Math 4329: Numerical Analysis Lecture 02

Natasha S. Sharma, PhD

Last Lecture

- $|f(-1) - p_1(-1)| \leq 0.5$ and $|f(-0.5) - p_1(-0.5)| \leq 0.125.$
- $f(-1) = 0.3679, \; p_1(-1) = 0, \; p_2(-1) = 0.5.$
- $f(-0.5) = 0.6065$
  $p_1(-0.5) = 0.5, \; p_2(-0.5) = 0.625$
- Taylor's Remainder to calculate the approximation error

$$R_n(x) := f(x) - p_n(x) = \frac{(x-a)^{n+1}}{(n+1)!} f^{(n+1)}(c_x)$$

c_x is an unknown number between x and a.

Three types of questions we are interested in answering

- Use the Taylor polynomial of degree 1 and 2 to find an approximation to $\sqrt{2} = 1.41421356237$.
  Solution:
  1. $f(x) = \sqrt{x+1}$, $x = 1$.
  2. $$f'(x) = \frac{1}{2\sqrt{x+1}}, \quad f''(x) = \frac{-1}{4(x+1)^{3/2}}.$$
  3. Next step: Pick the suitable choice of 'a'.
  4. $$p_1(x) = f(0) + f'(0)x = 1 + \frac{x}{2},$$ $$p_2(x) = p_1(x) + \frac{f''(0)x^2}{2} = 1 + \frac{x}{2} - \frac{x^2}{8}.$$
  5. $\sqrt{2} \approx 1.5$ and $\sqrt{2} \approx 1.375$.
- How to approximate the value of $\log(2)$?
  Hint The choice of a is non zero.

Three types of questions we are interested in answering

- Bound the error in using the degree 3 Taylor polynomial $p_3(x)$ to approximate e^x on $[-1, 1]$ using Taylor's remainder formula.
- Solution:

$$\begin{aligned} |f(x) - p_3(x)| &\leq \frac{|x|^4}{4!} e^{c_x} \\ &\leq \frac{1}{24} e^{c_x} \\ &\leq \frac{1}{24} e^{1} = 0.1133. \end{aligned}$$

Three types of questions we are interested in answering

- Bound the error in using the degree 3 Taylor polynomial $p_3(x)$ to approximate e^x on $[-1, 1]$ using Taylor's remainder formula.
- Solution:

$$\begin{aligned}
|f(x) - p_3(x)| &\leq \frac{|x|^4}{4!} e^{c_x} \\
&\leq \frac{1}{4!} e^{c_x} \\
&\leq \frac{1}{24} e^1 = 0.1133.
\end{aligned}$$

Three types of questions we are interested in answering

- How large should the degree ~~2n+1~~ **2n** be of the Taylor polynomial $p_{\mathbf{2n}}(x)$ to have

$$|\cos(x) - p_{\mathbf{2n}}(x)| \le 10^{-4}$$

for all $-\frac{\pi}{2} \le x \le \frac{\pi}{2}$?

- Solution:

$$\begin{aligned} |f(x) - p_{\mathbf{2n}}(x)| &\le \frac{|x|^{(2n+2)}}{((2n+2)!}|\cos(c_x)| \\ &\le \frac{|x|^{2n+2}}{(2n+2)!} * 1 \\ &\le \frac{|\frac{\pi}{2}|^{2n+2}}{(2n+2)!} \end{aligned}$$

$$\leq \frac{\left(\frac{\pi}{2}\right)^{2n+2}}{(2n+2)!}$$
$$\leq 10^{-4}$$

$n = 3$ gives $0.00091926027 > 10^{-4}$
$n = 4$ gives $0.00002520204 < 10^{-4}$.
Answer: $n \geq 4$.

Repeat the previous problem with $\cos(x)$ replaced with $\log(x+2)$.
You can now work out the problems from Worksheet 01!

With each lecture, our definition of numerical analysis is going to evolve.

Numerical Analysis is the study of techniques to computationally solve a problem that is, develop a sequence of numerical calculations to get a suitable solution.

This suitable answer is determined by the error tolerance denoted by ε.

Part of this process is to take into account the errors that arise in these calculations from the errors in the arithmetic operations or from other sources.

Computer use binary arithmetic, representing each number as a binary number: a finite sum of integer powers of 2.
Some numbers can be represented exactly, but others such as $\frac{1}{10}, \frac{1}{100}, \frac{1}{1000}, \cdots$ cannot be represented exactly.

$$2.125 = 2 + 2^{-3}$$

has an exact representation in binary but the following number has an inexact representation:

$$3.1 \approx 2^{1} + 2^{0} + 2^{-4} + 2^{-5} + 2^{-8} + \cdots .$$

Furthermore, π have no finite representation in either decimal or binary number system.
Please see Appendix E of the textbook for a more details.

