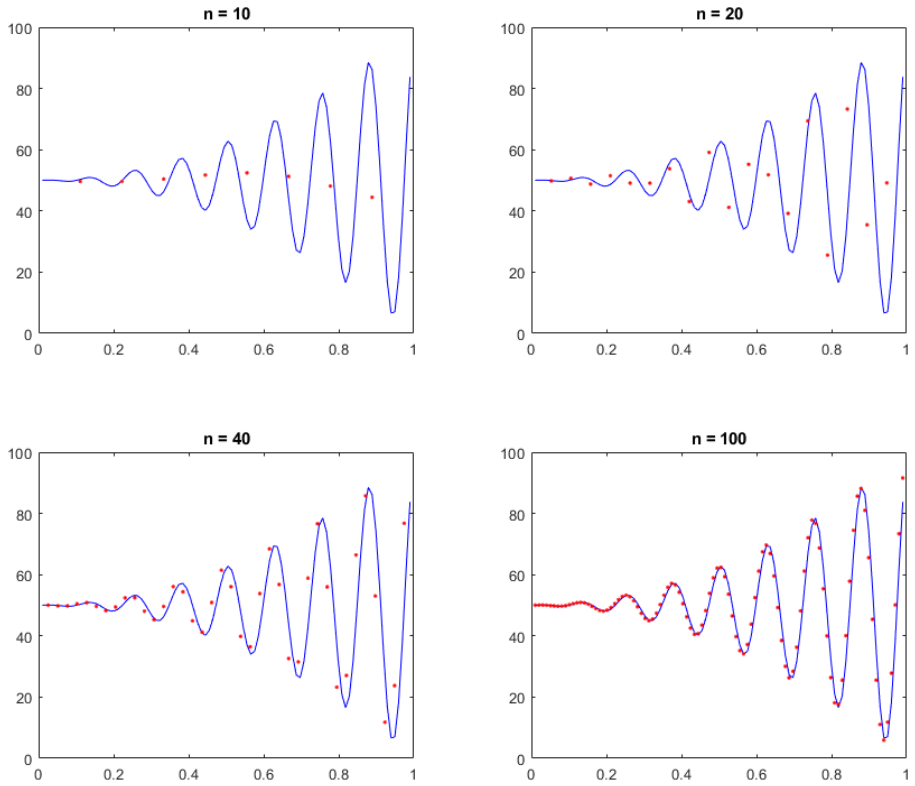
```
title('Backward error')

subplot(1,3,3)
plot(xx, errC,'r')
title('Centered error')
```



(c)

```
close all; clc;

%Creating all necessary x arrays, solving derivatives
x = linspace(0,1,10); xx10 = x(2:end-1); y = f(x);
dyF10 = deriv1O1(x(2:end),y(2:end),'f');

x = linspace(0,1,20); xx20 = x(2:end-1); y = f(x);
dyF20 = deriv1O1(x(2:end),y(2:end),'f');

x = linspace(0,1,40); xx40 = x(2:end-1); y = f(x);
dyF40 = deriv1O1(x(2:end),y(2:end),'f');

x = linspace(0,1,100); xx100 = x(2:end-1); y = f(x);
dyF100 = deriv1O1(x(2:end),y(2:end),'f');

%plotting results
figure('Name','Part c','Position',[100 100 1000 800])
subplot(2,2,1)
plot(xx,fP,'b')
hold on
plot(xx10,dyF10,'r.')
title('n = 10')
subplot(2,2,2)
plot(xx,fP,'b')
hold on
plot(xx20,dyF20,'r.')
title('n = 20')
subplot(2,2,3)
plot(xx,fP,'b')
hold on
plot(xx40,dyF40,'r.')
title('n = 40')
```

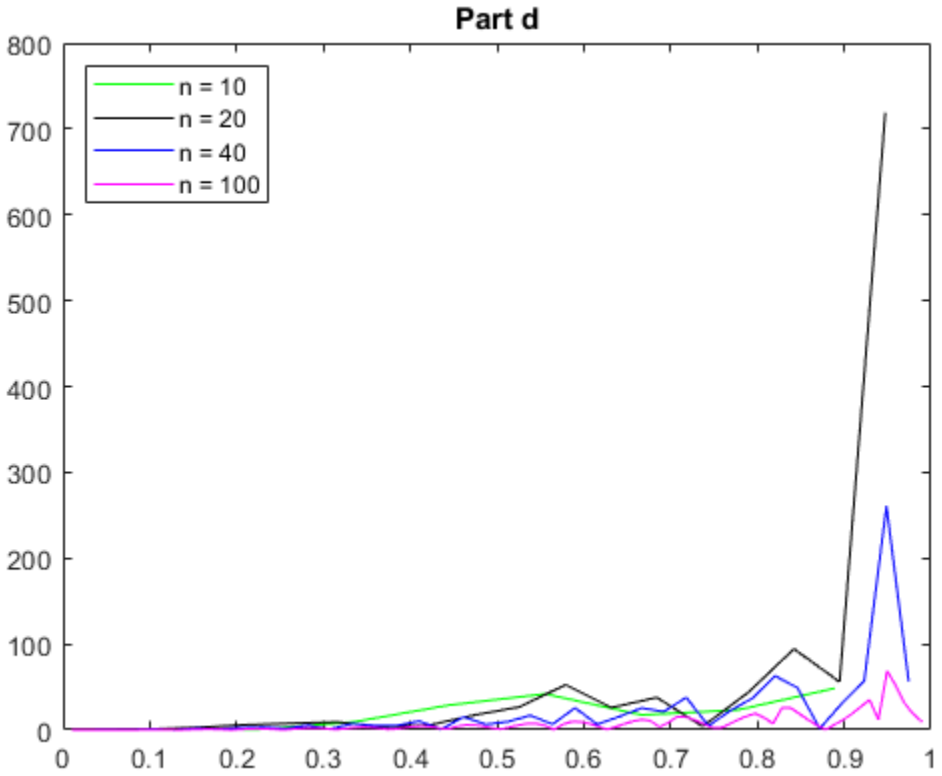
```
subplot(2,2,4)
plot(xx,fP,'b')
hold on
plot(xx100,dyF100,'r.')
title('n = 100')
```



