Numerical Methods

MTH4002

Lecture 01: Introduction

Dr. Kundan Kumar
Associate Professor
Department of ECE

Faculty of Engineering (ITER)
S'O'A Deemed to be University, Bhubaneswar, India-751030

Outline

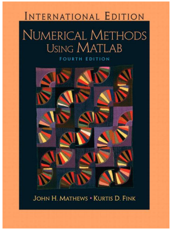
Google Classrooms

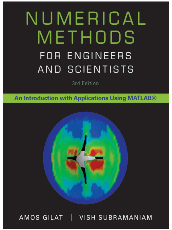
- All the communication will be through Google Classroom:
  - Course materials
  - Assignments
  - Announcements and Notices
- Join the course at https://classroom.google.com/
- Class Code for CSE, Section-F:
  - Joining classroom request will be sent to the email ids or student may joint through the following course code.

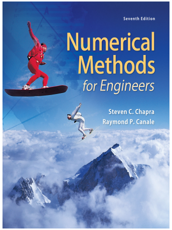
6hkyx6a

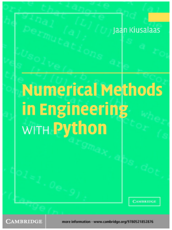
- Would you like to code in Python for the topics to be covered in Numerical Methods?

Text Books









Text Books:

- Numerical Methods Using MATLAB by Matthews and Fink, Pearson Education.
- Numerical Methods for Engineers and Scientists: An Introduction with Applications using MATLAB, Amos Gilat and Vish Subramaniam, Wiley.

Credits:

- 3 credits course,
  - 2 Classes/week (1hr/Class),
  - 1 Lab/week (2hr/Lab).

Grading pattern

- Grading pattern: 2

Attendance	:	5 Marks
2 Quizzes	:	10 Marks
Assignments	:	10 Marks
Mid-term examination	:	15 Marks
Total Internal	:	40 Marks

In lab exam	:	15 Marks
Theory exam	:	45 Marks
Total External	:	60 Marks

Regarding Attendance and Home Assignments

- Attendance will be taken by calling students name or last three digit of the registration number.
- Alternatively, attendance will be taken through the Google Classroom.
- Every weekend, home assignment will be shared and the solution is to be uploaded in the Google classroom before deadline.

In this course we are going to cover

- Preliminaries
- The Solution of Nonlinear Equations $f(x) = 0$
- The Solution of Linear Systems $AX = B$
- Interpolation and Polynomial Approximation
- Curve Fitting
- Numerical Differentiation
- Numerical Integration
- Numerical Optimization
- Solution of Differential Equations
- Solution of Partial Differential Equations
- Eigenvalues and Eigenvectors.

Why Numerical Methods?

- Numerical methods can be used for solution of complex problems.
- Make easier to understand and use “canned” software with insight.
- Solutions, if not already available, can be created.
- Using numerical methods, we can efficiently learning to use computers.
- Reinforce your understanding of mathematics.

Students will learn

1. A common numerical methods and how they are used to obtain approximate solutions to otherwise intractable mathematical problems.
2. To apply numerical methods to obtain approximate solutions to mathematical problems.
3. To derive numerical methods for various mathematical operations and tasks
   - interpolation,
   - differentiation,
   - integration,
   - the solution of linear and nonlinear equations
   - the solution of differential equations.
4. Analyse and evaluate the accuracy of common numerical methods.
5. Implement numerical methods in MATLAB/OCTAVE.
6. Write efficient, well-documented MATLAB/OCTAVE code and present numerical results in an informative way.

A problem and its solution

Better to start with a problem

- An engineering problem: forces acting on a falling object

Better to start with a problem

- An engineering problem: forces acting on a falling object

Better to start with a problem

- An engineering problem: forces acting on a falling object

- The problem can be simplified using Newton's law of motion.

Better to start with a problem

- An engineering problem: forces acting on a falling object

- The problem can be simplified using Newton's law of motion.
  - The time rate of change of momentum of a body is equal to the resultant force acting on it.

Better to start with a problem

- An engineering problem: forces acting on a falling object

- The problem can be simplified using Newton's law of motion.
  - The time rate of change of momentum of a body is equal to the resultant force acting on it.

$$F = F_D + F_U$$

Better to start with a problem

- An engineering problem: forces acting on a falling object

- The problem can be simplified using Newton's law of motion.
  - The time rate of change of momentum of a body is equal to the resultant force acting on it.

$$F = F_D + F_U$$

$$F = mg - cv$$

Better to start with a problem

- An engineering problem: forces acting on a falling object

- The problem can be simplified using Newton's law of motion.
  - The time rate of change of momentum of a body is equal to the resultant force acting on it.

$$F = F_D + F_U$$

$$F = mg - cv$$

$$ma = mg - cv \qquad \left[a = \frac{dv}{dt}\right]$$

Better to start with a problem

- An engineering problem: forces acting on a falling object

- The problem can be simplified using Newton's law of motion.
  - The time rate of change of momentum of a body is equal to the resultant force acting on it.

$$F = F_D + F_U$$

$$F = mg - cv$$

$$ma = mg - cv \qquad \left[a = \frac{dv}{dt}\right]$$

$$\Rightarrow \qquad \frac{dv}{dt} = \frac{mg - cv}{m}$$

Better to start with a problem

- An engineering problem: forces acting on a falling object

- The problem can be simplified using Newton's law of motion.
    - The time rate of change of momentum of a body is equal to the resultant force acting on it.

$$F = F_D + F_U$$

$$F = mg - cv$$

$$ma = mg - cv \qquad \left[a = \frac{dv}{dt}\right]$$

$$\Rightarrow \quad \frac{dv}{dt} = \frac{mg - cv}{m}$$

$$\Rightarrow \quad \frac{dv}{dt} = g - \frac{c}{m}v$$

Better to start with a problem

- An engineering problem: forces acting on a falling object

$$\boxed{\frac{dv}{dt} = g - \frac{c}{m}v} \tag{1}$$

Better to start with a problem

- An engineering problem: forces acting on a falling object

$$\boxed{\frac{dv}{dt} = g - \frac{c}{m}v} \tag{1}$$

- Above equation is a model that relates the acceleration of a falling object to the forces acting on it.

Better to start with a problem

- An engineering problem: forces acting on a falling object

$$\boxed{\frac{dv}{dt} = g - \frac{c}{m}v} \tag{1}$$

- Above equation is a model that relates the acceleration of a falling object to the forces acting on it.
- It is a differential equation because it is written in terms of the differential rate of change (dv/dt) of the variable that we are interested in predicting.

