Numerical Analysis / Scientific Computing

CS 450/ECE 491 - Spring 2016

Andreas Klöckner

1 Introduction to Scientific Computing

What's the point of this class?

'Scientific Computing' describes a family of approaches to obtain approximate solutions to problems...

...once they've been stated mathematically.

- Name some applications
  - Simulation
  - Data Analysis — Machine learning
  - Image processing
  - Signal processing

What do we study, and how?

- Problems with real numbers (i.e. 'continuous' problems)

  not discrete

- What mathematical ideas do we use?

  representation → compute with that

- How good of an answer can we expect to our problem?

  bound "error"

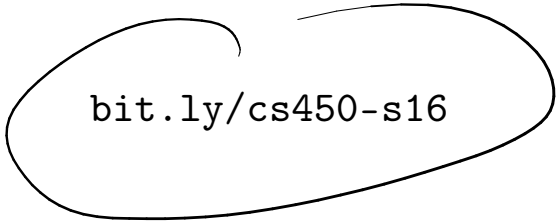
- How fast can we expect the computation to complete?

  complexity
  constants

- How do these numerical methods get implemented?

  tools → how do you use them?

Class web page



- Assignments
  - HW0!
  - Pre-lecture quizzes
  - In-lecture interactive content (bring computer or phone if possible)
- Textbook
- Exams
- Class outline (with links to notes/demos/activities/quizzes)
- Virtual Machine Image
- Piazza

- Policies
- Video

Programming Language: Python/numpy

- Reasonably readable
- Reasonably beginner-friendly
- Mainstream (top 5 in 'TIOBE Index')
- Free, open-source
- Great tools and libraries (not just) for scientific computing
- Python 2/3? 3!
- `numpy`: Provides an array datatype
  Will use this and `matplotlib` all the time.
- See class web page for learning materials

Demo: Sum the squares of the integers from 0 to 100.
First without numpy, then with numpy.

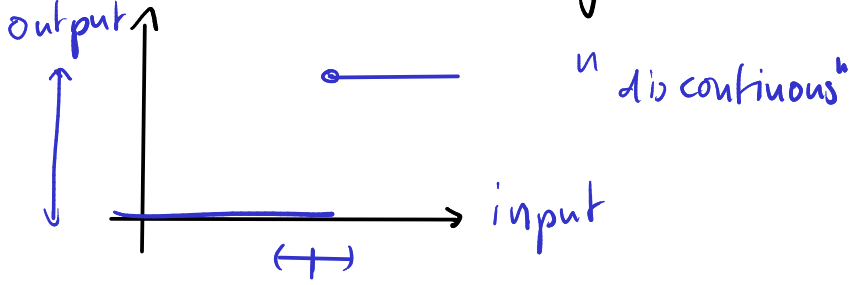
1.1 Errors, Conditioning, Accuracy, Stability

What problems *can* we study in the first place?

- To be able to compute a solution (through a process that introduces errors), the problem...

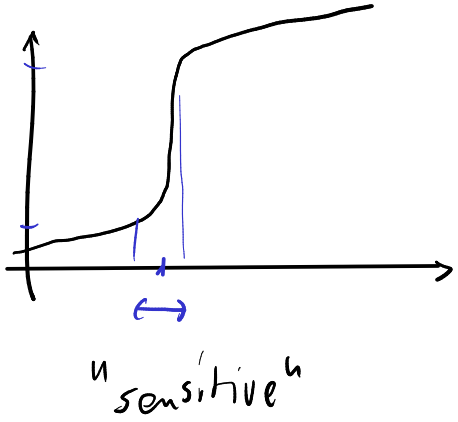
"well-posed"

- needs to have a solution
- that solution should be unique
- the solution should depend continously on the inputs



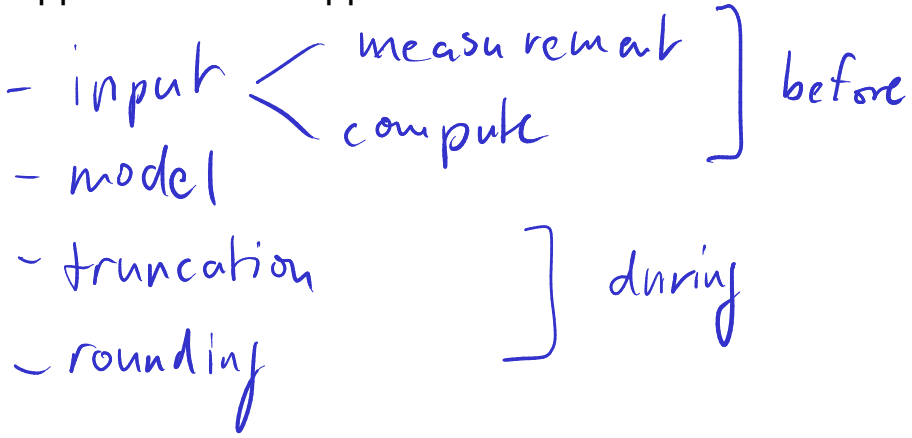
Dependency on Inputs

- We excluded discontinuous problems–because we don't stand much chance for those.
  ...what if the problem's input dependency is just close to discontinuous?



Approximation

- *When* does approximation happen?



Demo: Truncation vs. Rounding

Example: Surface Area of the Earth

- Compute the surface area of the earth.
  What parts of your computation are approximate?

Earth is a sphere

$A = 4\pi r^2$

Look up radius of the earth

Measuring Error

- How do we measure error?
  Idea: Consider all error as being *added onto* the result.

→ Absolute error = approx value − true value

→ Relative error = $\dfrac{\text{Absolute error}}{|\text{True value}|}$

"estimate" "bounds"

Recap: Norms

- What's a norm?

$$f: \mathbb{R}^n \longrightarrow \mathbb{R}_0^+$$

- Define norm.

For a vector $x \in \mathbb{R}^n$:
- $\|\vec{x}\| > 0 \Leftrightarrow \vec{x} \neq 0$
- $\|\gamma \vec{x}\| = |\gamma| \, \|\vec{x}\|$
- $\|x+y\| \leq \|x\| + \|y\|$ → Δ ineq

$\left\| \begin{pmatrix} 2 \\ 3 \end{pmatrix} \right\|$

$\|\vec{x}\|$

- Examples of norms?

$$\left\| \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \right\|_p = \sqrt[p]{|x_1|^p + \cdots |x_n|^p} \quad (p \geq 1)$$

- Does the choice of norm really matter much?

$\|\cdot\|$ $\quad$ $\|\cdot\|^*$

$$\alpha \|\vec{x}\| \leq \|x\|^* \leq \beta \|\vec{x}\|$$

In finitely many dimensions, all norms are equivalentt

Demo: Vector norms