

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 Mathematica 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*)

$$f'(x) \cong \frac{f(x+h)-f(x)}{h}$$

Backward Difference Approximation (*BDD*)

$$f'(x) \cong \frac{f(x)-f(x-h)}{h}$$

Central Difference Approximation (*CDD*)

$$f'(x) \cong \frac{f(x+h)-f(x-h)}{2h}$$

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

- function, $f(x)$
- point at which the derivative is to be found, xv
- starting step size, h
- number of times user wants to halve the starting step size, n

The outputs include

- approximate values of the first derivative at the point and initial step size given using different types of approximation
- exact value
- true error, absolute relative true error, approximate error and absolute relative approximate error, number of at least correct significant digits in the solution as a function of step size.

Function $f(x)$:

```
f[x_] := x * Exp[2 * x];
```

Value of x at which $f'(x)$ is desired, xv

```
xv = 4.0;
```

Starting step size, h

```
h = 0.2;
```

Number of times step size is halved

```
n = 12.0;
```

This is the end of the user section. All the information must be entered before proceeding to the next section.

Section 2: Procedure

The following procedure estimates the solution of first derivative of an equation at a point xv using different methods of approximation.

$f(x)$ = function

xv = value at which the solution is desired

h = step size value

n = number of times step size is halved

Forward Divided Difference Procedure

```
FDD[f_, xv_, h_] := Module[{deriv},
  deriv = (f[xv + h] - f[xv]) / (h);
  deriv]
```

Backward Divided Difference Procedure

```
BDD[f_, xv_, h_] := Module[{deriv},
  deriv = (f[xv] - f[xv - h]) / (h);
  deriv]
```

Central Divided Difference Procedure

```
CDD[f_, xv_, h_] := Module[{deriv},
  deriv = (f[xv + h] - f[xv - h]) / (2 * h);
  deriv]
```

Section 3: Calculation

The exact value Ev 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.

```
f[x_]
```

```
e2 x_ - x_
```

```
f'[x_]
```

```
e2 x_ - x_ + 2 e2 x_ - x_
```

```
Ev = N[f'[xv]]
```

```
26 828.6
```

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

```

Do[
  Nn[i] = 2^i;
  H[i] = h / Nn[i];
  AVFDD[i] = FDD[f, xv, H[i]];
  AVBDD[i] = BDD[f, xv, H[i]];
  AVCDD[i] = CDD[f, xv, H[i]];
  etFDD[i] = Abs[(Ev - AVFDD[i]) / Ev] * 100.0;
  etBDD[i] = Abs[(Ev - AVBDD[i]) / Ev] * 100.0;
  etCDD[i] = Abs[(Ev - AVCDD[i]) / Ev] * 100.0;
  If[i > 0,
    eaFDD[i] = Abs[(AVFDD[i] - AVFDD[i - 1]) / AVFDD[i]] * 100.0;
    eaBDD[i] = Abs[(AVBDD[i] - AVBDD[i - 1]) / AVBDD[i]] * 100.0;
    eaCDD[i] = Abs[(AVCDD[i] - AVCDD[i - 1]) / AVCDD[i]] * 100.0;
    SigFDD[i] = Floor[(2 - Log[10, eaFDD[i] / 0.5])];
    SigBDD[i] = Floor[(2 - Log[10, eaBDD[i] / 0.5])];
    SigCDD[i] = Floor[(2 - Log[10, eaCDD[i] / 0.5])];
    If[SigFDD[i] < 0, SigFDD[i] = 0];
    If[SigBDD[i] < 0, SigBDD[i] = 0];
    If[SigCDD[i] < 0, SigCDD[i] = 0];
  ]
, {i, 0, n - 1, 1}]

```

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 is shown as zero.

Section 4: Spreadsheet

The next table shows the step size value, exact value, approximate value, and the absolute relative true error as a function of the step size value for each method of approximation.

```

Print["      ", "h", "      ", "AvFDD", "      ", "etFDD",
      "      ", "AvBDD", "      ", "etBDD", "      ", "AvCDD", "      ", "etCDD"];
Print[" "]
Grid[Table[{H[i], AVFDD[i], etFDD[i], AVBDD[i], etBDD[i], AVCDD[i], etCDD[i]}, {i, 0, n - 1}]]