Better to start with a problem

- An engineering problem: forces acting on a falling object

$$\boxed{\frac{dv}{dt} = g - \frac{c}{m}v} \tag{1}$$

- Above equation is a model that relates the acceleration of a falling object to the forces acting on it.
- It is a differential equation because it is written in terms of the differential rate of change (dv/dt) of the variable that we are interested in predicting.
- If the object is initially at rest ($v = 0$ at $t = 0$), the solution of the equation

$$\boxed{v(t) = \frac{gm}{c}\left(1 - e^{-(c/m)t}\right)} \tag{2}$$

Better to start with a problem

- Can you derive $v(t) = \frac{gm}{c}\left(1 - e^{-(c/m)t}\right)$ from $\frac{dv}{dt} = g - \frac{c}{m}v$?
  hint: $\frac{dv}{dt} = g - \frac{c}{m}v$ is a first order linear differential equation.

Mathematical Model

$$\boxed{v(t) = \frac{gm}{c}\left(1 - e^{-(c/m)t}\right)} \tag{3}$$

- A mathematical model can be broadly defined as

$$\boxed{\text{Dependent variable} = f(\text{independent variable, parameters, forcing action})}$$

where

- dependent variable $\rightarrow$ a characteristic that usually reflects the behavior or state of the system, e.g., $v(t)$
- independent variables $\rightarrow$ are usually dimensions, such as time and space, along which the system's behavior is being determined, e.g., t
- parameters $\rightarrow$ the reflective of the system's properties, e.g., m, c
- forcing functions $\rightarrow$ external influences acting upon the system, e.g., g

Example

Example 01: A parachutist of mass 68.1 kg jumps out of a stationary hot air balloon. Use Eq. (2) to compute velocity prior to opening the parachute. The drag coefficient is equal to 12.5 kg/s.

Example

Example 01: A parachutist of mass 68.1 kg jumps out of a stationary hot air balloon. Use Eq. (2) to compute velocity prior to opening the parachute. The drag coefficient is equal to 12.5 kg/s.

Solution: Given values, $m = 68.1$ kg, $g = 9.81$ m/s, and $c = 12.5$ kg/s, put in Eq. (2), we get

Example

Example 01: A parachutist of mass 68.1 kg jumps out of a stationary hot air balloon. Use Eq. (2) to compute velocity prior to opening the parachute. The drag coefficient is equal to 12.5 kg/s.

Solution: Given values, $m = 68.1$ kg, $g = 9.81$ m/s, and $c = 12.5$ kg/s, put in Eq. (2), we get

$$v(t) = \frac{9.81 \times 68.1}{12.5}\left(1 - e^{-(12.5/68.1)t}\right)$$
$$= 53.44(1 - e^{-0.18355t})$$

which can be used to compute the velocity attained after time t.

Example

Example 01: A parachutist of mass 68.1 kg jumps out of a stationary hot air balloon. Use Eq. (2) to compute velocity prior to opening the parachute. The drag coefficient is equal to 12.5 kg/s.

Solution: Given values, $m = 68.1$ kg, $g = 9.81$ m/s, and $c = 12.5$ kg/s, put in Eq. (2), we get

$$v(t) = \frac{9.81 \times 68.1}{12.5}\left(1 - e^{-(12.5/68.1)t}\right)$$
$$= 53.44(1 - e^{-0.18355t})$$

which can be used to compute the velocity attained after time t.

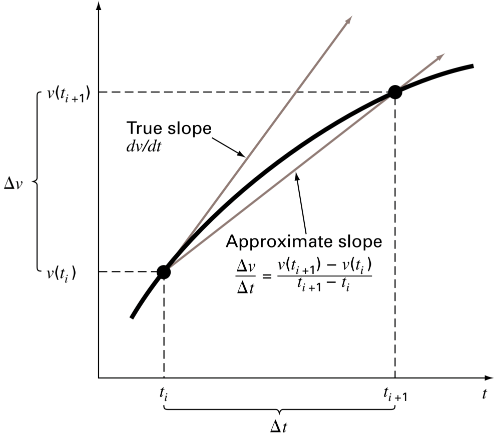
t(s)	v (m/s)
0	0.00
2	16.42
4	27.80
6	35.68
8	41.14
10	44.92
12	47.54
∞	53.44

Analytical Solution

- The problem, just we discussed, is solved using analytical approach, which gives analytical or exact solution.
- Drawbacks of the analytical approach:
  - Sometimes difficult to solve
  - Many problems cannot be solved using this approach.
  - How to solve a problem using computer?
- We need to adopt one or more advanced techniques to find out the solution.
- In many of these cases, the only alternative is to develop a numerical solution that approximates the exact solution.

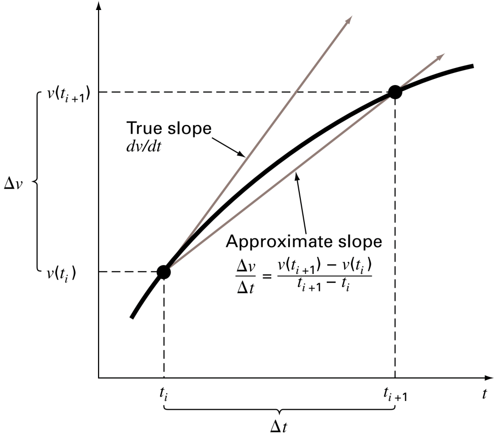
Approximate solution

- Now let us try to reformulate the problem to find the approximate solution close to exact solution.



Approximate solution

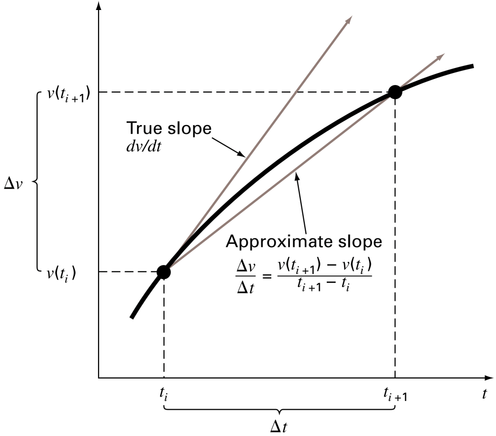
- Now let us try to reformulate the problem to find the approximate solution close to exact solution.



- This can be illustrated for Newton's second law by realizing that the time rate of change of velocity can be approximated by

Approximate solution

- Now let us try to reformulate the problem to find the approximate solution close to exact solution.



- This can be illustrated for Newton's second law by realizing that the time rate of change of velocity can be approximated by

$$\boxed{\frac{dv}{dt} \cong \frac{\Delta v}{\Delta t} = \frac{v(t_{i+1}) - v(t_i)}{t_{i+1} - t_i}} \tag{4}$$

where Δv and Δt are differences in velocity and time, respectively, computed over finite intervals, $v(t_i)$ is velocity at an interval time t_i, and $v(t_{t+1})$ is the velocity at some later time t_{i+1}.

