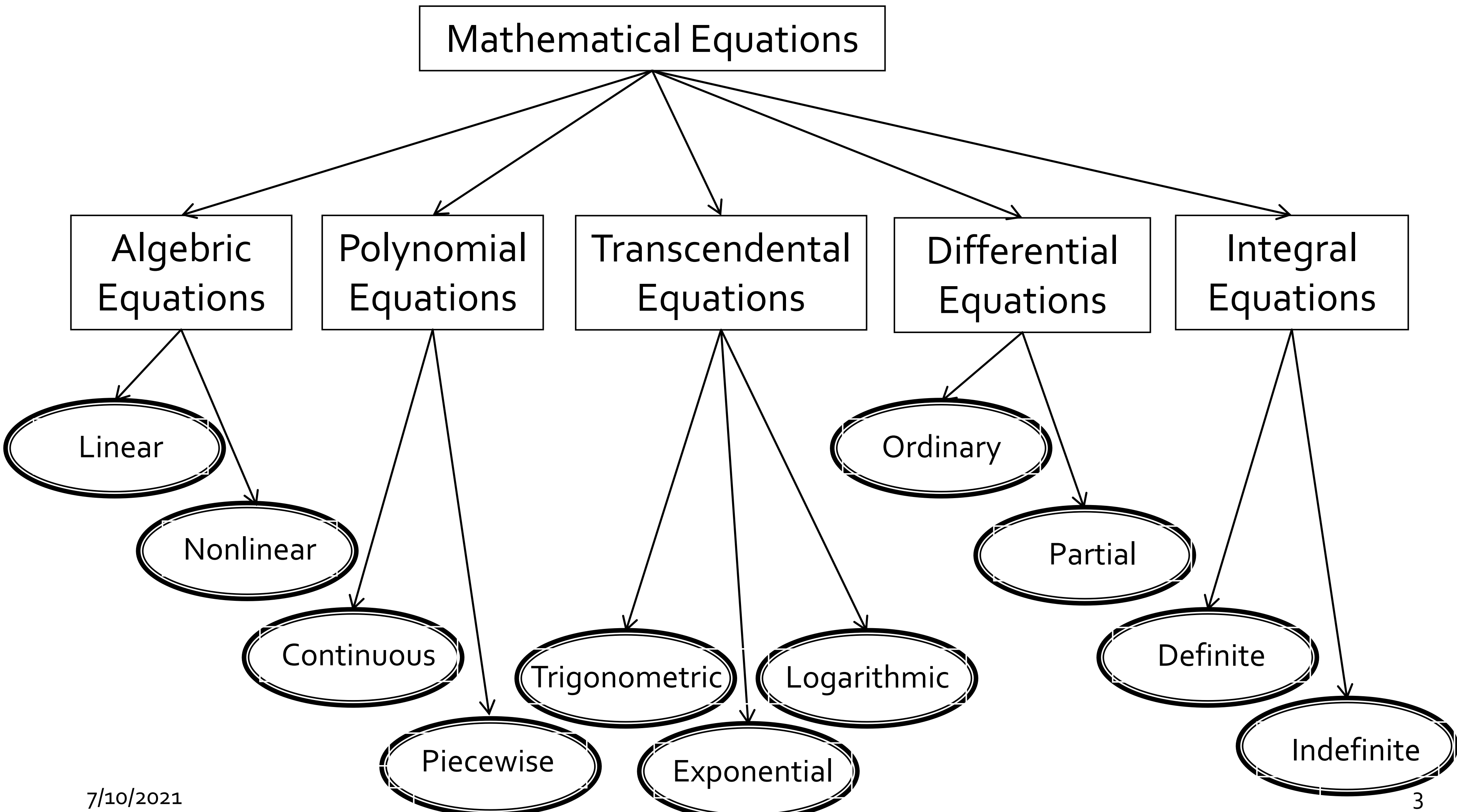
MATH2231: Numerical Methods

Lecture 1
Introduction

Introduction

- To solve mathemetical equations analytically, you may use your experiences in the calculus courses you have studied so far
- But, in most cases, the equations need to be solved approximately using numerical methods.
- Numerical computations play an indispensable role in solving real life mathematical, physical and engineering problems
- Great mathematicians like Gauss, Newton, Langrange, Fourier and many others in the eighteen and nineteeth centuries developed numerical techniques which are still being used
- The advent of computers has, however, enhanced the speed and accuracy of numerical computations

Different forms of mathematical equations



What is numerical computing?