```

h	AvFDD	etFDD	AvBDD	etBDD	AvCDD	etCDD
0.2	33 769.2	25.8702	21 653.4	19.2898	27 711.3	3.29021
0.1	30 040.6	11.9724	24 054.8	10.3389	27 047.7	0.816743
0.05	28 375.3	5.76494	25 391.3	5.35729	26 883.3	0.203824
0.025	27 587.7	2.82941	26 096.9	2.72754	26 842.3	0.0509334
0.0125	27 204.7	1.40171	26 459.4	1.37624	26 832.	0.012732
0.00625	27 015.8	0.697638	26 643.2	0.691272	26 829.5	0.0031829
0.003125	26 922.	0.348019	26 735.7	0.346428	26 828.8	0.000795719
0.0015625	26 875.3	0.17381	26 782.1	0.173412	26 828.7	0.00019893
0.00078125	26 851.9	0.0868553	26 805.3	0.0867558	26 828.6	0.0000497324
0.000390625	26 840.3	0.0434152	26 817.	0.0433903	26 828.6	0.0000124331
0.000195313	26 834.4	0.0217045	26 822.8	0.0216983	26 828.6	3.10818×10^{-6}
0.0000976563	26 831.5	0.0108515	26 825.7	0.0108499	26 828.6	7.77234×10^{-7}

