Basic Ideas

Unit 1

content

1. What methods are "computational"

2. Analytical methods vs numerical methods

3. Computational experiments

4. Computational abstract algebra, computational topology, computational algebraic geometry

5. Journals

what methods are computational

Given a mathematical model of a phenomenon, e.g.

- definite integral – a model of ????

what methods are computational

Given a mathematical model of a phenomenon, e.g.

- definite integral – a model of ????
- differential equation – a model of ????

what methods are computational

Given a mathematical model of a phenomenon, e.g.

- definite integral – a model of ????
- differential equation – a model of ????
- spline (piecewise polynomial) – a model of ????

what methods are computational

Given a mathematical model of a phenomenon, e.g.

- definite integral – a model of ????
- differential equation – a model of ????
- spline (piecewise polynomial) – a model of ????

Analytical solutions remain the best (if known).

analytical solutions

- definite integral – analytical solution ????

analytical solutions

- definite integral – analytical solution ????
- differential equation – analytical solution ????

analytical solutions

- definite integral – analytical solution ????
- differential equation – analytical solution ????

Analytical solutions in many cases are not known.

numerical solutions

- definite integral – numerical solutions ????

numerical solutions

- definite integral – numerical solutions ????
- differential equation – numerical solutions ????

numerical solutions

- definite integral – numerical solutions ????
- differential equation – numerical solutions ????

How close is the numerical solution to the analytical one?

numerical solutions

- definite integral – numerical solutions ????

- differential equation – numerical solutions ????

How close is the numerical solution to the analytical one?

Which algorithm is better?

numerical solutions

- definite integral – numerical solutions ????

- differential equation – numerical solutions ????

How close is the numerical solution to the analytical one?

Which algorithm is better?

Which algorithm is faster?

computational experiment: splines

- analytically deduce dimension of C^1 quadratic splines over two triangles sharing an edge

computational experiment: splines

- analytically deduce dimension of C^1 quadratic splines over two triangles sharing an edge

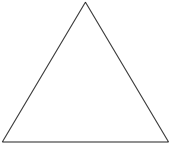


Figure: $ax^2 + by^2 + cxy + dx + ey + f$,

computational experiment: splines

- analytically deduce dimension of C^1 quadratic splines over two triangles sharing an edge

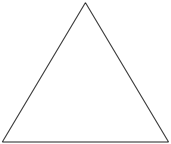


Figure: $ax^2 + by^2 + cxy + dx + ey + f, \;\; dim\ S_2^1(\Delta_1) = 6$

computational experiment: splines

- analytically deduce dimension of C^1 quadratic splines over two triangles sharing an edge

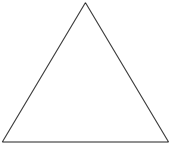


Figure: $ax^2 + by^2 + cxy + dx + ey + f\,,\;\; dim\ S_2^1(\Delta_1) = 6$

- How about two triangles? Check numerically.

computational experiment: splines

- analytically deduce dimension of C^1 quadratic splines over two triangles sharing an edge

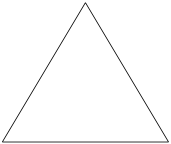


Figure: $ax^2 + by^2 + cxy + dx + ey + f$, $dim\ S_2^1(\Delta_1) = 6$

- How about two triangles? Check numerically.

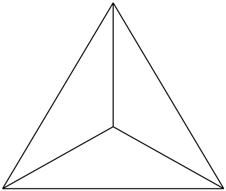
Why the computational experiment is not a proof?

computational experiment: splines

- analytically deduce dimension of C^1 quadratic splines over three triangles sharing a vertex

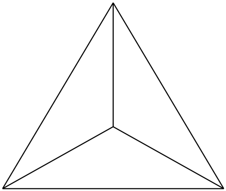
computational experiment: splines

• analytically deduce dimension of C^1 quadratic splines over three triangles sharing a vertex



computational experiment: splines

- analytically deduce dimension of C^1 quadratic splines over three triangles sharing a vertex



- Check numerically.