- Numerical computing is an approach for solving complex mathematical problems using only simple arithmatic operations
- The appoarch involves, in most of the cases, formulation of mathematical models of physical situations that can be solved with arithmatic operations
- It requires development, analysis and use of algorithm
- Algorithm is a systematic procedure that solves a problem or a number of problems
- Its efficiency may be measured by the number of steps in the algoritm, the computer time, and the amount of memory (of the computing instrument) that is required

Advantage of Numerical Methods

- The major advantage of numerical methods is that a numerical value can be obtained even when the problem has no “analytical” solution
- The mathematical operations required are essentially addition, subtraction, multiplication, and division plus making comparisons
- It is important to realize that solution by numerical analysis is always numerical
- Analytical methods, on the other hand, usually give a result in terms of mathematical functions that can then be evaluated for specific instances

Scope of Numerical Analysis

- Finding roots of equations
- Solving systems of linear algebric equations
- Interpolation and regression analysis
- Numerical differentiation
- Numerical Integration
- Solution of ordinary differential equations
- Boundary value problems
- Solution of matrix problem

Steps of Solving a Practical Problem

<u>Step #1</u>:

- State the problem clearly, including any simplifying assumptions.

<u>Step #2</u>:

- Develop a mathematical statement of the problem in a form that can be solved for a numerical answer
- This process may involve the use of calculus.
- In some situations, other mathematical procedures may be employed.
- When this statement is a differential equation, appropiate initial conditions and/or boundary conditions must be specified

Steps of Solving a Practical Problem

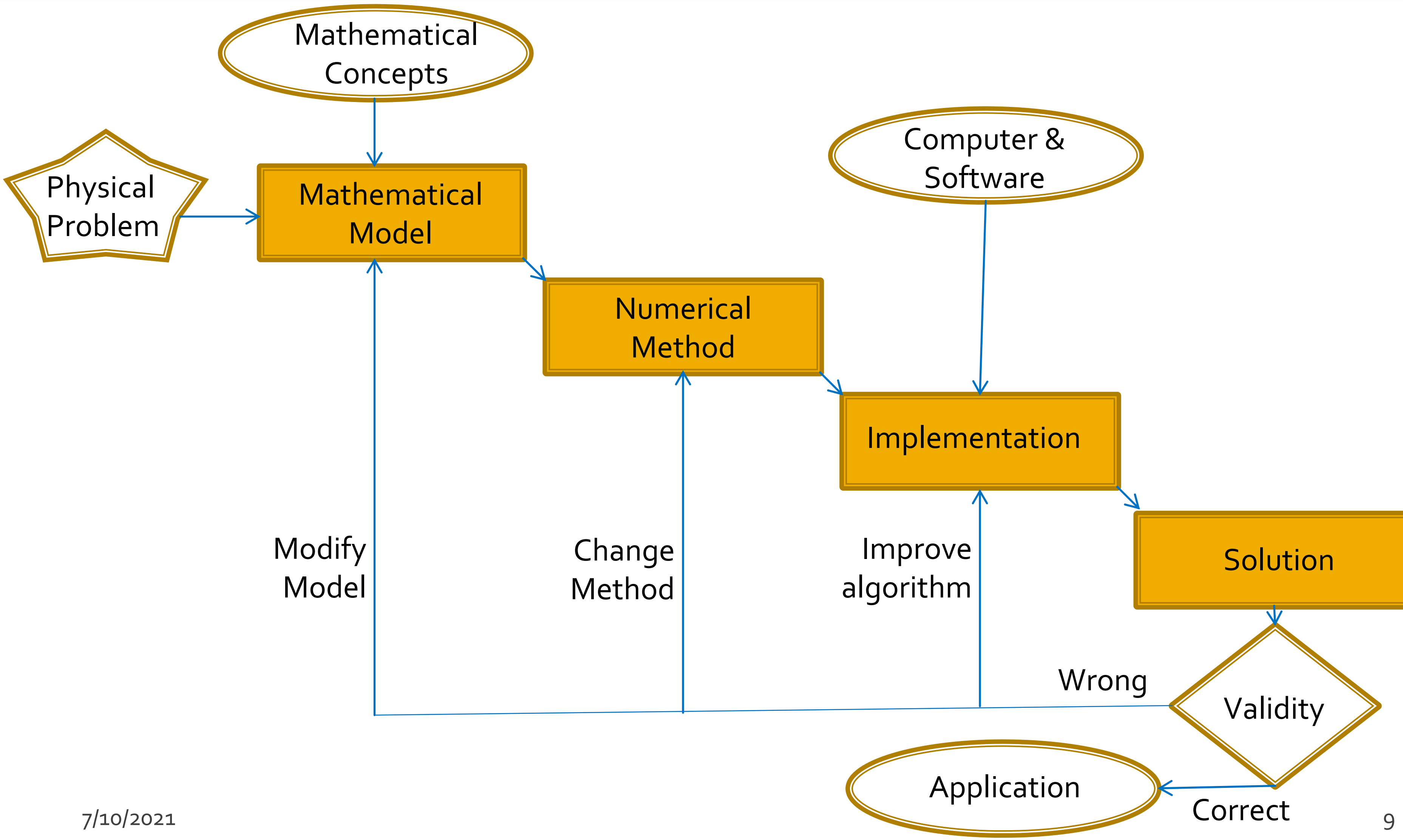
Step #3:

- Solve the equations that are obtained from step #2
- Sometimes the method will be algebric
- But frequently more advanced methods will be needed
- The result of this step is a numerical answer or set of answers

Step #4:

- Interpret the numerical result to arrive at a decision
- This will require experience and understanding of the situation in which the problem is embedded

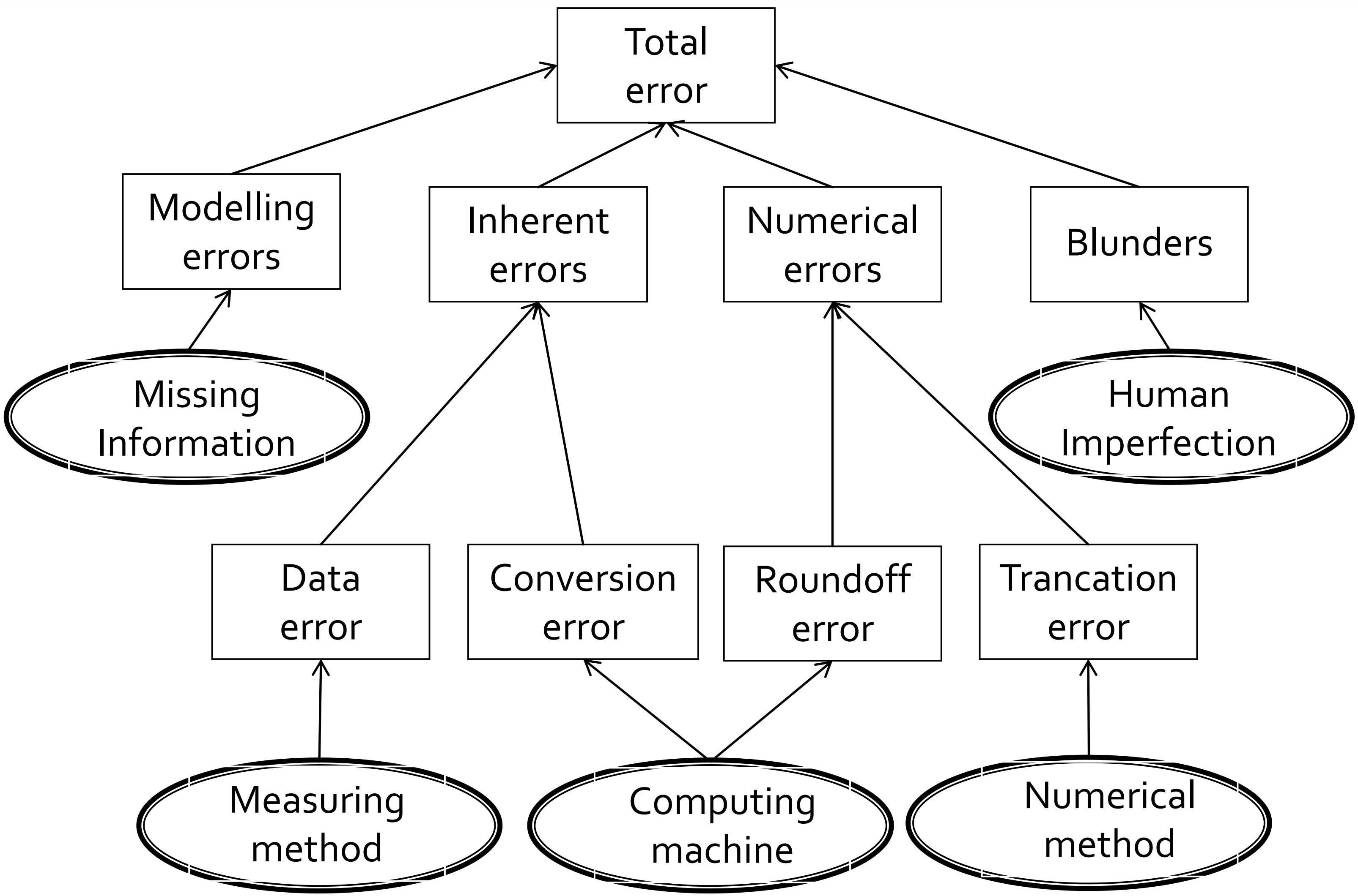
Numerical Computing Process



Accuracy in Numerical Analysis

- Numerical analysis is an approximation, but results can be made as accurately as desired.
- Errors come in a variety forms and sizes; some are avoidable and some are not
- For example, data conversion and roundoff errors can not be avoided, but human errors can be eliminated
- Although certain errors can not be eliminated completely, we must atleast know the bounds of these errors to make use of our final selection
- It is therefore essential to know that how errors arise, how they grow during numerical process and how they affect the accuracy of a solution

Taxonomy of errors



Modelling errors

- In many situations it is impractical to model each of the components accurately and so certain simplifying assumptions are made
- For example, while developing a model for calculaing the force acting on a falling body, we may not be able to estimate the air resistance coefficient (drag coefficient) properly or determine the direction and magnitude of wind force acting on the body and so on
- Since the model is the basic input to the numerical process, no numerical method will provide adequate results if the model is erroneously conceived and formulated

Inherent errors

- Inherent error (also known as input error) contain two components, namely, data errors and conversion errors

Data error

- Data error (also known as emperical error) arises when data for a problem are obtained by some experimental means and are, therefore, of limited accuracy and precision

Conversion error

- Conversion error (also known as representational error) arise due to the limitations of the computer to store the data exactly

Numerical Errors

- Numerical errors (also known as procedural error) are introduced during the process of implementation of a numerical method