Approximate solution

- Note that $dv/dt \cong \Delta v/\Delta t$ is approximate because Δt is finite.
- Remember from calculus that

$$\frac{dv}{dt} = \lim_{\Delta t \to 0} \frac{\Delta v}{\Delta t}$$

- We can substitute this value in Eq. (1), we get

$$\frac{v(t_{i+1}) - v(t_i)}{t_{i+1} - t_i} = g - \frac{c}{m} v(t_i)$$

$$\boxed{v(t_{i+1}) = v(t_i) + \left[g - \frac{c}{m} v(t_i)\right](t_{i+1} - t_i)} \tag{5}$$

- If the initial velocity at t_i is available then we can easily compute velocity at t_{i+1}.

Example

Example 02: A parachutist of mass 68.1 kg jumps out of a stationary hot air balloon. Compute the velocity attained by parachutist after t s using approximation approach. Employ a step size of 2 s for the calculation.

Example

Example 02: A parachutist of mass 68.1 kg jumps out of a stationary hot air balloon. Compute the velocity attained by parachutist after t s using approximation approach. Employ a step size of 2 s for the calculation.

Solution: Assume at $t_0 = 0$, $v(t_0) = 0$. Given $t_{i+1} - t_i = 2$ (step size). Compute velocity $v(t_1)$ at t_1 as

Example

Example 02: A parachutist of mass 68.1 kg jumps out of a stationary hot air balloon. Compute the velocity attained by parachutist after t s using approximation approach. Employ a step size of 2 s for the calculation.

Solution: Assume at $t_0 = 0$, $v(t_0) = 0$. Given $t_{i+1} - t_i = 2$ (step size). Compute velocity $v(t_1)$ at t_1 as

$$\begin{aligned} v(t_1) =& v(t_0) + \left[9.81 - \frac{12.5}{68.1} v(t_0)\right] \times 2 \\ v(t_1) =& 0 + \left[9.81 - \frac{12.5}{68.1}(0)\right] \times 2 = 19.62 \text{ m/s} \end{aligned}$$

Example

Example 02: A parachutist of mass 68.1 kg jumps out of a stationary hot air balloon. Compute the velocity attained by parachutist after t s using approximation approach. Employ a step size of 2 s for the calculation.

Solution: Assume at $t_0 = 0$, $v(t_0) = 0$. Given $t_{i+1} - t_i = 2$ (step size). Compute velocity $v(t_1)$ at t_1 as

$$\begin{aligned} v(t_1) &= v(t_0) + \left[9.81 - \frac{12.5}{68.1} v(t_0)\right] \times 2 \\ v(t_1) &= 0 + \left[9.81 - \frac{12.5}{68.1}(0)\right] \times 2 = 19.62 \text{ m/s} \end{aligned}$$

For the next interval (from $t = 2$ to 4s),

Example

Example 02: A parachutist of mass 68.1 kg jumps out of a stationary hot air balloon. Compute the velocity attained by parachutist after t s using approximation approach. Employ a step size of 2 s for the calculation.

Solution: Assume at $t_0 = 0$, $v(t_0) = 0$. Given $t_{i+1} - t_i = 2$ (step size). Compute velocity $v(t_1)$ at t_1 as

$$v(t_1) = v(t_0) + \left[9.81 - \frac{12.5}{68.1} v(t_0)\right] \times 2$$

$$v(t_1) = 0 + \left[9.81 - \frac{12.5}{68.1}(0)\right] \times 2 = 19.62 \text{ m/s}$$

For the next interval (from $t = 2$ to 4s),

$$v(t_2) = v(t_1) + \left[9.81 - \frac{12.5}{68.1} v(t_1)\right] \times 2$$

$$v(t_2) = 19.62 + \left[9.81 - \frac{12.5}{68.1}(19.62)\right] \times 2 = 32.04 \text{ m/s}$$

Example

Example 02: A parachutist of mass 68.1 kg jumps out of a stationary hot air balloon. Compute the velocity attained by parachutist after t s using approximation approach. Employ a step size of 2 s for the calculation.

Solution: Assume at $t_0 = 0$, $v(t_0) = 0$. Given $t_{i+1} - t_i = 2$ (step size). Compute velocity $v(t_1)$ at t_1 as

$$v(t_1) = v(t_0) + \left[9.81 - \frac{12.5}{68.1} v(t_0)\right] \times 2$$

$$v(t_1) = 0 + \left[9.81 - \frac{12.5}{68.1}(0)\right] \times 2 = 19.62 \text{ m/s}$$

For the next interval (from $t = 2$ to 4s),

$$v(t_2) = v(t_1) + \left[9.81 - \frac{12.5}{68.1} v(t_1)\right] \times 2$$

$$v(t_2) = 19.62 + \left[9.81 - \frac{12.5}{68.1}(19.62)\right] \times 2 = 32.04 \text{ m/s}$$

Example

Example 02: A parachutist of mass 68.1 kg jumps out of a stationary hot air balloon. Compute the velocity attained by parachutist after t s using approximation approach. Employ a step size of 2 s for the calculation.

Solution: Assume at $t_0 = 0$, $v(t_0) = 0$. Given $t_{i+1} - t_i = 2$ (step size). Compute velocity $v(t_1)$ at t_1 as

t(s)	v (m/s)
0	0.00
2	19.62
4	32.04
6	39.90
8	44.87
10	48.02
12	50.01
∞	53.44

$$v(t_1) = v(t_0) + \left[9.81 - \frac{12.5}{68.1} v(t_0)\right] \times 2$$

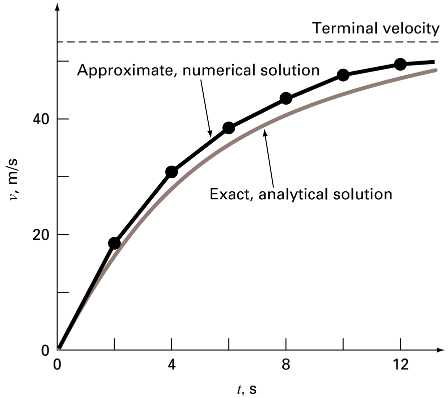
$$v(t_1) = 0 + \left[9.81 - \frac{12.5}{68.1}(0)\right] \times 2 = 19.62 \text{ m/s}$$

For the next interval (from $t = 2$ to 4s),

$$v(t_2) = v(t_1) + \left[9.81 - \frac{12.5}{68.1} v(t_1)\right] \times 2$$

$$v(t_2) = 19.62 + \left[9.81 - \frac{12.5}{68.1}(19.62)\right] \times 2 = 32.04 \text{ m/s}$$

Numerical Methods



- Formally, numerical methods used for calculating approximated solutions to problems that cannot be solved (or are difficult to solve) analytically.
- Numerical methods are techniques by which mathematical problems are formulated so that they can be solved with arithmetic operations.
- Used to develop fast and efficient digital computations.
- Numerical solutions can be very accurate but in general are not exact. In general, they are always associated with some error.