(d)

```
close all; clc;

%Finding exact solutions
yp10 = fprime(xx10);
yp20 = fprime(xx20);
yp40 = fprime(xx40);
yp100 = fprime(xx100);

%Finding errors
err10 = abs((yp10-dyF10)./yp10)*100;
err20 = abs((yp20-dyF20)./yp20)*100;
err40 = abs((yp40-dyF40)./yp40)*100;
err100 = abs((yp100-dyF100)./yp100)*100;

%Plotting results
figure
```

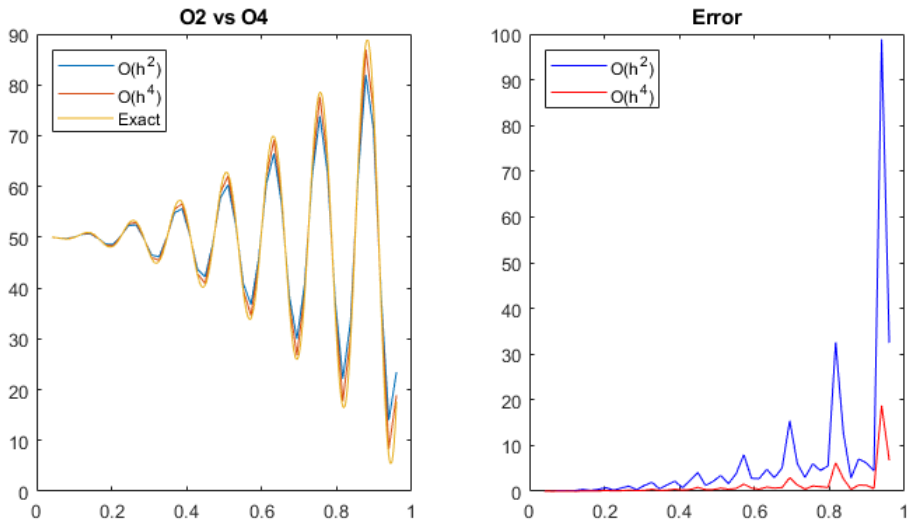
```
plot(xx10,err10,'g')
hold on
plot(xx20,err20,'k')
plot(xx40,err40,'b')
plot(xx100,err100,'m')
legend('n = 10','n = 20','n = 40','n = 100','Location','northwest')
title('Part d')
```



(e)

```
close all; clc;

%Creating x arrays
n = 50;
x = linspace(0,1,n); xx = x(3:end-2); y = f(x);

%Solving for second order accurate derivative
for i = 3:length(y)-2
    dy50_O2(i) = (y(i+1)-y(i-1)) / (2*(1/(n-1)));
end
dy50_O2(1:2) = [];

%Solving for 4th order accurate derivative
for i = 3:length(y)-2
    dy50_O4(i) = (-y(i+2)+8*y(i+1)-8*y(i-1)+y(i-2))/(12*(1/(n-1)));
end
```

```
dy50_O4(1:2) = [];

%Finding errors
dyTrue = fprime(xx);
errO2 = abs((dyTrue-dy50_O2)./dyTrue)*100;
errO4 = abs((dyTrue-dy50_O4)./dyTrue)*100;

%Plotting results
figure('Position',[100 100 800 400])
subplot(1,2,1)
plot(xx,dy50_O2)
hold on
plot(xx,dy50_O4)
fplot(fprime,[xx(1),xx(end)])
hold off
legend('O(h^2)','O(h^4)','Exact','Location','Northwest')
title('O2 vs O4')
subplot(1,2,2)
plot(xx,errO2,'b')
hold on
plot(xx,errO4,'r')
hold off
legend('O(h^2)','O(h^4)','Location','Northwest')
title('Error')
```



Published with MATLAB® R2018b