Comparing Methods of First Derivative Approximation

Forward, Backward and Central Divided Difference

Ana Catalina Torres, Autar Kaw

University of South Florida

United States of America

kaw@eng.usf.edu

Introduction

This worksheet demonstrates the use of Maple to compare the approximation of first order derivatives using three different methods. Each method uses a point \( h \) ahead, behind or both of the given value of \( x \) at which the first derivative of \( f(x) \) is to be found.

Forward Difference Approximation (FDD)

\[
f'(x) \approx \frac{f(x + h) - f(x)}{h}
\]

Backward Difference Approximation (BDD)

\[
f'(x) \approx \frac{f(x) - f(x - h)}{h}
\]

Central Difference Approximation (CDD)

\[
f'(x) \approx \frac{f(x + h) - f(x - h)}{2h}
\]

Initialization

\[
> \text{restart;} \\
> \text{with (plots);} \\
\]

Section 1: Input

The following simulation approximates the first derivative of a function using different methods of approximation (FDD, BDD, CDD). The user inputs are

a) function, \( f(x) \)

b) point at which the derivative is to be found, \( x_v \)

c) starting step size, \( h \)

d) number of times user wants to halve the step size, \( n \)

The outputs include

a) approximate values of the first derivative at the point and initial step size given using different types of approximation

b) exact value

c) absolute relative true error, absolute relative approximate error, and number of at least correct significant digits in the solution as a function of step size.
Section 2: Procedure

The following procedure estimates the solution of first order derivate of an equation at a point \( x_v \) using different methods of approximation.

\( f(x) = \text{function} \)

\( x_v = \text{value at which the solution is desired} \)

\( h = \text{step size value} \)

\( n = \text{number of times step size is halved} \)

\( > \) Forward Divided Difference Procedure

\[
FDD := \text{proc}(f, x_v, h) \\
\text{local} \ deriv : \\
\text{deriv} := (f(x_v + h) - f(x_v)) / h \\
\text{return} (\text{deriv}) : \\
\text{end proc}:
\]

\( > \) Backward Divided Difference Procedure

\[
BDD := \text{proc}(f, x_v, h) \\
\text{local} \ deriv : \\
\text{deriv} := (f(x_v) - f(x_v - h)) / h \\
\text{return} (\text{deriv}) : \\
\text{end proc}:
\]

\( > \) Central Divided Difference Precedure

\[
CDD := \text{proc}(f, x_v, h) \\
\text{local} \ deriv : \\
\text{deriv} := (f(x_v + h) - f(x_v - h)) / (2 \cdot h) \\
\text{return} (\text{deriv}) :
\]
end proc:

### Section 3: Calculation

The exact value $E_v$ of the first derivative of the equation:

First, using the `diff` command the solution is found. In a second step, the exact value of the derivative is shown.

\[
\begin{align*}
(y(x) &= f(x); \\
&= \quad y(x) = x e^{2x} \quad \text{(5.1)} \\
&= \quad Soln := diff(f(x), x); \\
&= \quad Soln := e^{2x} + 2x e^{2x} \quad \text{(5.2)} \\
&= \quad Ev := evalf(subs(x = xv, Soln)); \\
&= \quad Ev := 26828.62188 \quad \text{(5.3)}
\end{align*}
\]

The next loop calculates the following:

- **$Av$:** Approximate value of the first derivative using various first derivative approximation methods by calling the procedures "FDD", "BDD", and "CDD"
- **$Ev$:** Exact value of the second derivative
- **$et$:** Absolute relative true percentage error
- **$ea$:** Absolute relative approximate percentage error
- **$Sig$:** Least number of correct significant digits in an approximation

