Universidad Industrial de Santander, Colombia
Numerical Analysis, 2020-2
Henry Arguello
February 10, 2021

# Lab 8. Numerical Differentiation

## 1 Instructions

- Make a **pdf** report including the solution to each point of the practice with name *Lab8_name_lastname.pdf*.
- Send the report and all created files in a rar or zip file with name *Lab8_name_lastname.rar* in the Moodle.
- You are allowed to use internet, notes, and .m files that you have created before.

## 2 Purposes

- To apply numerical differentiation methods.
- To implement numerical differentiation methods in Matlab.

## 3 Practice

### 3.1 Applying

(1.5 points) Let $f(x) = \cos(x)$ and carry eight or nine decimal places.

a) Use the equation (6) with $h = 0.05$ to approximate $f''(1)$

b) Use la equation (6) with $h = 0.01$ to approximate $f''(1)$

c) Use la equation (12) with $h = 0.1$ to approximate $f''(1)$

d) What answer, (a), (b) or (c), is most accurate?

(1.0 point) The following Table presents some values of $f(x)$ rounded to six decimal places. Use the numerical differentiation forward-formula of order $\mathcal{O}(h^2)$ with $h = 0.05$ to find an approximation of $f''(-3.5)$ by assuming $f''(x) = f''(-x + 0.5)$. *Hint.* Recall that $f''(x_0) \approx \frac{2f_0 - 5f_1 + 4f_2 - f_3}{h^2}$.

| $f(x)$ | 0.697448 | 0.673210 | 0.650527 | 0.629492 | 0.610192 | 0.592710 | 0.577125 |
|---|---|---|---|---|---|---|---|
| $x$ | 3.85 | 3.90 | 3.95 | 4.00 | 4.05 | 4.10 | 4.15 |

### 3.2 Implementing

(2.5 points) Use the Program 1 to approximate the derivatives of each of the following functions at the given value of $x$. Approximations should be accurate to 13 decimal places.

*Note.* It may be necessary to change the values of max1 and the initial value of h in the program.

a) $f(x) = 60x^{45} - 32x^{33} + 233x^5 - 47x^2 - 77;\ x = 1/\sqrt{3}$

b) $f(x) = \tan\left(\cos\left(\dfrac{\sqrt{5}+\sin(x)}{1+x^2}\right)\right);\ x = \dfrac{1+\sqrt{5}}{3}$

c) $f(x) = \sin(x^3 - 7x^2 + 6x + 8);\ x = \frac{1-\sqrt{5}}{2}$

# 4 Appendices

- Equation (6)
  $$f''(x_0) = \frac{f_1 - 2f_0 + f_{-1}}{h^2} + E(f,h)$$
  $$E(f,h) = \frac{4\varepsilon}{h^2} + \frac{h^2 f^{(4)}(c)}{12}$$
- Equation (12)
  $$f''(x_0) = \frac{-f_2 + 16f_1 - 30f_0 + 16f_{f-1} - f_{-2}}{12h^2} + E(f,h)$$
  $$E(f,h) = \frac{16\varepsilon}{3h^2} + \frac{h^4 f^{(6)}(c)}{90}$$

Listing 1: Program 1 (Differentiation using Limits)

```
function [L,n]=difflim(f,x,toler)

%Input  - f is the function input as a string 'f'
%       - x is the differentiation point
%       - toler is the desired tolerance
%Output - L=[H' D' E']: H is the vector of step sizes
%                       D is the vector of approximate derivatives
%                       E is the vector of error bounds
%       - n is the coordinate of the "best approimation"

% NUMERICAL METHODS: MATLAB Programs
%(c) 1999 by John H. Mathews and Kurtis D. Fink
%To accompany the textbook:
%NUMERICAL METHODS Using MATLAB,
%by John H. Mathews and Kurtis D. Fink
%ISBN 0-13-270042-5, (c) 1999
%PRENTICE HALL, INC.
%Upper Saddle River, NJ 07458

max1=15;
h=1/10;
H(1)=h;
D(1)=(feval(f,x+h)-feval(f,x-h))/(2*h);
E(1)=0;
R(1)=0;

for n=1:2
   h=h/10;
   H(n+1)=h;
   D(n+1)=(feval(f,x+h)-feval(f,x-h))/(2*h);
   E(n+1)=abs(D(n+1)-D(n));
   R(n+1)=2*E(n+1)*(abs(D(n+1))+abs(D(n))+eps);
end

n=2;

while((E(n)>E(n+1))&(R(n)>toler))&n<max1
   h=h/10;
   H(n+2)=h;
   D(n+2)=(feval(f,x+h)-feval(f,x-h))/(2*h);
   E(n+2)=abs(D(n+2)-D(n+1));
   R(n+2)=2*E(n+2)*(abs(D(n+2))+abs(D(n+1))+eps);
   n=n+1;
end

n=length(D)-1;
L=[H' D' E'];
```