Computers use 2 formats for storing numbers:

1. Fixed-Point numbers used to store integers.
   Each number is stored in a computer word of 32 binary digits (bits) with values 0 or 1. Hence there are 2^{32} different numbers can be stored.
   If we permit negative numbers, we can represent integers in the range $-2^{-31} \leq x \leq 2^{31} - 1$ since there are 2^{32} such numbers. Since $2^{31} \approx 2.1 \times 10^9$.
   The range of the fixed-point numbers is too restrictive for scientific computing. The stored numbers that are stored are equally spaced.
2. Floating-point numbers approximate real numbers. The numbers are not equally spaced and a wide range of numbers are represented exactly.

Floating-Point Representation

For $x \neq 0$ written in decimal system, we can uniquely write it as

$$x = \sigma \cdot \bar{x} \cdot 10^e$$

where

1. $\sigma = +1$ or -1 is the sign,
2. e is an integer and is the exponent and
3. $1 \leq \bar{x} < 10$, the significand or mantissa

Example: $124.62 = \sigma \underbrace{(1.2462)}_{\bar{x}} \cdot 10^e$, with $\sigma = 1$ and $e = 2$.

Floating-Point Representation

Limitations on the the floating point representation of any $x \in \mathbb{R}$ is

1. number of digits in the mantissa $\bar{x}$
2. size of e

Suppose we limit

1. number of digits in the mantissa $\bar{x}$ to 4.
2. $-99 \leq e \leq 99$

This is the four-digit decimal floating point arithmetic. That is, we can only store the first four digits of a number accurately even if the fourth digit is obtained by rounding.

Floating-Point Representation

Limitations on the the floating point representation of any $x \in \mathbb{R}$ is

1. number of digits in the mantissa $\bar{x}$
2. size of e

Suppose we limit

1. number of digits in the mantissa $\bar{x}$ to 4.
2. $-99 \leq e \leq 99$

This is the four-digit decimal floating point arithmetic. That is, we can only store the first four digits of a number accurately even if the fourth digit is obtained by rounding.

Floating-Point Representation of a binary number x

For $x \neq 0$ written in binary system, we can express it as

$$x = \sigma \cdot \bar{x} \cdot 2^e$$

where

1. $\sigma = +1$ or -1 is the sign,
2. e is an integer and is the exponent and
3. $\bar{x}$ is a binary fraction satisfying

   $$(1)_2 \leq \bar{x} < (10)_2,$$

   which in decimal translates to $1 \leq \bar{x} < 2$.
4. Example: $x = (11011.0111)_2 = \sigma \underbrace{(1.10110111)_2}_{\bar{x}} \cdot 2^e$, with

   $\sigma = 1$ and $e = 4 = (100)_2$.

Floating-point representation of a binary number x is given by the definition on the previous page with a restriction on

1. Number of digits in $\bar{x}$: the precision of the binary floating-point representation of x,
2. size of e.

The IEEE single precision floating-point representation of x has

1. Precision of 24 bits
2. $-126 \leq e \leq 127$
3. $$x = \sigma \cdot (1.a_1 a_2 \cdots a_{23}) \cdot 2^e$$

   stores 32 bits with

$$\underbrace{b_1}_{\sigma} \underbrace{b_2 b_3 \cdots b_9}_{E=e+127} \underbrace{b_{10} b_{11} \cdots b_{32}}_{\bar{x}}$$

The IEEE double precision floating-point representation of x has

1. Precision of 53 bits
2. $-1022 \leq e \leq 1023$
3. $$x = \sigma \cdot (1.a_1 a_2 \cdots a_{52}) \cdot 2^e$$

   stores 64 bits with

   $$\underbrace{b_1}_{\sigma} \underbrace{b_2 b_3 \cdots b_{12}}_{E=e+1023} \underbrace{b_{13} b_{14} \cdots b_{64}}_{\bar{x}}$$

Error in a computational science problem:

1. Original Errors
   - Modeling Errors
   - Blunders and mistakes
   - Physical Measurement Errors
   - Machine Representation and Arithmetic Errors
   - Mathematical Approximation Errors. For instance: $\int_0^1 e^{-x^2} dx$ using Taylor approximation.
2. Consequence of Errors
   - Loss of Significance
   - Noise in function evaluation
   - Under and overflow errors

Error in a computational science problem:

1. Original Errors
   - Modeling Errors
   - Blunders and mistakes
   - Physical Measurement Errors
   - Machine Representation and Arithmetic Errors
   - Mathematical Approximation Errors. For instance: $\int_0^1 e^{-x^2}\,dx$ using Taylor approximation.
2. Consequence of Errors
   - Loss of Significance
   - Noise in function evaluation
   - Under and overflow errors

Error in a computational science problem:

1. Original Errors
   - Modeling Errors
   - Blunders and mistakes
   - Physical Measurement Errors
   - Machine Representation and Arithmetic Errors
   - Mathematical Approximation Errors. For instance: $\int_0^1 e^{-x^2}\,dx$ using Taylor approximation.
2. Consequence of Errors
   - Loss of Significance
   - Noise in function evaluation
   - Under and overflow errors

Error in a computational science problem:

1. Original Errors
   - Modeling Errors
   - Blunders and mistakes
   - Physical Measurement Errors
   - Machine Representation and Arithmetic Errors
   - Mathematical Approximation Errors. For instance: $\int_0^1 e^{-x^2}dx$ using Taylor approximation.
2. Consequence of Errors
   - Loss of Significance
   - Noise in function evaluation
   - Under and overflow errors

Error in a computational science problem:

1. Original Errors
   - Modeling Errors
   - Blunders and mistakes
   - Physical Measurement Errors
   - Machine Representation and Arithmetic Errors
   - Mathematical Approximation Errors. For instance: $\int_0^1 e^{-x^2}\,dx$ using Taylor approximation.
2. Consequence of Errors
   - Loss of Significance
   - Noise in function evaluation
   - Under and overflow errors

Error in a computational science problem:

1. Original Errors
   - Modeling Errors
   - Blunders and mistakes
   - Physical Measurement Errors
   - Machine Representation and Arithmetic Errors
   - Mathematical Approximation Errors. For instance: $\int_0^1 e^{-x^2} dx$ using Taylor approximation.
2. Consequence of Errors
   - Loss of Significance
   - Noise in function evaluation
   - Under and overflow errors

Error in a computational science problem:

1. Original Errors
   - Modeling Errors
   - Blunders and mistakes
   - Physical Measurement Errors
   - Machine Representation and Arithmetic Errors
   - Mathematical Approximation Errors. For instance: $\int_0^1 e^{-x^2}\,dx$ using Taylor approximation.
2. Consequence of Errors
   - Loss of Significance
   - Noise in function evaluation
   - Under and overflow errors

Error in a computational science problem:

1. Original Errors
   - Modeling Errors
   - Blunders and mistakes
   - Physical Measurement Errors
   - Machine Representation and Arithmetic Errors
   - Mathematical Approximation Errors. For instance: $\int_0^1 e^{-x^2} dx$ using Taylor approximation.
2. Consequence of Errors
   - Loss of Significance
   - Noise in function evaluation
   - Under and overflow errors

Error in a computational science problem:

1. Original Errors
   - Modeling Errors
   - Blunders and mistakes
   - Physical Measurement Errors
   - Machine Representation and Arithmetic Errors
   - Mathematical Approximation Errors. For instance: $\int_0^1 e^{-x^2}\,dx$ using Taylor approximation.
2. Consequence of Errors
   - Loss of Significance
   - Noise in function evaluation
   - Under and overflow errors

Error in a computational science problem:

1. Original Errors
   - Modeling Errors
   - Blunders and mistakes
   - Physical Measurement Errors
   - Machine Representation and Arithmetic Errors
   - Mathematical Approximation Errors. For instance: $\int_0^1 e^{-x^2} dx$ using Taylor approximation.
2. Consequence of Errors
   - Loss of Significance
   - Noise in function evaluation
   - Under and overflow errors

Consequence of Errors: Loss of Significance

Consider evaluation of

$$f(x) = x(\sqrt{x+1} - \sqrt{x}) \quad \text{for } x = 10^p, p = 0, 1, 2, 3, 4, 5.$$

As x increases there are fewer values of accuracy in the computed value $f(x)$.

$\sqrt{101} = \underbrace{10.04999}_{\text{rounded}}, \quad \sqrt{100} = 10, \; \sqrt{x+1} - \sqrt{x} = 0.0499000$

however the true value is 0.0498756.

This calculation admits a loss of significance error. Three digits of accuracy were canceled by subtraction of the corresponding digits in $\sqrt{x} = \sqrt{100}$.

There are two causes of loss of this accuracy:

1. the mathematical form of $f(x)$
2. the finite precision 6-digit decimal arithmetic used

Increasing the precision is not possible always so instead we can consider a reformulation of $f(x)$.

$$f(x) = \frac{x}{\sqrt{x+1} + \sqrt{x}}$$

yields the right values on a 6 digit decimal calculator that is f(100)$=$ 4.98756.

Consider evaluation of

$$f(x) = \frac{1-\cos(x)}{x^2} \qquad \text{for } x = 10^{-p}, p = 1, 2, 3, 4, 5,$$

on a computer with 9-digit decimal arithmetic used.
For $x = 0.01$, $\cos(x) = 0.9999500004$ $(= 0.999950000416665)$

$$1 - \cos(0.01) = 0.0000499996 \; (= 4.999958333495869e - 05)$$

which only have 5 significant digits with 4 lost due to subtraction.
To avoid loss due to subtraction of nearly equal quantities, we use the Taylor approximation for $\cos(x)$ about 0.