Roundoff error

- Roundoff error occur when a fixed number of digits are used to represent exact numbers
- 42.7893 will be rounded off upto 2 decimal digits as 42.79

Trancation error

- Trancation error arise from using an approximation in place of an exact mathematical procedure
- Typically it is the error resulting from the trancation of numerical process
- We often use finite number of terms to estimate the sum of infinite series

Blunders

- Blunders are the errors that are caused due to human imperfection
- Some common type of this error are:
  - Lack of understanding of the problem
  - Wrong assumption
  - Overlooking of some basic assumptions required for formulating the model
  - Error in deriving the mathematical equation or using a model that does not describe adequately the physical system under study
  - Selecting a wrong numerical method for solving the mathematical model
  - Selecting a wrong algorithm for implementing the numerical method
  - Making mistakes in the computer program
  - Mistake in data input
  - Wrong guessing the initial value

Significant Digits

- The following statement describe the notion of significant digits
  - All non-zero digits are significant
  - All zeros occurring between non-zero digits are significant digits
  - Trailing zeros following a decimal point are significant. For example, 3.50, 65.0 and 0.230 have three significant digits
  - Zeros between the decimal point and preceeding a non-zero digit are not significant. For example, the following numbers have four significant digits
    - 0.0001234 (1234×10^{-7})
    - 0.001234 (1234×10^{-6})
    - 0.01234 (1234×10^{-5})
  - When the decimal point is not written, trailing zeros are not considered to be significant, 5600 (56×10^{2}) has two significant digit

Examples of showing the number of significant digits

- 0.0459 has three significant digits
- 4.590 has four significant digits
- 4008 has four significant digits
- 4008.0 has five significant digits
- $1.079X10^3$ has four significant digits
- $1.0790X10^3$ has five significant digits
- $1.07900X10^3$ has six significant digits
- So, how do we differentiate the number of digits correct in 1,000,000 and 1,079,587? Well for that, one may use scientific notation.