Section 5: Graphs

The following graphs show the approximate solution, absolute relative true error and absolute relative approximate error as a function of step size.

```
Needs["PlotLegends`"]
data = Table[{H[i], AVFDD[i]}, {i, 0, n - 1}];
data2 = Table[{H[i], AVBDD[i]}, {i, 0, n - 1}];
data3 = Table[{H[i], AVCDD[i]}, {i, 0, n - 1}];
plot1 = ListPlot[{data},
  PlotJoined → True,
  PlotStyle → {Green},
  DisplayFunction → Identity,
  PlotRange → Full ];
plot2 = ListPlot[{data2},
  PlotJoined → True,
  PlotStyle → {{Blue, Thick}},
  DisplayFunction → Identity,
  PlotRange → Full ];
plot3 = ListPlot[{data3},
  PlotJoined → True,
  PlotStyle → {{Red, Dashed}},
  DisplayFunction → Identity,
  PlotRange → Full ];
alltogetherplot = Show[plot1, plot3, plot2, PlotRange → Automatic,
  PlotLabel → "Approximate Solution of the First Derivative of a Function
  using\ndifferent Methods of Approximation as a Function of Step Size",
  AxesLabel → {"Step Size", "Approximate Value"}];
ShowLegend[alltogetherplot, {{{Graphics[{Green, Line[{{0, 0}, {1, 0}}]}], "FDD"},
  {Graphics[{{Blue, Line[{{0, 0}, {1, 0}}]}], "BDD"},
  {Graphics[{Red, Line[{{0, 0}, {1, 0}}]}], "CDD"}},
  LegendPosition → {.5, -.6}, LegendSize → {0.3, 0.4}]

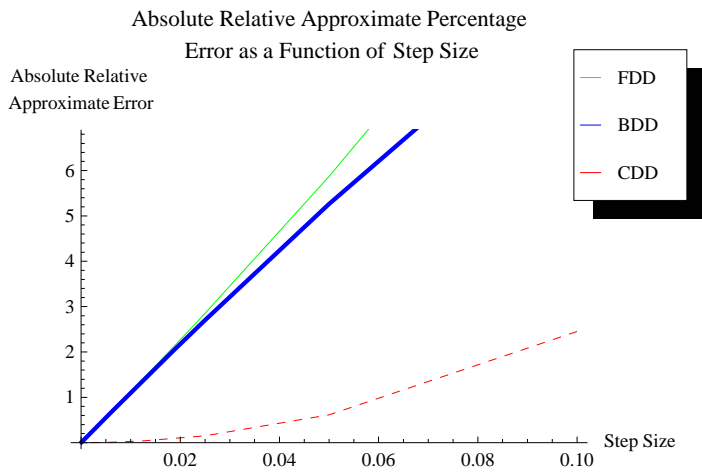
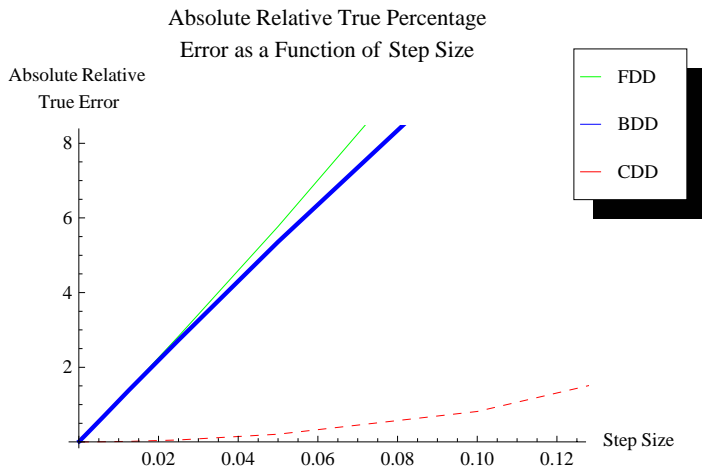
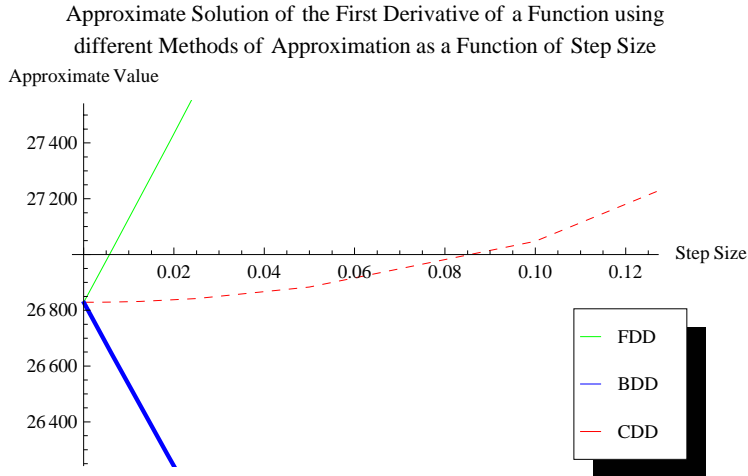
data = Table[{H[i], etFDD[i]}, {i, 0, n - 1}];
data2 = Table[{H[i], etBDD[i]}, {i, 0, n - 1}];
```

```

data3 = Table[{H[i], etCDD[i]}, {i, 0, n - 1}];
plot1 = ListPlot[{data},
  PlotJoined → True,
  PlotStyle → {Green},
  DisplayFunction → Identity,
  PlotRange → Full ];
plot2 = ListPlot[{data2},
  PlotJoined → True,
  PlotStyle → {{Blue, Thick}},
  DisplayFunction → Identity,
  PlotRange → Full ];
plot3 = ListPlot[{data3},
  PlotJoined → True,
  PlotStyle → {{Red, Dashed}},
  DisplayFunction → Identity,
  PlotRange → Full ];
alltogetherplot = Show[plot1, plot3, plot2, PlotRange → Automatic,
  PlotLabel → "Absolute Relative True Percentage\nError as a Function of Step Size",
  AxesLabel → {"Step Size", "Absolute Relative
True Error"}];
ShowLegend[alltogetherplot, {{{Graphics[{Green, Line[{{0, 0}, {1, 0}}]}]}, "FDD"},
  {Graphics[{{Blue, Line[{{0, 0}, {1, 0}}]}]}, "BDD"},
  {Graphics[{Red, Line[{{0, 0}, {1, 0}}]}]}, "CDD"}},
  LegendPosition → {.5, .1}, LegendSize → {0.3, 0.4}]]

data = Table[{H[i], eaFDD[i]}, {i, 0, n - 1}];
data2 = Table[{H[i], eaBDD[i]}, {i, 0, n - 1}];
data3 = Table[{H[i], eaCDD[i]}, {i, 0, n - 1}];
plot1 = ListPlot[{data},
  PlotJoined → True,
  PlotStyle → {Green},
  DisplayFunction → Identity,
  PlotRange → Full ];
plot2 = ListPlot[{data2},
  PlotJoined → True,
  PlotStyle → {{Blue, Thick}},
  DisplayFunction → Identity,
  PlotRange → Full ];
plot3 = ListPlot[{data3},
  PlotJoined → True,
  PlotStyle → {{Red, Dashed}},
  DisplayFunction → Identity,
  PlotRange → Full ];
alltogetherplot = Show[plot1, plot3, plot2, PlotRange → Automatic,
  PlotLabel → "Absolute Relative Approximate Percentage\nError as a Function of Step Size",
  AxesLabel → {"Step Size", "Absolute Relative
Approximate Error"}];
ShowLegend[alltogetherplot, {{{Graphics[{Green, Line[{{0, 0}, {1, 0}}]}]}, "FDD"},
  {Graphics[{{Blue, Line[{{0, 0}, {1, 0}}]}]}, "BDD"},
  {Graphics[{Red, Line[{{0, 0}, {1, 0}}]}]}, "CDD"}},
  LegendPosition → {.5, .1}, LegendSize → {0.3, 0.4}]]

```



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

$$v(t) = 2000 \ln \frac{140000}{140000 - 2100t} - 9.8t$$

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.