\[
\begin{align*}
&> \text{for } i \text{ from 0 by 1 to } n-1 \text{ do} \\
&\quad N[i] := 2^i; \\
&\quad H[i] := \frac{h}{N[i]}; \\
&\quad AvFDD[i] := FDD(f, xv, H[i]); \\
&\quad AvBDD[i] := BDD(f, xv, H[i]); \\
&\quad AvCDD[i] := CDD(f, xv, H[i]); \\
&\quad etFDD[i] := \text{abs}\left(\frac{Ev - AvFDD[i]}{Ev}\right) \cdot 100; \\
&\quad etBDD[i] := \text{abs}\left(\frac{Ev - AvBDD[i]}{Ev}\right) \cdot 100; \\
&\quad etCDD[i] := \text{abs}\left(\frac{Ev - AvCDD[i]}{Ev}\right) \cdot 100; \\
&\quad \text{if } (i > 0) \text{ then} \\
&\quad \quad eaFDD[i] := \text{abs}\left(\frac{AvFDD[i] - AvFDD[i-1]}{AvFDD[i]}\right) \cdot 100; \\
&\quad \quad eaBDD[i] := \text{abs}\left(\frac{AvBDD[i] - AvBDD[i-1]}{AvBDD[i]}\right) \cdot 100; \\
&\quad \quad eaCDD[i] := \text{abs}\left(\frac{AvCDD[i] - AvCDD[i-1]}{AvCDD[i]}\right) \cdot 100; \\
&\quad \quad SigFDD[i] := \text{floor}\left(2 - \log_{10}\left(\frac{eaFDD[i]}{0.5}\right)\right); \\
&\quad \text{end if;}
\end{align*}
\]
The loop halves the value of the step size \( n \) times. Each time, the approximate values of the first derivative are calculated and saved in different vectors depending on the method of approximation. The approximate error is calculated after at least two approximate values of the first derivative have been saved. The number of significant digits is calculated and written as the lowest real number for each method. If the number of significant digits calculated is less than zero, then it is shown as zero.

### Section 4: Spreadsheet

The next table shows the step size value, exact value, approximate value, and the least number of correct significant digits in an approximation as a function of the step size value for each method of approximation.

```plaintext
with(Spread):
    tableoutput := CreateSpreadsheet("FDD, BDD and CDD Comparison"):
    SetCellFormula(tableoutput, 1, 2, "Step Size"):
    SetCellFormula(tableoutput, 1, 3, "Exact Value"):
    SetCellFormula(tableoutput, 1, 4, "Approx Value FDD"):
    SetCellFormula(tableoutput, 1, 5, "Abs Rel True Error FDD"):
    SetCellFormula(tableoutput, 1, 6, "Approx Value BDD"):
    SetCellFormula(tableoutput, 1, 7, "Abs Rel True Error BDD"):
    SetCellFormula(tableoutput, 1, 8, "Approx Value CDD"):
    SetCellFormula(tableoutput, 1, 9, "Abs Rel True Error CDD");
```

```plaintext
for i from 0 by 1 to n-1 do
    SetCellFormula(tableoutput, i+2, 1, i+1):
    SetCellFormula(tableoutput, i+2, 2, evalf(H[i])):
    SetCellFormula(tableoutput, i+2, 3, evalf(Ev)):
    SetCellFormula(tableoutput, i+2, 4, evalf(AvFDD[i])):
    SetCellFormula(tableoutput, i+2, 5, evalf(etFDD[i])):
    SetCellFormula(tableoutput, i+2, 6, evalf(AvBDD[i])):
    SetCellFormula(tableoutput, i+2, 7, evalf(etBDD[i])):
```
SetCellFormula(tableoutput, i+2, 8, evalf(AvCDD[i]));
SetCellFormula(tableoutput, i+2, 9, evalf(etCDD[i]));
end do:
EvaluateSpreadsheet(tableoutput);

\[
\begin{array}{cccccccc}
\hline
& B & C & D & E & F & G & H \\
\hline
1 & Step Size & Exact Value & Approx Value FDD & Abs Rel True Error FDD & Approx Value BDD & Abs Rel True Error BDD & Approx Value CDD & Abs Rel True Error CDD \\
2 & 0.2 & 26828.62188 & 25.87020720 & 19.28978746 & 27711.33985 & 3.290209888 & \\
3 & 0.1 & 26828.62188 & 11.97263755 & 5.35729265 & 26883.30500 & 0.2038238127 & \\
4 & 0.05 & 26828.62188 & 5.764936891 & 3.57258352 & 26832.03800 & 0.0509336535 & \\
5 & 0.025 & 26828.62188 & 2.822405563 & 1.376241693 & 26832.03800 & 0.0127331170 & \\
6 & 0.0125 & 26828.62188 & 0.6976280650 & 0.6912717352 & 26832.03800 & 0.003183614886 & \\
\hline
\end{array}
\]