  $1,000,000= 1X10^6$; 1 significant digit

  $1,079,587=1.079587X10^6$; 7 significant digits

Relation between accuracy and precision

- Accuracy refers to the number of significant digits in a value. For example, the number 57.396 is accurate to five significant digits
- Precision refers to the number of decimal positions, i.e., the order of magnitude of the last digit in a value. The number 57.396 has a precission of 0.001 or 10^{-3}

An example of a problem created by round off errors

- Twenty-eight Americans were killed on February 25, 1991. An Iraqi Scud hit the Army barracks in Dhahran, Saudi Arabia.
- The patriot defense system had failed to track and intercept the Scud.
- The Patriot defense system consists of an electronic detection device called the range gate.
- It calculates the area in the air space where it should look for a Scud.
- To find out where it should aim next, it calculates the velocity of the Scud and the last time the radar detected the Scud.

The cause for this failure

- Time is saved in a register that has 24 bits length.
- Since the internal clock of the system is measured for every one-tenth of a second, 1/10 is expressed in a 24 bit-register as 0.00011001100110011001100.
- However, this is not an exact representation.
- In fact, it would need infinite numbers of bits to represent 1/10 exactly.
- This caused a error in calculation and the defence system did not work

What is true error?

- True error is the difference between the true value (also called the exact value) and the approximate value.
- True Error = True value – Approximate value

<u>Example 1</u>

- The derivative of a function at a particular value of can be approximately calculated by

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

For $f(x) = 7e^{0.5x}$ and *h=0.3*, find at x=2

a) the approximate value of *f′(x)*

b) the true value of *f′(x)*

c) the true error

True error for the example

- The approximate value is obtained from the previous equation as 10.265
- The true value can be obtainted from the derivative of the function

$$f'(x) = 7 \times 0.5 \times e^{0.5x}$$

- The true value from the above equation is 9.514
- True error = True value – Approximate value = -0.7506

Magnitude of the true error

- The magnitude of true error does not show how bad the error is.
- A true error -0.75061 may seem to be small, but if the function given in the Example 1 were

$$f(x) = 7 \times 10^{-6} e^{0.5x}$$

the true error in calculating *f′(2)* with *h=0.3* would be -0.75061X10^{-6}

- This value of true error is smaller, even when the two problems are similar in that they use the same value of the function argument, *x=2* and the step size, *h=0.3*
- This brings us to the definition of relative true error.

Relative True Error

- Relative true error is denoted by $\in_t$ and is defined as the ratio between the true error and the true value.

$$\text{Relative True Error}, \quad = \in_t = \frac{\text{True Error}}{\text{True value}}$$

- In both the case, the relative true error is 0.758895%

What is approximate error?

- In the previous section, we discussed how to calculate true errors
- Such errors are calculated only if true values are known.
- An example where this would be useful is when one is checking if a program is in working order and you know some examples where the true error is known
- But mostly we will not have the luxury of knowing true values as why would you want to find the approximate values if you know the true values
- So when we are solving a problem numerically, we will only have access to approximate values
- We need to know how to quantify error for such cases

Definition of Approximate Error

- Approximate error is defined as the difference between the present approximation and previous approximation.

Approximate Error= Present Approximation – Previous Approximation

- Relative approximate error is defined as the ratio between the approximate error and the present approximation
- In the previous exmple if we find the value of the derivative of the function at h=0.3 and h=0.15, the values 10.265 and 9.8799 respectively
- So the relative approximate error in percentage is -3.8942%

Use relative approximate errors to minimize the error

- In a numerical method that uses iterative process, a user can calculate relative approximate error at the end of each iteration
- The user may pre-specify a minimum acceptable tolerance called the pre-specified tolerance
- If the absolute relative approximate error is less than or equal to the pre-specified tolerance, then the acceptable error has been reached and no more iterations would be required

Reference

- Numerical Methods with Applications: http://mathforcollege.com/nm/topics/textbook_index.html

Thanks