Numerical vs Analytical Methods

- Analytical method is a non-computer method; however, Numerical method can be implemented on computers.
- Numerical methods are extremely powerful problem-solving tools compare to analytical methods.

Numerical vs Analytical Methods

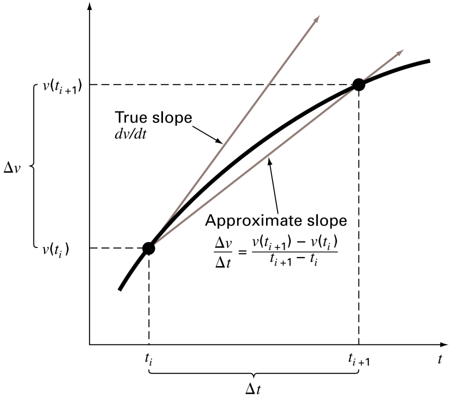
- Analytical method is a non-computer method; however, Numerical method can be implemented on computers.
- Numerical methods are extremely powerful problem-solving tools compare to analytical methods.
- Capable of handling large systems of equations, non-linearities, and complicated geometries that are often impossible to solve analytically.
- Graphical solutions were used to characterize the behavior of systems. Although graphical techniques can often be used to solve complex problems, the results are not very precise.

Numerical vs Analytical Methods

- Analytical method is a non-computer method; however, Numerical method can be implemented on computers.
- Numerical methods are extremely powerful problem-solving tools compare to analytical methods.
- Capable of handling large systems of equations, non-linearities, and complicated geometries that are often impossible to solve analytically.
- Graphical solutions were used to characterize the behavior of systems. Although graphical techniques can often be used to solve complex problems, the results are not very precise.
- Numerical methods provide a vehicle for you to reinforce your understanding of mathematics and use of computers because a function of numerical methods can reduce higher mathematics to basic arithmetic operations.

Error in Numerical Solutions

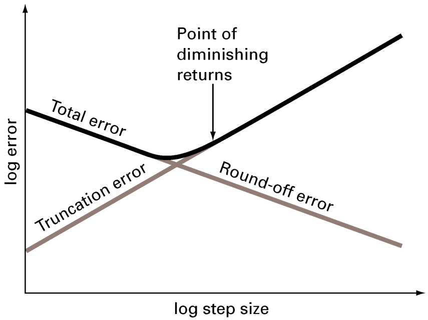
Error in Numerical Solutions



- In general, numerical solutions are always associated with some error.
- We have seen that the numerical method captures the essential features of the exact solution.
- However, because we have employed straight-line segments in numerical method to approximate a continuously curving function, there is some discrepancy between the two results.
- One way to minimize such discrepancies is to use a smaller step size.

Error in numerical solutions

- Two kinds of errors are introduced when numerical methods are used for solving a problem.



- Round-off errors: Occurs because of the way that machine (or digital computers) store the number and execute numerical operations.
- Truncation errors: Introduced by the numerical method.

Round-off Error

- A mathematical quantity or real number x is not always stored in the real form.
- Instead, a machine (or computer) store or process a number in a standard form to support a trade-off between range and precision.

$$\text{mantissa} \times 10^{\text{exponent}} \qquad or \qquad \text{mantissa} \times 2^{\text{exponent}}$$

- A computer's representation of real numbers is limited to the fixed precision of the mantissa. True values are sometimes not stored exactly by a computers representation.
- Numbers are represented on a computer by a finite number of bits. Consequently, real numbers that have a mantissa longer than the number of bits that are available for representing them have to be shortened.

Round-off Error

- The actual number that is stored in the computer may undergo chopping or rounding of the last digit.

Round-off Error

- The actual number that is stored in the computer may undergo chopping or rounding of the last digit.
- Chopping off the extra digits:
  - In chopping, the digits in the mantissa beyond the length, that can be stored, are simply left out.
  - For illustration, consider the number $2/3$. In decimal form with four significant digits, 2/3 can be written as 0.6666.

Round-off Error

- The actual number that is stored in the computer may undergo chopping or rounding of the last digit.
- Chopping off the extra digits:
  - In chopping, the digits in the mantissa beyond the length, that can be stored, are simply left out.
  - For illustration, consider the number $2/3$. In decimal form with four significant digits, $2/3$ can be written as 0.6666.
- Rounding:
  - In rounding, the last digit, that is stored, is rounded. Ex: $2/3$ can be written as 0.6667

Round-off Error

- The actual number that is stored in the computer may undergo chopping or rounding of the last digit.
- Chopping off the extra digits:
  - In chopping, the digits in the mantissa beyond the length, that can be stored, are simply left out.
  - For illustration, consider the number $2/3$. In decimal form with four significant digits, 2/3 can be written as 0.6666.
- Rounding:
  - In rounding, the last digit, that is stored, is rounded. Ex: 2/3 can be written as 0.6667
- Either way, such chopping and rounding of real numbers lead to errors in numerical computations, especially when many operations are performed. This is called Round-off error. (More details needed)

Trucation Error

- Truncation error usually refers to errors introduced when a more complicated mathematical expression is “replaced” with a more elementary formula.

Trucation Error

- Truncation error usually refers to errors introduced when a more complicated mathematical expression is "replaced" with a more elementary formula.
- Let us consider an example of the infinite Taylor series expansion of sinusoidal function

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \frac{x^{11}}{11!} + \cdots \tag{6}$$

might be replaced with just the first one or two terms.

Trucation Error

- Truncation error usually refers to errors introduced when a more complicated mathematical expression is "replaced" with a more elementary formula.
- Let us consider an example of the infinite Taylor series expansion of sinusoidal function

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \frac{x^{11}}{11!} + \cdots \tag{6}$$

  might be replaced with just the first one or two terms.
- The truncation error is dependent on the specific numerical method or algorithm used to solve a problem.

