# Training and Workshop on Python Programming and Numerical Methods

An Introduction to Numerical Linear Algebra and Differential Equations

**RHUDAINA MOHAMMAD**

Numerical Analysis & Scientific Computing Group
Institute of Mathematics, UP Diliman
*rzmohammad@up.edu.ph*

# Training and Workshop on Python Programming and Numerical Methods

An Introduction to Numerical Linear Algebra and Differential Equations

| | |
|---|---|
| Day 1 | Python Basics and Programming<br>DR. CRISANTO ABAS. |
| **Day 2** | **Introduction to Numerical Linear Algebra.** |
| Day 3 | Introduction to Numerical Differential Equations and Optimization<br>DR. ARRIANNE CRYSTAL VELASCO. |

## Day 2. Introduction to Numerical Linear Algebra

| | |
|---|---|
| 08:30 | **Gauss Elimination.** |
| 10:00 | *Break.* |
| 10:30 | **Matrix Factorization.** |
| 12:00 | *Lunch.* |
| 13:00 | **Singular Value Decomposition.** |
| 15:00 | *Break.* |
| 15:30 | **Linear Sparse Systems and Iterative Methods.** |
| 17:00 | *End of Day 2.* |

# Our Goal Today:

Find $x$.

# Find $x$.

$$2x - 3 = 0$$

$$\boxed{x = \frac{3}{2}}$$

# Find $x = (x_1, x_2)$.

$$\begin{aligned} 2x_1 &- 3x_2 &= 5 \\ 3x_1 &+ x_2 &= 2 \end{aligned}$$

$$\boxed{x = (1, -1)}$$

## Find $x = (x_1, x_2, x_3)$.

$$\begin{aligned} 4x_1 - x_2 + x_3 &= 8 \\ 2x_1 + 5x_2 + 2x_3 &= 3 \\ x_1 + 2x_2 + 4x_3 &= 11 \end{aligned}$$

$$\boxed{x = (1, -1, 3)}$$

## Find $x = (x_1, x_2, x_3, x_4)$.

$$\begin{array}{rcrcrcrcr}
x_1 & - & x_2 & + & 2x_3 & - & x_4 & = & -8 \\
x_1 & + & x_2 & + & x_3 & & & = & -2 \\
2x_1 & - & 2x_2 & + & 3x_3 & - & 3x_4 & = & -20 \\
x_1 & - & x_2 & + & 4x_3 & + & 3x_4 & = & 4
\end{array}$$

# Linear Systems

Solve a system of $n$ linear equations with $m$ unknowns:

$$
\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \cdots + a_{1m}x_1 &= y_1 \\
a_{21}x_1 + a_{22}x_2 + \cdots + a_{2m}x_2 &= y_2 \\
&\vdots \\
a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nm}x_n &= y_n
\end{aligned}
$$

# Linear Systems

In matrix notation, find $x$ such that

$$Ax = y$$

where

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix} \qquad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} \qquad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

# Linear Systems

In matrix notation, find $x$ such that

$$Ax = y$$

where

$$A = \underbrace{\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix}}_{n\times m} \qquad x = \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}}_{m\times 1} \qquad y = \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}}_{n\times 1}$$

Note: We also write $x = (x_1, x_2, \ldots, x_m)$.

# Our Goal Today:

Find $x$ such that $Ax = y$.

Why?

# Why?

Engineering systems, at steady state, can be modeled with linear equations.

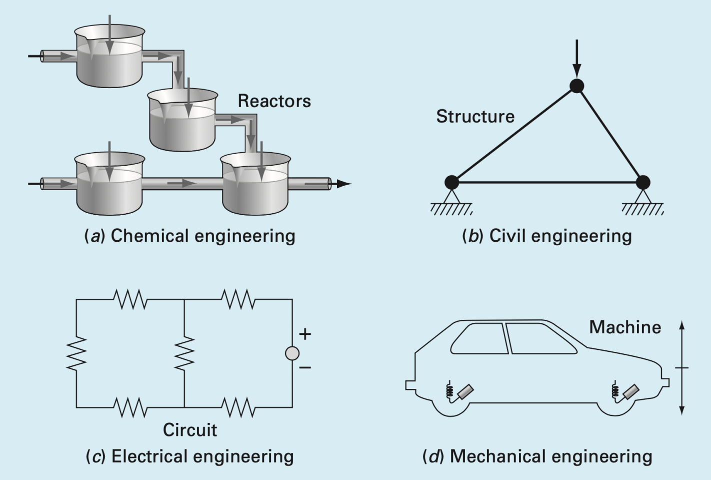



Image Source: Chapra (2008) Applied Numerical Methods, p. 223

# Why?

Physical systems can be modeled with differential equations.

Heat flows through the rod as well as between the rod and the surrounding air.

