Giving Data Formula-Like Properties

Steven F. Bellenot

Department of Mathematics
Florida State University

Math Tech, Valdosta State University, Valdosta, GA,
Feb 22, 2008

MapleHood

- Use to do Maple assignments in Calculus 3 but stopped.
- Computer literacy has replaced programming.
- Computer classrooms are hell, bored students are surfing.
- Our Maple license didn't include student copies.
- Zebrahood: Failing students could blame Maple and by association the few professors that required it.

Why Scilab? Why not Calculus 3?

- Zebrahood: Use in all sections or in none.
- Free matlab clone selected from a list at a time when matlab licenses were too high. (Another professor (Steve Blumsack) selected Scilab over Octave and other clones.)
- Matlab is very popular and useful. Scilab is better in some ways and worse in others.
- Some Scilab classes: ODE, PDE for engineers, Matrix Algebra.
- Graph Theory, Scilab has nice graph theory package.
- Matlab in BioCalculus Lab.

- Created with a Hughes grant joint with Biology. Has only a Calculus I co-requisite.
- 1-hour computer lab mets 75 minutes in a computer classroom.
- Hughes grant had money to buy software. Maple was tried, replaced with Matlab in Word notebooks. Matlab had a symbolic toolbox, a version of maple builtin.
- Hughes grant had money, 4 TAs for 96 students plus a professor to coordinate (4 sections of 24 students.)
- The grant ran out. Moved to larger computer rooms (40 seats) and no Matlab.
- Oops, the grant had still had money for matlab. But not for the students.

Biocalculus Lab – MAP 2480

- Created with a Hughes grant joint with Biology. Has only a Calculus I co-requisite.
- 1-hour computer lab mets 75 minutes in a computer classroom.
- Hughes grant had money to buy software. Maple was tried, replaced with Matlab in Word notebooks. Matlab had a symbolic toolbox, a version of maple builtin.
- Hughes grant had money, 4 TAs for 96 students plus a professor to coordinate (4 sections of 24 students.)
- The grant ran out. Moved to larger computer rooms (40 seats) and no Matlab.
- Oops, the grant had still had money for matlab. But not for the students.

- Created with a Hughes grant joint with Biology. Has only a Calculus I co-requisite.
- 1-hour computer lab mets 75 minutes in a computer classroom.
- Hughes grant had money to buy software. Maple was tried, replaced with Matlab in Word notebooks. Matlab had a symbolic toolbox, a version of maple builtin.
- Hughes grant had money, 4 TAs for 96 students plus a professor to coordinate (4 sections of 24 students.)
- The grant ran out. Moved to larger computer rooms (40 seats) and no Matlab.
- Oops, the grant had still had money for matlab. But not for the students.

- Created with a Hughes grant joint with Biology. Has only a Calculus I co-requisite.
- 1-hour computer lab mets 75 minutes in a computer classroom.
- Hughes grant had money to buy software. Maple was tried, replaced with Matlab in Word notebooks. Matlab had a symbolic toolbox, a version of maple builtin.
- Hughes grant had money, 4 TAs for 96 students plus a professor to coordinate (4 sections of 24 students.)
- The grant ran out. Moved to larger computer rooms (40 seats) and no Matlab.
- Oops, the grant had still had money for matlab. But not for the students.

Matlab in BioCalcLab

- Ten zillion toolboxes, links with word.
- Derivatives, integrals, solving and evaluating functions using maple via the symbolic toolbox.
- Modelling is presented as curve fitting then doing calculus on the fitted curve and not dealing with the data directly.
- Motivation: replace this with direct data tools.
- And this turned out more enjoyable than I thought it would. Still haven't tried it on students.

Matlab in BioCalcLab

- Ten zillion toolboxes, links with word.
- Derivatives, integrals, solving and evaluating functions using maple via the symbolic toolbox.
- Modelling is presented as curve fitting then doing calculus on the fitted curve and not dealing with the data directly.
- Motivation: replace this with direct data tools.
- And this turned out more enjoyable than I thought it would. Still haven't tried it on students.

Matlab in BioCalcLab

- Ten zillion toolboxes, links with word.
- Derivatives, integrals, solving and evaluating functions using maple via the symbolic toolbox.
- Modelling is presented as curve fitting then doing calculus on the fitted curve and not dealing with the data directly.
- Motivation: replace this with direct data tools.
- And this turned out more enjoyable than I thought it would. Still haven't tried it on students.

Matlab in BioCalcLab

- Ten zillion toolboxes, links with word.
- Derivatives, integrals, solving and evaluating functions using maple via the symbolic toolbox.
- Modelling is presented as curve fitting then doing calculus on the fitted curve and not dealing with the data directly.
- Motivation: replace this with direct data tools.
- And this turned out more enjoyable than I thought it would. Still haven't tried it on students.

Matlab in BioCalcLab

- Ten zillion toolboxes, links with word.
- Derivatives, integrals, solving and evaluating functions using maple via the symbolic toolbox.
- Modelling is presented as curve fitting then doing calculus on the fitted curve and not dealing with the data directly.
- Motivation: replace this with direct data tools.
- And this turned out more enjoyable than I thought it would. Still haven't tried it on students.

DataDiff 1

$$data = \begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_n \\ y_1 & y_2 & y_3 & \dots & y_n \end{bmatrix}$$

$$datadiff = \begin{bmatrix} \frac{x_1+x_2}{2} & \frac{x_2+x_3}{2} & \dots & \frac{x_{n-1}+x_n}{2} \\ \frac{y_2-y_1}{x_2-x_1} & \frac{y_3-y_2}{x_3-x_2} & \dots & \frac{y_n-y_{n-1}}{x_n-x_{n-1}} \end{bmatrix}$$

One less data point at the midpoints. Data could be the piecewise linear curve connecting the points. DataDiff could be piecewise constant with jumps.