Truncation Error

- For example, if only the first term is used in Taylor series expansion of sinusoidal function

$$\sin\left(\frac{\pi}{6}\right) = \frac{\pi}{6} = 0.5235988$$

$$E_{Trunc} = 0.5 - 0.5235988 = -0.0235988$$

Truncation Error

- For example, if only the first term is used in Taylor series expansion of sinusoidal function

$$\sin\left(\frac{\pi}{6}\right) = \frac{\pi}{6} = 0.5235988$$

$$E_{Trunc} = 0.5 - 0.5235988 = -0.0235988$$

- If two terms of the Taylor's series are used

Truncation Error

- For example, if only the first term is used in Taylor series expansion of sinusoidal function

$$\sin\left(\frac{\pi}{6}\right) = \frac{\pi}{6} = 0.5235988$$

$$E_{Trunc} = 0.5 - 0.5235988 = -0.0235988$$

- If two terms of the Taylor's series are used

$$\sin\left(\frac{\pi}{6}\right) = \frac{\pi}{6} - \frac{(\pi/6)^3}{3!} = 0.4996742$$

$$E_{Trunc} = 0.5 - 0.4996742 = 0.0003258$$

Exercise Problem

Question 01: The Taylor series expansion of $\cos(x)$ is given by:

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \frac{x^8}{8!} - \frac{x^{10}}{10!} + \dots \tag{7}$$

Use the first three terms to calculate the value of $\cos(\pi/4)$. Use the decimal format with six significant digits (apply rounding at each step). Calculate the truncation error.

Solution: Can you do it?

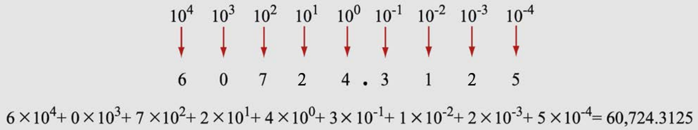
Number Representation on a Computer

Representation of numbers on a computer

- Numbers can be represented in various forms using bases such as $10,\ 2,\ 8$, etc.

Representation of numbers on a computer

- Numbers can be represented in various forms using bases such as $10,\ 2,\ 8$, etc.
  - Decimal representation: Uses ten digits $0, 1, ..., 9$. A number is written by a sequence of digits that correspond to multiples of powers of 10.



Representation of numbers on a computer

- Numbers can be represented in various forms using bases such as $10,\ 2,\ 8$, etc.
  - Decimal representation: Uses ten digits $0, 1, ..., 9$. A number is written by a sequence of digits that correspond to multiples of powers of 10.

10^4	10^3	10^2	10^1	10^0		10^{-1}	10^{-2}	10^{-3}	10^{-4}
6	0	7	2	4	.	3	1	2	5

$$6\times10^4+0\times10^3+7\times10^2+2\times10^1+4\times10^0+3\times10^{-1}+1\times10^{-2}+2\times10^{-3}+5\times10^{-4}=60{,}724.3125$$

  - Binary representation

2^4	2^3	2^2	2^1	2^0		2^{-1}	2^{-2}	2^{-3}
1	0	0	1	1	.	1	0	1

$$1\times 2^4+0\times2^3+0\times2^2+1\times2^1+1\times2^0+1\times2^{-1}+0\times2^{-2}+1\times2^{-3}$$

$$1\times16+0\times8+0\times4+1\times2+1\times1+1\times0.5+0\times0.25+1\times0.125 = 19.625$$

Base 10	Base 2			
	2^3	2^2	2^1	2^0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0

Representation of numbers on a computer

- Can you write the number 60,724.3125 in binary form?

2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}		2^{-1}	2^{-2}	2^{-3}	2^{-4}
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓		↓	↓	↓	↓
1	1	1	0	1	1	0	1	0	0	1	1	0	1	0	0	.	0	1	0	1

$$1\times 2^{15}+1\times 2^{14}+1\times 2^{13}+0\times 2^{12}+1\times 2^{11}+1\times 2^{10}+0\times 2^{9}+1\times 2^{8}+0\times 2^{7}+0\times 2^{6}+1\times 2^{5}$$
$$+1\times 2^{4}+0\times 2^{3}+1\times 2^{2}+0\times 2^{1}+0\times 2^{0}+0\times 2^{-1}+1\times 2^{-2}+0\times 2^{-3}+1\times 2^{-4}=60{,}724.3125$$

Representation of numbers on a computer

- Can you write the number 60,724.3125 in binary form?

2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}		2^{-1}	2^{-2}	2^{-3}	2^{-4}
1	1	1	0	1	1	0	1	0	0	1	1	0	1	0	0	.	0	1	0	1

$$1\times 2^{15}+1\times 2^{14}+1\times 2^{13}+0\times 2^{12}+1\times 2^{11}+1\times 2^{10}+0\times 2^{9}+1\times 2^{8}+0\times 2^{7}+0\times 2^{6}+1\times 2^{5}$$
$$+1\times 2^{4}+0\times 2^{3}+1\times 2^{2}+0\times 2^{1}+0\times 2^{0}+0\times 2^{-1}+1\times 2^{-2}+0\times 2^{-3}+1\times 2^{-4}=60{,}724.3125$$

- Computers store and process numbers in binary (base 2) form. Each binary digit (one or zero) is called a bit (for binary digit).

Representation of numbers on a computer

- Can you write the number 60,724.3125 in binary form?

2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}		2^{-1}	2^{-2}	2^{-3}	2^{-4}
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓		↓	↓	↓	↓
1	1	1	0	1	1	0	1	0	0	1	1	0	1	0	0	.	0	1	0	1

$$1\times 2^{15}+1\times 2^{14}+1\times 2^{13}+0\times 2^{12}+1\times 2^{11}+1\times 2^{10}+0\times 2^{9}+1\times 2^{8}+0\times 2^{7}+0\times 2^{6}+1\times 2^{5}$$
$$+1\times 2^{4}+0\times 2^{3}+1\times 2^{2}+0\times 2^{1}+0\times 2^{0}+0\times 2^{-1}+1\times 2^{-2}+0\times 2^{-3}+1\times 2^{-4}=60{,}724.3125$$

- Computers store and process numbers in binary (base 2) form. Each binary digit (one or zero) is called a bit (for binary digit).
- Scientific Notation: A standard way to present a real number, called scientific notation, is obtained by shifting the decimal point and supplying an appropriate power of 10.

$$\begin{aligned} 0.0000747 &= 7.47 \times 10^{-5} \\ 31.4159265 &= 3.14159265 \times 10 \\ 9,700,000,000 &= 9.7 \times 10^{9} \end{aligned} \tag{8}$$

Floating point representation

- To accommodate large and small numbers, real numbers are written in floating-point representation.
- Decimal floating point representation has the form

$$d.dddddd \times 10^p \tag{9}$$

The decimal floating point representation also known as scientific notation. The number $0.dddddd$ is called the mantissa and p is called exponent.

Floating point represenation

Example 04: Floating Point Addition

Add the following two decimal numbers in scientific notation:

8.70×10^{-1} with 9.95×10^{1}

Floating point representation

- Binary floating point representation has the form:

$$1.bbbbbb \times 2^{bbb} \qquad (b \text{ is a binary digit}) \tag{10}$$

- In this form, the mantissa is $.bbbbbb$, and the power of 2 is called the exponent.
- Both the mantissa and the exponent are written in a binary form.
- The form in Eq. (4) is obtained by normalizing the number (when it is written in the decimal form) with respect to the largest power of 2 that is smaller than the number itself.
- To store numbers accurately, computers must have floating-point binary numbers with at least 24 binary bits used for the mantissa; this translates to about seven decimal places. If a 32-bit mantissa is used, numbers with nine decimal places can be stored.

