5.4.3

```
%nodes
t1 = [0.25 0.35 0.42 0.5 0.60]';
t2 = [-1.0 5.0 0.6 1.20 0.53]';
t3 = [0.25 0.35 0.42 0.75 0.829]';
t4 = [-1.0 1.0 0.60 0.7 0.50]';
f = @(x) exp(-x);
dfdx = @(x) -exp(-x);
ddfdx = @(x) exp(-x);
dddfdx = @(x) -exp(-x);
ddddfdx = @(x) exp(-x);
exact_values = [dfdx(0.5), ddfdx(0.5) dddfdx(0.5), ddddfdx(0.5)];
fd_values = 0*exact_values;
errors = fd_values*0;

%1st derivative
w1= fd_weights1(t1-0.5, 1);
fd_values(1)= w1'*f(t1);
errors(1) = exact_values(1) - fd_values(1);


%2nd derivative
w2= fd_weights1(t2-0.5, 2);
fd_values(2)= w2'*f(t2);
errors(2) = exact_values(2) - fd_values(2);

%3rd derivative
w3= fd_weights1(t3-0.5, 3);
fd_values(3)= w3'*f(t3);
errors(3) = exact_values(3) - fd_values(3);
%4th derivative
w4= fd_weights1(t4-0.5, 4);
fd_values(4)= w4'*f(t4);
errors(4) = exact_values(4) - fd_values(4);
table(fd_values, errors)
```

```
ans =

  1×2 table

                     fd_values
  errors
    ____________________________________________
 _______________________________________________

    -0.62828     0.72216    -0.58877     0.72941    0.021746
 -0.11563    -0.017759     -0.12288
```

```
%5.4.5
%a
```

```
%nodes
t = [-0.07, -0.2, 0.03, 0.08, 0.13]';

f = @(x) tan(2*x);
ddfdx = @(x) 8*(sec(2*x))^2*tan(2*x);

exact_value = ddfdx(0.3);

%2nd derivative
w= fd_weights1(t-0.3, 2);
fd_value= w'*f(t);
fd_value
exact_value

%b
%nodes
t = [0.65, 0.7, 0.75, 0.8, 0.85]';
exact_value = ddfdx(0.75);

%2nd derivative
w= fd_weights1(t-0.75, 2);
fd_value= w'*f(t);
fd_value
exact_value
```

```

fd_value =

  337.0065


exact_value =

    8.0347


fd_value =

   3.9744e+03


exact_value =

   2.2545e+04
```

5.5.1

```
h = 2.^(-1:-1:-7);
x = pi/7;
%point around which the centered-finite difference formula is to be
 applied
f = cos(x);
dfdx = -sin(x);
```

```
n=length(h);
bfdxx=zeros(n);

exact_value = dfdx;

error = 0*h;

for i =1:length(h)
    xp(i) = x+h(i);
    xn(i) = x-h(i);
    %values at x+-h(i)
    fp(i) = cos(xp(i));
    fn(i) = cos(xn(i));
    %f(x) values at f(x+-h(i))
    cfdx(i) = (fp(i)-fn(i))/ (2*h(i));
    %centered finite difference formula
    cfdxx=cfdx(i);
    error(i) = cfdx(i) - exact_value;
end
cfdx
error
exact_value

loglog(h, error);
%2nd order convergence
hold on, loglog(h, [h.^2], '--');
function w = fd_weights1(t,m)
    r = length(t)-1;
    w=zeros(size(t));

    for k=0:r
        w(k+1) = weight(t,m,r,k);
    end
end

function c =weight(t,m,r,k)
    if(m<0) || (m>r)
        c=0;
    elseif (m==0) && (r==0)
        c=1;
    else
        if k<r
            c= (t(r+1)*weight(t,m,r-1,k) - m*weight(t,m-1,r-1,k))/(t(r
+1)-t(k+1));
        else
            beta = prod(t(r)-t(1:r-1)) / prod(t(r+1)-t(1:r));
            c = beta*(m*weight(t,m-1,r-1,r-1)-
t(r)*weight(t,r,r-1,r-1));
        end
    end
end
```

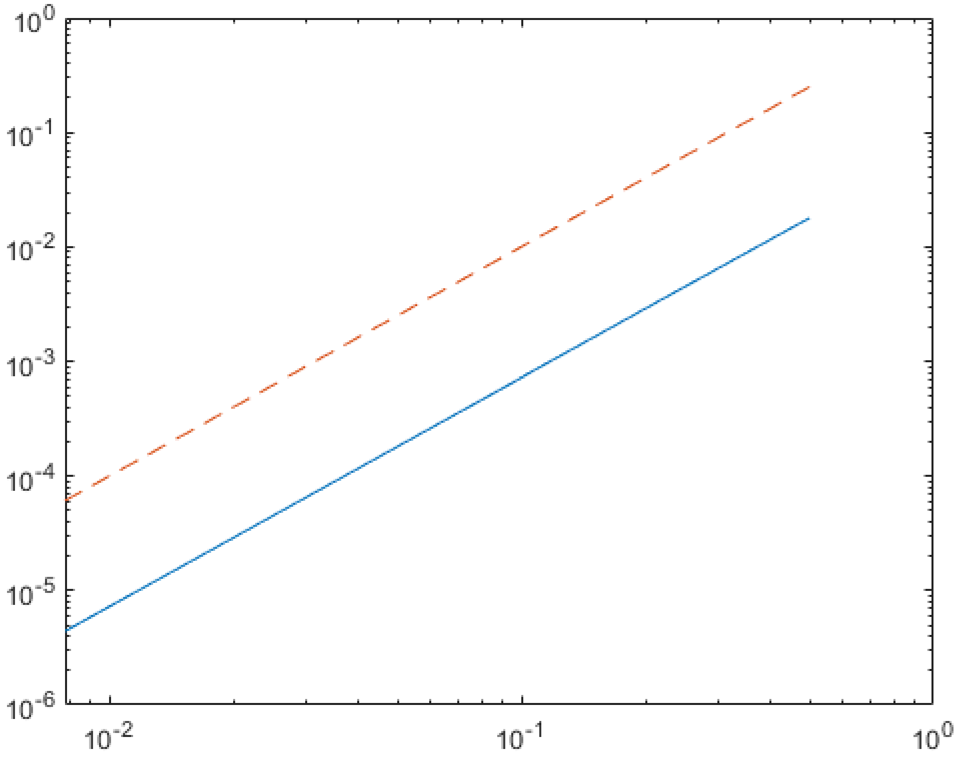
```
cfdx =
   -0.4160   -0.4294   -0.4328   -0.4336   -0.4338   -0.4339   -0.4339


error =

    0.0179    0.0045    0.0011    0.0003    0.0001    0.0000    0.0000


exact_value =

   -0.4339
```



Published with MATLAB® R2018a