Dividing 2π into 100 equal pieces, using the midpoint reduces the error on the derivative of $\sin t$ from 0.0314056 to 0.000164.

DataDiff 2 – code

```
function yprime = datadiff(w)
% assumes domain w(1,:) range w(2,:)
x = w(1,:); y = w(2,:);
midpts = (x(1:end-1)+x(2:end))/2;
slopes = diff(y) ./ diff(x);
yprime = [midpts ; slopes];
```

It is in a matlab m-file, named datadiff.m; Scilab requires $ instead of end, but doesn't require the separate file. While % is the matlab comment character, // is the scilab comment character.

DataSolve 1

Lets look at a DataZero instead.

$$data = \begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_n \\ y_1 & y_2 & y_3 & \dots & y_n \end{bmatrix}$$

We want to find all points x_c where the piecewise linear curve connecting the points crosses the y-axis. So if $y_i > 0 > y_{i+1}$ we would want $x_c = (-y_{i+1}x_i + y_i x_{i+1})/(y_i - y_{i+1})$. Two other cases, direct hits (x_i when y_i is zero) and $y_i < 0 < y_{i+1}$

DataZero 2 – code

```
function roots = datazero(w)
% assumes domain w(1,:) range w(2,:)
x = w(1,:); y = w(2,:); ay = abs(y);
jy = ay(1:end-1)+ay(2:end);
oscilate = y(1:end-1) .* y(2:end);
i = find(oscilate < 0);
g0 = (ay(i+1).*x(i)+ay(i).*x(i+1))./jy(i);
d0 = x(find(y==0));
roots = sort[d0 g0];
```

Vectorized code, there are no for loops. Handles multiple roots. If no roots, returns an empty array. Fails(?) for data with two consecutive zeros.

Given data below and a point x_c with $x_1 < x_c < x_n$ what should be the value of y_c? Many possilbe answers. Two that we have already consisted

- Piecewise linear approximation.
- Nearest neighbor evaluation.
- General curve fitting also fits this model.

DataEval 1

Given data below and a point x_c with $x_1 < x_c < x_n$ what should be the value of y_c? Many possilbe answers. Two that we have already consisted

- Piecewise linear approximation.
- Nearest neighbor evaluation.
- General curve fitting also fits this model.

Given data below and a point x_c with $x_1 < x_c < x_n$ what should be the value of y_c? Many possilbe answers. Two that we have already consisted

- Piecewise linear approximation.
- Nearest neighbor evaluation.
- General curve fitting also fits this model.

DataEval 1

Given data below and a point x_c with $x_1 < x_c < x_n$ what should be the value of y_c? Many possilbe answers. Two that we have already consisted

- Piecewise linear approximation.
- Nearest neighbor evaluation.
- General curve fitting also fits this model.

Piecewise Linear DataEval – code

```
function y0 = dataeval(x0, w)
% assumes domain w(1,:) range w(2,:)
x = w(1,:); y = w(2,:); xx = x - x0;
eval0 = y(find(x==0));
oscilate = xx(1:end-1) .* xx(2:end);
i = find(oscilate < 0);
eval2 = (xx(i+1).*y(i)-xx(i).*y(i+1))./(xx(i+1)-xx(
y0 = union[eval0 eval2];
```

This is not vector friendly, x0 needs to be a scalar.

DataInt

```
function wint = dataint(w)
% assumes domain w(1,:) range w(2,:)
x = w(1,:); y = w(2,:); n = size(x,1);
newx = zeros(1,n+1); newy = zeros(1,n+1);
dx = x(2:end) - x(1:end);
newx(2:end-1) = (x(2:end)+x(1:end-1))/2;
newx(1) = newx(2)-dx(1);
newx(n+1) = newx(n)+dx(n-1);
newy(1) = 0;
for i = 1:n,
newy(i+1) = newy(i)+y(i)*dx(i);
end;
wint = [newx; newy];
```

Nearest Neighbor Evaluation

In the abstract, what all evaluations can do is given

$$data = \begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_n \\ y_1 & y_2 & y_3 & \dots & y_n \end{bmatrix}$$

is to find functions $f_i(x)$ based on the first row and evaluate according to

$$\sum_{i=1}^{n} f_i(x) y_i$$

the goal is to find vector friendly f_i. For nearest neighbor, we want f_i to be the characteristic function of the interval (m_{i-1}, m_i) where m_i is the midpoint of x_i and x_{i+1}. (First approximation.)

Step Functions

The goal is vector friendly step functions. And it turns out the sign function works as a building block. Define step = (sign + 1)/2 so that?

$$\mathrm{step}(x) = \begin{cases} 1 & x > 0 \\ \frac{1}{2} & x = 0 \\ 0 & x < 0 \end{cases}$$

so that the resulting functions are vector friendly and make sense at the midpoints.

$$f_i(x) = \mathrm{step}(x - m_{i-1})\mathrm{step}(m_i - x)$$

Summary

- Working with data rather than curve fitting is doable and at least mildly interesting.
- 3D applications: scanner point clouds data are worked with directly. So there are applications.

Summary

- Working with data rather than curve fitting is doable and at least mildly interesting.
- 3D applications: scanner point clouds data are worked with directly. So there are applications.