Floating point representation

Example 04: Write the number 50 in binary floating point representation.

Example

Example 05: Perform $0.5 + (-0.4375)$ {Addition in binary}

Exercise Problem

Question 2: Compute $\left(\frac{1}{10}+\frac{1}{5}\right)+\frac{1}{6}$ if a computer had only a 4-bit mantissa and Exponent of $n \in \{-3,-2,-1,0,1,2,3,4\}$.

Computer Floating-Point Numbers

- Computers have both an integer mode and a floating-point mode for representing numbers.
- The integer mode is used for performing calculations that are known to be integer valued and has limited usage for numerical analysis.
- Floating-point numbers are used for scientific and engineering applications.

Computer Floating-Point Numbers

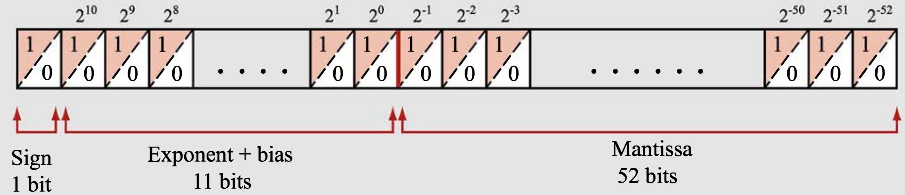
- Computers have both an integer mode and a floating-point mode for representing numbers.
- The integer mode is used for performing calculations that are known to be integer valued and has limited usage for numerical analysis.
- Floating-point numbers are used for scientific and engineering applications.
- The computer stores the values of the exponent and mantissa separately, while the leading 1 in front of the decimal point is not stored.

Computer Floating-Point Numbers

- Computers have both an integer mode and a floating-point mode for representing numbers.
- The integer mode is used for performing calculations that are known to be integer valued and has limited usage for numerical analysis.
- Floating-point numbers are used for scientific and engineering applications.
- The computer stores the values of the exponent and mantissa separately, while the leading 1 in front of the decimal point is not stored.
- According to the IEEE-754 standard (1985), computers store numbers and carry out calculations in
  - Single precision (32 bit representation)
  - Double precision (64 bit representation)

Storing a number in computer memory: IEEE-754 standard

- In single precision, the numbers are stored in a string of 32 bits (4 bytes), and in double precision in a string of 64 bits (8 bytes).
- In both cases, the first bit stores the sign (0 corresponds to $+$ and 1 corresponds to $-$) of the number.
- The next 8 bits in single precision (11 bits in double precision) are used for storing the exponent.
- The following 23 bits in single precision (52 bits in double precision) are used for storing the mantissa.



Storing a number in computer memory: IEEE-754 standard

- The value of the mantissa is in a binary form. The value of the exponent is entered with a bias. A bias means that a constant is added to the value of the exponent.

Storing a number in computer memory: IEEE-754 standard

- The value of the mantissa is in a binary form. The value of the exponent is entered with a bias. A bias means that a constant is added to the value of the exponent.
- The bias is introduced in order to avoid using one of the bits for the sign of the exponent (since the exponent can be positive or negative).
- In binary notation, the largest number that can be written with 11 bits is 2047 (when all 11 digits are 1).

Storing a number in computer memory: IEEE-754 standard

- The value of the mantissa is in a binary form. The value of the exponent is entered with a bias. A bias means that a constant is added to the value of the exponent.
- The bias is introduced in order to avoid using one of the bits for the sign of the exponent (since the exponent can be positive or negative).
- In binary notation, the largest number that can be written with 11 bits is 2047 (when all 11 digits are 1).
- In this case, the bias 1023 is used, which means that if, for example, the exponent is 4, then the value that is stored is $4 + 1023 = 1027$.

Storing a number in computer memory: IEEE-754 standard

- The value of the mantissa is in a binary form. The value of the exponent is entered with a bias. A bias means that a constant is added to the value of the exponent.
- The bias is introduced in order to avoid using one of the bits for the sign of the exponent (since the exponent can be positive or negative).
- In binary notation, the largest number that can be written with 11 bits is 2047 (when all 11 digits are 1).
- In this case, the bias 1023 is used, which means that if, for example, the exponent is 4, then the value that is stored is $4 + 1023 = 1027$.
- Smallest exponent that can be stored by the computer is -1023, and the largest is 1024 (which will be stored as 2047).

Storing a number in computer memory: IEEE-754 standard

- However, the smallest and largest values of the exponent plus bias are reserved for zero and infinity (Inf) or not-a-number (NaN) due to invalid mathematical operation.

Storing a number in computer memory: IEEE-754 standard

- However, the smallest and largest values of the exponent plus bias are reserved for zero and infinity (Inf) or not-a-number (NaN) due to invalid mathematical operation.
- The 11 bits for the exponent plus bias store values between -1023 and 1024.
- If the exponent plus bias and mantissa are both zero, then the number actually stored is 0.

Storing a number in computer memory: IEEE-754 standard

- However, the smallest and largest values of the exponent plus bias are reserved for zero and infinity (Inf) or not-a-number (NaN) due to invalid mathematical operation.
- The 11 bits for the exponent plus bias store values between -1023 and 1024.
- If the exponent plus bias and mantissa are both zero, then the number actually stored is 0.
- If the exponent plus bias is 2047 the number stored is Inf if the mantissa is zero, and It is NaN if the mantissa is not zero.

Storing a number in computer memory: IEEE-754 standard

- However, the smallest and largest values of the exponent plus bias are reserved for zero and infinity (Inf) or not-a-number (NaN) due to invalid mathematical operation.
- The 11 bits for the exponent plus bias store values between -1023 and 1024.
- If the exponent plus bias and mantissa are both zero, then the number actually stored is 0.
- If the exponent plus bias is 2047 the number stored is Inf if the mantissa is zero, and It is NaN if the mantissa is not zero.
- In single precision, 8 bits are allocated to the value of the exponent and the bias is 127.

Example

Example 07: How the number 22.5 can be stored in double precision according to the IEEE-754 standard.

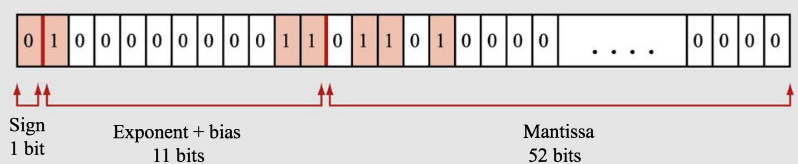
Example

Example 07: How the number 22.5 can be stored in double precision according to the IEEE-754 standard.