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + R_6(x) \tag{1}$$

$$\text{where } R_6(x) = \frac{x^8}{8!}\cos(c_x).$$

$$f(x) = \frac{1-\cos(x)}{x^2} = \frac{1}{x^2}\Big(\frac{x^2}{2} - \frac{x^4}{4!} + \frac{x^6}{6!} - \frac{\cos(c_x)x^8}{8!}\Big)$$

$$= \frac{1}{2} - \frac{x^2}{4!} + \frac{x^4}{6!} - \frac{\cos(c_x)x^6}{8!}.$$

$$\to \frac{1}{2} \quad \text{as } x \to 0.$$

This is in conformity with applying L'Hopital's Rule to obtain the true limiting value $\frac{1}{2}$.

For $|x| \leq 0.1$,

$$|\frac{\cos(c_x)x^6}{8!}| \leq \frac{(0.1)^6}{8!} \leq 2.5e-11$$

We can choose a smaller polynomial degree however that will increase the approximation error.

Loss-of-Significance Error

When two nearly equal quantities are subtracted, leading significant digits are lost. This can be circumvented by:

1. Replace the function with a simpler function (example use the Taylor polynomial).
   Example:

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + R_6(x)$$

   where $R_6(x) = \frac{x^8}{8!}\cos(c_x)$ where c_x is an unknown between 0 and x.

2. Reformulate the mathematical expression for example

$$\sqrt{x+1} - \sqrt{x} = \frac{(\sqrt{x+1})^2 - (\sqrt{x})^2}{\sqrt{x+1} + \sqrt{x}}.$$

Message:

1. In the evaluation of function, avoid the operation of subtraction especially when the quantities being subtracted are close to each other.
2. This can be done by reformulating the function in a mathematically equivalent but numerically more accurate manner.

Another example to evaluate e^{-7}, instead of using the Taylor series (with the remainder term) applied to $f(x) = e^{-7}$ which will involve lots of subtraction, we consider applying the Taylor series to $f(x) = e^{7}$. That is:

$$e^{-7} = \frac{1}{e^7} = \frac{1}{\text{Taylor Series for } e^7}$$

Relative Error

Absolute Error is denoted by error(x_a) and is defined as

$$\text{Error}(x_a) := x_t - x_a,$$

where x_t denotes a true value. This can be a positive or a negative quantity.

Relative Error is defined as

$$\text{Rel}(x_a) := \frac{\text{Error}(x_a)}{\text{true value}} = \frac{x_t - x_a}{x_t},$$

Example: For the approximation x_a to
$x_t = \pi = \frac{22}{7} \approx 3.14159265\cdots$.
x_{a_7} using 7-digit precision is 3.1415927, $\text{Rel}(x_{a_7}) = \frac{\pi - x_{a_7}}{\pi} = ?$.
For $x_{a_6} = 3.141593$ what is $\text{Rel}(x_{a_6}) = \frac{\pi - x_{a_6}}{\pi} = ?$

Relative Error versus Absolute Error?

Consider the following two problems:

1. The precise distance between two cities A and B is $x_{T1} = 100$ km and the measured distance is $x_{a1} = 99$ km.

$$\text{Error}(x_{a1}) := 1 \text{ km} \ , \ \text{Rel}(x_{a1}) = \frac{1}{100} = 0.01 = 1\%.$$

2. The precise distance between two cities A and B is $x_{T2} = 2$ km and the measured distance is $x_{a2} = 1$ km.

$$\text{Error}(x_{a2}) := 1 \text{ km} \ , \ \text{Rel}(x_{a2}) = \frac{1}{2} = 0.5 = 50\%.$$

Relative Error is more true representation of the aprroximation error!

Significant Digits

The number of significant digits in an approximated value x_a is the number of its leading digits that are correct relative to the corresponding digits in the true value x_t.

Example: The following approximation x_a has at least m digits of significance.

$$
\begin{aligned}
x_t &= a_1\ a_2\ a_3 \cdot a_4\ a_5\ a_6 \cdots\ a_m\ a_{m+1}\ a_{m+2} \\
|x_t - x_a| &= 0\ 0\ 0 \cdot\ 0\ 0\ 0 \cdots 0\ b_{m+1}\ b_{m+2}
\end{aligned}
$$

Workout-example:

$$x_a = 0.222,\ x_t = \frac{2}{9} \approx 0.222222 \text{ on a 6-digit precision computer,}$$

$|x_t - x_a| = 0.000222 \Rightarrow$ 3 digits of significance.