computational algebra, topology, algebraic geometry

- Link

computational algebra, topology, algebraic geometry

- Link
- Link

computational algebra, topology, algebraic geometry

- ▸ Link
- ▸ Link
- ▸ Link

computational algebra, topology, algebraic geometry

- ▸ Link
- ▸ Link
- ▸ Link
- ▸ Link

journals, meetings, people

- SIAM ▸ Link

journals, meetings, people

- SIAM Link
- Journal Ranking Link

journals, meetings, people

- SIAM ▸ Link
- Journal Ranking ▸ Link

BREAK 10 min

content

1. Discretization (discrete vs continuous)
2. Stability
3. Well-posed vs ill-posed problems
4. Conditioning
5. Types of errors
6. Accuracy
7. Operation count
8. Loss of significant digits

discretization

continuous model $\qquad f'(a) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$

discretization

continuous model $$f'(a) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$$

discrete model $$f'(a) \approx df(a,h) := \frac{f(a+h) - f(a)}{h}$$

discretization

continuous model $$f'(a) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$$

discrete model $$f'(a) \approx df(a,h) := \frac{f(a+h) - f(a)}{h}$$

approximation error $$e(a,h) := |f'(a) - df(a,h)|$$

discretization

continuous model $$f'(a) = \lim_{h\to 0} \frac{f(a+h) - f(a)}{h}$$

discrete model $$f'(a) \approx df(a,h) := \frac{f(a+h) - f(a)}{h}$$

approximation error $$e(a,h) := |f'(a) - df(a,h)|$$

$$f(a+h) = f(a) + f'(a)h + f''(c)h^2, \quad a \leq c \leq a+h$$

discretization

continuous model $$f'(a) = \lim_{h\to 0} \frac{f(a+h) - f(a)}{h}$$

discrete model $$f'(a) \approx df(a,h) := \frac{f(a+h) - f(a)}{h}$$

approximation error $$e(a,h) := |f'(a) - df(a,h)|$$

$$f(a+h) = f(a) + f'(a)h + f''(c)h^2, \quad a \leq c \leq a+h$$

$$f'(a) = df(a,h) - f''(c)h$$

discretization

continuous model $$f'(a) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$$

discrete model $$f'(a) \approx df(a,h) := \frac{f(a+h) - f(a)}{h}$$

approximation error $$e(a,h) := |f'(a) - df(a,h)|$$

$$f(a+h) = f(a) + f'(a)h + f''(c)h^2, \quad a \leq c \leq a+h$$

$$f'(a) = df(a,h) - f''(c)h$$

$$e(a,h) \leq Kh, \qquad |f''(x)| \leq K$$

discretization

continuous model $\quad f'(a) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$

discrete model $\quad f'(a) \approx df(a,h) := \frac{f(a+h) - f(a)}{h}$

approximation error $\quad e(a,h) := |f'(a) - df(a,h)|$

$$f(a+h) = f(a) + f'(a)h + f''(c)h^2, \quad a \leq c \leq a+h$$

$$f'(a) = df(a,h) - f''(c)h$$

$$e(a,h) \leq Kh, \qquad |f''(x)| \leq K$$

$$e(a,h) = \mathcal{O}(h)$$

stability

algorithm is unstable= arbitrarily small errors in input (the measurement data) may lead to indefinitely large errors in the output (solutions)

stability

algorithm is unstable= arbitrarily small errors in input (the measurement data) may lead to indefinitely large errors in the output (solutions)

The problem of finding a solution $q = R(u)$ is well-posed, if for any $\varepsilon > 0$ there is a $\delta > 0$ such that for any u_1, u_2 such that $q_1 = R(u_1)$, $q_2 = R(u_2)$ and $|u_1 - u_2| < \delta$ it follows that $|q_1 - q_2| < \varepsilon$.

stability

algorithm is unstable= arbitrarily small errors in input (the measurement data) may lead to indefinitely large errors in the output (solutions)

The problem of finding a solution $q = R(u)$ is well-posed, if for any $\varepsilon > 0$ there is a $\delta > 0$ such that for any u_1, u_2 such that $q_1 = R(u_1)$, $q_2 = R(u_2)$ and $|u_1 - u_2| < \delta$ it follows that $|q_1 - q_2| < \varepsilon$.