Solution: First, the number is normalized:

$$\frac{22.5}{2^4} = 1.40625 \times 2^4$$

In double precision, the exponent with the 2^4 bias is $4 + 1023 = 1027$, which is stored in binary form as 10000000011. The mantissa is 0.40625, which is stored in binary form as $.01101000....000$. The storage of the number is illustrated below



Limitations

- The smallest positive number that can be expressed in double precision is:

$$2^{-1022} \approx 2.2 \times 10^{-308}$$

This means that there is a (small) gap between zero and the smallest number that can be stored on the computer. Attempts to define a number in this gap causes an underflow error. (In the same way, the closest negative number to zero is -2.2×10^{-308}).

Limitations

- The smallest positive number that can be expressed in double precision is:

$$2^{-1022} \approx 2.2 \times 10^{-308}$$

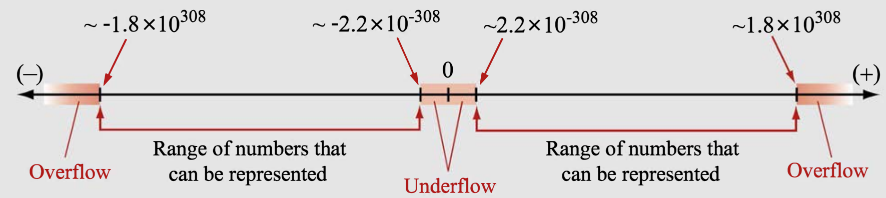
  This means that there is a (small) gap between zero and the smallest number that can be stored on the computer. Attempts to define a number in this gap causes an underflow error. (In the same way, the closest negative number to zero is -2.2×10^{-308}).

- The largest positive number that can be expressed in double precision is approximately:

$$2^{1024} \approx 1.8 \times 10^{308}$$

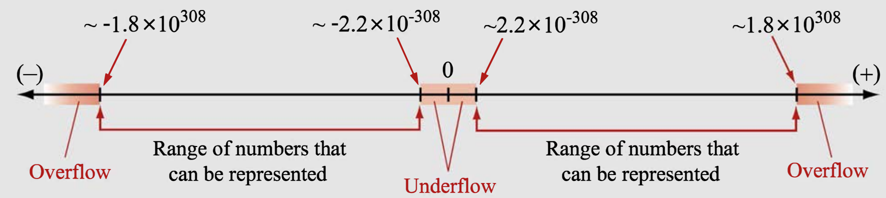
  Attempts to define a larger number causes overflow error. (The same applies to numbers smaller than -2^{1024}.)

Limitations



- Since a finite number of bits is used, not every number can be accurately written in binary form.

Limitations



- Since a finite number of bits is used, not every number can be accurately written in binary form.
- For example, the number 0.1 cannot be represented exactly in finite binary format when single precision is used. To be written in binary floating point representation, 0.1 is normalized: $0.1 = 1.6 \times 2^{-4}$. The exponent -4 (with a bias) can be stored exactly, but the mantissa 0.6 cannot be written exactly in a binary format that uses 23 bits.

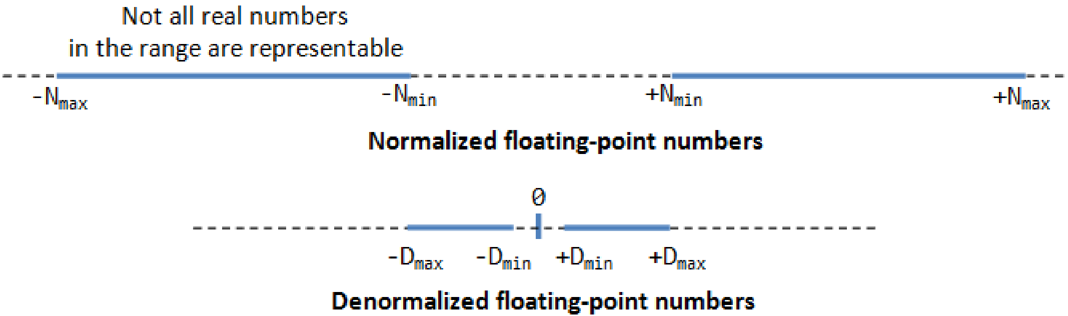
Limitations

- The interval between numbers that can be represented depends on their magnitude. In double precision, the smallest value of the mantissa that can be stored is $2^{-52} \approx 2.22 \times 10^{-16}$.
- For numbers of the order of 1, the smallest difference between two numbers that can be represented in double precision is then 2.22×10^{-16}. This value is also defined as the machine epsilon in double precision.

Limitations

- The interval between numbers that can be represented depends on their magnitude. In double precision, the smallest value of the mantissa that can be stored is $2^{-52} \approx 2.22 \times 10^{-16}$.
- For numbers of the order of 1, the smallest difference between two numbers that can be represented in double precision is then 2.22×10^{-16}. This value is also defined as the machine epsilon in double precision.
- For single precision the smallest difference between two number is 1.1920929×10^{-7}.

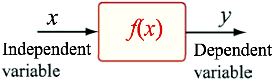
Smaller than smallest postive number



Mathematical Background

Function

- A function written as $y = f(x)$ associates a unique number y (dependent variable) with each value of x (independent variable).
- Domain: the span of values that x can have from its minimum to its maximum value.
- Range: the span of the corresponding values of y.



- The domain and range of the variables are also called intervals.
- When the interval includes the endpoints (the first and last values of the variable), then it is called a closed interval, $[a, b]$; when the endpoints are not included, the interval is called an open interval,(a, b). Where a and b are endpoints of the interval of x.
- $T = f(x, y, z)$, function can have more than one independent variable.

Limit of a function

- If a function $f(x)$ comes arbitrarily close to a single number L as x approaches a number a from either the right side or the left side, then the limit of $f(x)$ is said to approach L as x approaches a. Symbolically, the limit is expressed by:

$$\boxed{\lim_{x \to a} f(x) = f(a) = L} \tag{11}$$

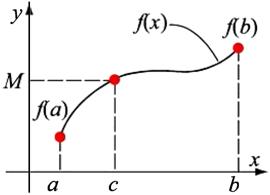
- The formal definition states that if $f(x)$ is a function defined on an open interval containing a and L is a real number, then for each number $\epsilon > 0$, there exists a number $\delta > 0$ such that if $0 < |x - a| < \delta$ then $|f(x) - L| < \epsilon$. Since δ can be chosen to be arbitrarily small, $f(x)$ can be made to approach the limit L as closely as desired.