Section 5: Graphs

The following graphs show the approximate solution, absolute relative true error, absolute relative approximate error and least number of significant digits as a function of step size. Each graph displays the resaults of each of the methods of approximation.

```
> data := [seq([H[i], AvFDD[i]], i = 0 .. n-1)]:
data1 := [seq([H[i], AvBDD[i]], i = 0 .. n-1)]:
data2 := [seq([H[i], AvCDD[i]], i = 0 .. n-1)]:
plot([data, data1, data2], x=H[0]..H[n-1], color=[cyan, maroon, blue], thickness=2, title = "Approximate Solution of the First Derivative of a Function using different Methods of Approximation as a Function of Step Size", linestyle=[3, 2, 1], legend=[FDD, BDD, CDD], legendstyle=[location=right], labels = ["Step Size", "Approximate Value"], titlefont=[TIMES, BOLD, 12], labelfont=[TIMES, ROMAN, 12]);
data := [seq([H[i], etFDD[i]], i = 0 .. n-1)]:
data1 := [seq([H[i], etBDD[i]], i = 0 .. n-1)]:
data2 := [seq([H[i], etCDD[i]], i = 0 .. n-1)]:
plot([data, data1, data2], x=H[0]..H[n-1], color=[cyan, maroon, blue], thickness=2, title = "Absolute Relative True Percentage Error as a Function of Step Size", labels = ["Step Size", "Absolute Relative True Error"], linestyle=[3, 2, 1], legend=[FDD, BDD, CDD], legendstyle=[location=right], titlefont=[TIMES, BOLD, 12], labelfont=[TIMES, ROMAN, 12]);
data := [seq([H[i], eaFDD[i]], i = 0 .. n-1)]:
```
data1 := [seq([H[i], eaBDD[i]], i = 0 .. n - 1)];
data2 := [seq([H[i], eaCDD[i]], i = 0 .. n - 1)];
plot([data, data1, data2], x = H[0]..H[n - 1], color = [cyan, maroon, blue], thickness = 2, title = "Absolute Relative Approximate Percentage Error as a Function of Step Size", labels = ["Step Size", "Absolute Relative Approximate Error "])
linestyle = [3, 2, 1], legend = [FDD, BDD, CDD], legendstyle = [location = right], titlefont = [TIMES, BOLD, 12], labelfont = [TIMES, ROMAN, 12]);
data := [seq([H[i], SigFDD[i]], i = 0 .. n - 1)];
data1 := [seq([H[i], SigBDD[i]], i = 0 .. n - 1)];
data2 := [seq([H[i], SigCDD[i]], i = 0 .. n - 1)];
plot([data, data1, data2], x = H[0]..H[n - 1], color = [cyan, maroon, blue], thickness = 2, title = "Least Correct Significant Digits as a Function of Step Size", labels = ["Step size", "Least number of significant digits"])
linestyle = [3, 2, 1], legend = [FDD, BDD, CDD], legendstyle = [location = right], titlefont = [TIMES, BOLD, 12], labelfont = [TIMES, ROMAN, 12]);
Approximate Solution of the First Derivative of a Function using different Methods of Approximation as a Function of Step Size

Approximate Value

Step Size
Absolute Relative True Percentage Error as a Function of Step Size

Absolute Relative True Error

- **FDD**
- **BDD**
- **CDD**
Absolute Relative Approximate Percentage Error as a Function of Step Size

Absolute Relative Approximate Error

- **FDD**
- **BDD**
- **CDD**
Least Correct Significant Digits as a Function of Step Size

References
Numerical Differentiation of Continuous Functions.
See http://numericalmethods.eng.usf.edu/mws/gen/02def/

Questions
1. The velocity of a rocket is given by

   \[ v(t) = 2000 \cdot \ln \left( \frac{140000}{140000 - 2100 t} \right) - 9.8 t \]

   Use three different methods with a step size of 0.25 to find the acceleration at \( t=5 \) s. Compare with the exact answer and study the effect of the step size.

2. Look at the true error vs. step size data for problem 1. Do you see a relationship between the value of the true error and step size? Is this coincidental? Is it similar for Forward and Backward Divided Difference? Is it different for Central Divided Difference method?

Conclusions
The worksheet shows the nature of accuracy of the three different methods of finding the first
derivative of a continuous function. Forward and Backward Divided Difference methods exhibit similar accuracies as they are first order accurate, while central divided difference shows more accuracy as it is second order accurate.

Legal Notice: The copyright for this application is owned by the author(s). Neither Maplesoft nor the author are responsible for any errors contained within and are not liable for any damages resulting from the use of this material. This application is intended for non-commercial, non-profit use only. Contact the author for permission if you wish to use this application for-profit activities.