- Example 1: $q = \frac{1}{u}$, $0 < u < 1$

- Example 2: $q = \frac{1}{u}$, $10^{-3} < u < 1$

well-posed vs ill-posed problems

An *ill-posed problem* is a problem that either has no solutions (in the desired class),

well-posed vs ill-posed problems

An *ill-posed problem* is a problem that either has no solutions (in the desired class), or has many (two or more) solutions, or

well-posed vs ill-posed problems

An *ill-posed problem* is a problem that either has no solutions (in the desired class), or has many (two or more) solutions, or the solution is unstable.

well-posed vs ill-posed problems

An *ill-posed problem* is a problem that either has no solutions (in the desired class), or has many (two or more) solutions, or the solution is unstable.

In this course we shall work with well posed problems. “Definitions and examples of inverse and ill-posed problems”,S. I. Kabanikhin ▸ Link See examples on p. 324

conditioning

• conditioning = sensitivity of the solution to perturbations of the input data.

low sensitivity = well conditioned

high sensitivity = ill conditioned

conditioning

- conditioning = sensitivity of the solution to perturbations of the input data.

low sensitivity = well conditioned

high sensitivity = ill conditioned

- a well posed problem can be ill conditioned

- Example: $q = \frac{1}{u}$, $10^{-3} < u < 1$

types of errors

- unavoidable error (input data, measurement, modeling)

types of errors

- unavoidable error (input data, measurement, modeling)
- approximation error (algorithm)

types of errors

- unavoidable error (input data, measurement, modeling)
- approximation error (algorithm)
- round-off error (computer)

types of errors

- unavoidable error (input data, measurement, modeling)
- approximation error (algorithm)
- round-off error (computer)

$x_0 \approx a, \quad a - \delta \leq a \leq a + \delta$

types of errors

- unavoidable error (input data, measurement, modeling)
- approximation error (algorithm)
- round-off error (computer)

$x_0 \approx a, \quad a - \delta \leq a \leq a + \delta$

$f(x) \approx \sin x, \qquad \sin x - \varepsilon \leq f(x) \leq \sin x + \varepsilon$

types of errors

- unavoidable error (input data, measurement, modeling)
- approximation error (algorithm)
- round-off error (computer)

$x_0 \approx a, \quad a - \delta \leq a \leq a + \delta$

$f(x) \approx \sin x, \qquad \sin x - \varepsilon \leq f(x) \leq \sin x + \varepsilon$

$\sin x \approx x - \frac{x^3}{3!} + \frac{x^5}{5!}$

accuracy and operation count

$\sin x \approx x - \frac{x^3}{3!} + \frac{x^5}{5!}$

$\sin x \approx x - \frac{x^3}{3!}$

accuracy and operation count

$$\sin x \approx x - \frac{x^3}{3!} + \frac{x^5}{5!}$$

$$\sin x \approx x - \frac{x^3}{3!}$$

$$y = 1 + x + x^2 + \cdots + x^{215}$$

$$y = \frac{1-x^{216}}{1-x}$$

loss of significant digits: a famous example

$$\sqrt{x} - \sqrt{x-1}$$

Suppose we format short, and $x = 20157$. Then

$\sqrt{x} \approx 141.9753$, $\sqrt{x-1} = 141.9718$ and $\sqrt{x} - \sqrt{x-1} = 0.0035$

loss of significant digits: a famous example

$\sqrt{x} - \sqrt{x-1}$

Suppose we format short, and $x = 20157$. Then

$\sqrt{x} \approx 141.9753$, $\sqrt{x-1} = 141.9718$ and $\sqrt{x} - \sqrt{x-1} = 0.0035$

Suppose we compute in format long, and $x = 20157$. Then

$\sqrt{x} - \sqrt{x-1} = 0.003521781785139$

loss of significant digits: a famous example

$\sqrt{x} - \sqrt{x-1}$

Suppose we format short, and $x = 20157$. Then

$\sqrt{x} \approx 141.9753$, $\sqrt{x-1} = 141.9718$ and $\sqrt{x} - \sqrt{x-1} = 0.0035$

Suppose we compute in format long, and $x = 20157$. Then

$\sqrt{x} - \sqrt{x-1} = 0.003521781785139$

$\frac{1}{\sqrt{x}+\sqrt{x-1}}$

Try this with Matlab to see if it helps