Continuity of a function

- A function $f(x)$ is said to be continuous at $x = a$ if the following three conditions are satisfied:
  (1) $f(a)$ exists,
  (2) $\lim_{x\to a} f(x)$ exists, and
  (3) $\lim_{x\to a} f(x) = f(a)$
- A function is continuous on an open interval (a, b) if it is continuous at each point in the interval.
- A function that is continuous on the entire real axis $(-\infty, \infty)$ is said to be everywhere continuous.
- Numerically, continuity means that small variations in the independent variable give small variations in the dependent variable.

Intermediate value theorem

- The intermediate value theorem is a useful theorem about the behavior of a function in a closed interval.
- Formally, it states that if $f(x)$ is continuous on the closed interval $[a, b]$ and M is any number between $f(a)$ and $f(b)$, then there exists at least one number c in $[a, b]$ such that $f(c) = M$

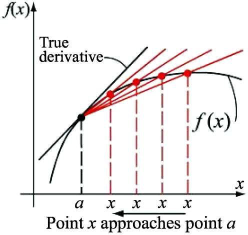


- The intermediate value theorem implies that the graph of a continuous function cannot have a vertical jump.

Derivatives of a function

- The ordinary derivative, first derivative, or, simply, derivative of a function $y = f(x)$ at a point $x = a$ in the domain of $f(x)$ is denoted by $\frac{dy}{dx}$, y', $\frac{df}{dx}$, or $f'(a)$, and is defined as:

$$\boxed{\left.\frac{dy}{dx}\right|_{x=a} = f'(a) = \lim_{x \to a} \frac{f(x) - f(a)}{x - a}}$$



- The derivative of the function $f(x)$ at the point $x = a$ is the slope of the tangent to the curve $y = f(x)$ at that point.
- A function must be continuous before it can be differentiable.
- A function that is continuous and differentiable over a certain interval is said to be smooth.

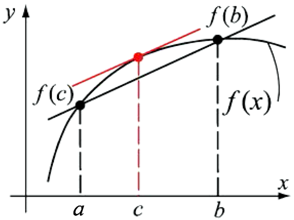
Derivatives of a function

- There are two important ways to interpret the first derivative of a function.
  - As the slope of the tangent to the curve described by $y = f(x)$ at a point which is very useful in finding the maximum or minimum of the curve $y = f(x)$ since the slope (and hence the first derivative) must be zero at those points.
  - The second interpretation of the derivative is as the rate of change of the function $y = f(x)$ with respect to x. In other words, $\frac{dy}{dx}$ represents how fast y changes as x is changed.
- Higher-order derivatives may be obtained by successive application of first order derivative.

Mean value theorem for derivatives

- Formally, it states that if $f(x)$ is a continuous function on the closed interval $[a, b]$ and differentiable on the open interval (a, b) , then there exists a number c within the interval, $c \in (a, b)$, such that:

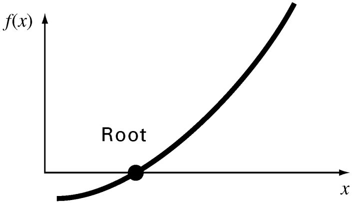
$$\boxed{f'(c) = \left.\frac{dy}{dx}\right|_{x=c} = \frac{f(b) - f(a)}{b - a}}$$



- Simply stated, the mean value theorem for derivatives states that within the interval there exists a point c such that the value of the derivative of $f(x)$ is exactly equal to the slope of the secant line joining the endpoints $(a, f(a))$, and $(b, f(b))$.
- The mean value theorem is very useful in numerical analysis when finding bounds for the order of magnitude of numerical error for different methods.

Type of Problems

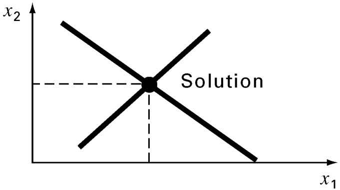
- Roots of equations:
  Solve $f(x) = 0$ for x.



- Linear algebraic equations:
  Given the a's and c's, Solve

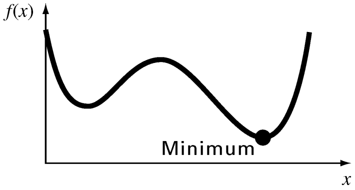
$$a_{11}x_1 + a_{12}x_2 = c_1$$
$$a_{21}x_1 + a_{22}x_2 = c_2$$

for the x's

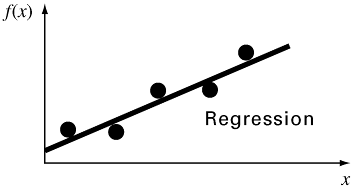


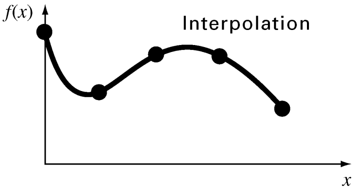
Type of Problems

- Optimization: Determine x that gives optimum $f(x)$.



- Curve fitting



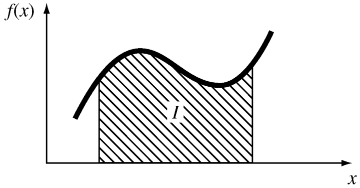


Type of Problems

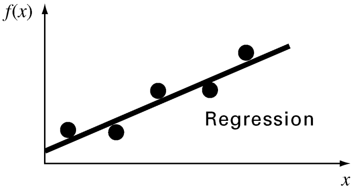
- Integration

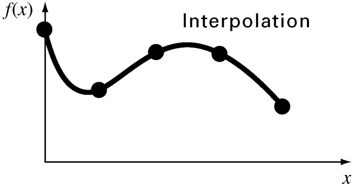
$$I = \int_a^b f(x) f x$$

find the area under the curve.



- Curve fitting





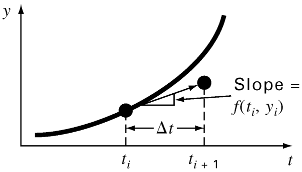
Type of Problems

- Ordinary differential equations
  Given

$$\frac{dy}{dt} \approx \frac{\Delta y}{\Delta t} = f(t, y)$$

solve for y as a funtion of t.

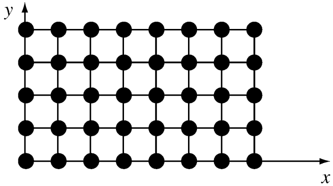
$$y_{i+1} = y_i + f(t_i, y_i)\Delta t$$



- Partial differential equations
  Given

$$\frac{\delta^2 u}{\delta x^2} + \frac{\delta^2 u}{\delta y^2} = f(x, y)$$

solve for u as a function of x and y.



References

- Numerical Methods Using MATLAB by Matthews and Fink, Pearson
- Applied Numerical Methods with MATLAB for Engineers and Scientists, Third Edition Steven C. Chapra, McGraw-Hill
- Numerical Methods for Engineers and Scientists An Introduction with Applications using MATLAB, Third Edition, Amos Gilat and Vish Subramaniam, John Wiley & Sons

Thank you!