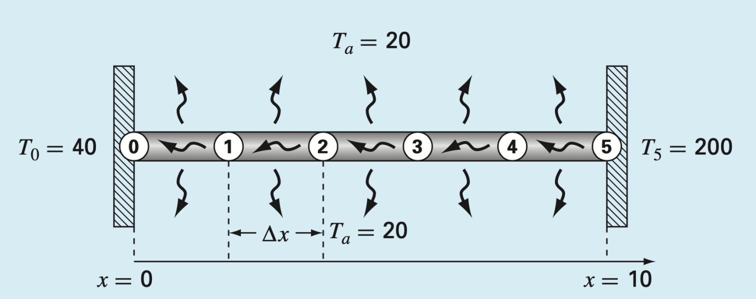



Image Source: Chapra (2008) Applied Numerical Methods, p. 248

At steady state, we have $\dfrac{d^2u}{dx^2} + h(u_{\text{air}} - u) = 0$.

Approximating the solution requires solving a linear system.

# Why?

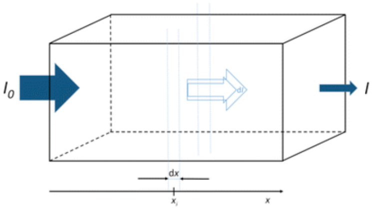
Some DE models in chemistry.

Chemical reaction kinetics

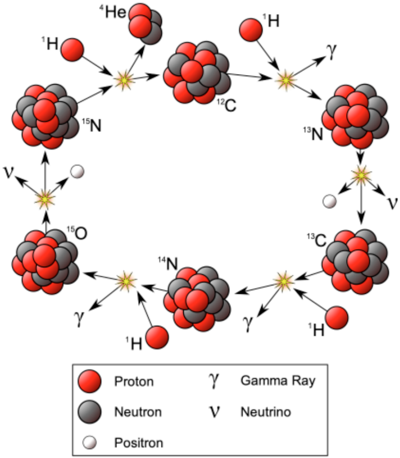
$$A + B \rightarrow C + B$$

$$-\frac{d[A]}{dt} = k\,[A][B]$$

Bougert-Lambert-Beer law



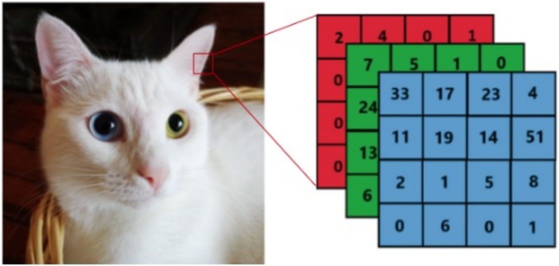

Radioactive decay



# Why?

Data science deals with matrices : table of values, images, videos, audio signals.

| | feature 1 | feature 2 | $\cdots$ | feature $m$ |
|---|---|---|---|---|
| sample 1 | $*$ | $*$ | | $*$ |
| sample 2 | $*$ | $*$ | | $*$ |
| $\vdots$ | $\vdots$ | $\vdots$ | | $\vdots$ |
| sample $n$ | $*$ | $*$ | | $*$ |



## Our Goal Today:

Find $x$ such that $Ax = y$.

Is there a solution? Is it unique?

# Recall: Linear Algebra

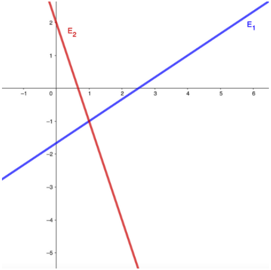
The following are equivalent:

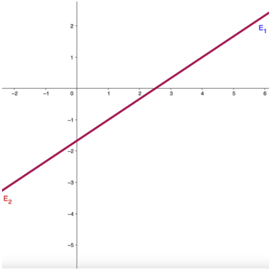
1. For any $y \in \mathbb{R}^n$ , linear system

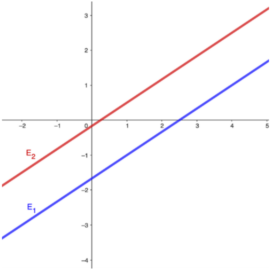
   $$Ax = y$$

   has a unique solution
2. $A$ is nonsingular.
3. Inverse matrix $A^{-1}$ exists
4. Homogeneous system $Ax = 0$ has only trivial solution.
5. $\det A \neq 0$
6. Rows (columns) of $A$ are linearly independent.

# Recall: Linear Algebra




$$\begin{cases} 2x_1 - 3x_2 &= 5 \\ 3x_1 + x_2 &= 2 \end{cases}$$

$$\begin{cases} 2x_1 - 3x_2 &= 5 \\ 4x_1 - 6x_2 &= 10 \end{cases}$$

$$\begin{cases} 2x_1 - 3x_2 &= 5 \\ 4x_1 - 6x_2 &= 1 \end{cases}$$

## Our Goal Today:

Find $x$ such that $Ax = y$.

How?

# GAUSS ELIMINATION

# Recall: Linear Algebra

In solving a system of equations, we performed the following (elementary) row operations:

1. Replace row $j$ by the sum of itself and a multiple of row $i$:

$$mR_i + R_j \to R_j$$

2. Scale a row by multiplying all entries by a nonzero number
3. Swap two rows

- Two matrices are row equivalent if one can be transformed to the other by a sequence of row operations.
- Two linear systems are equivalent if every solution of one is a solution of the other.

**Two linear systems are equivalent if their augmented matrices are row equivalent.**

# Gauss Elimination ($m = n$)

- Construct the augmented matrix $\hat{A} := [A \mid y]$
- Elimination. Reduce $\hat{A}$ to an upper triangular system

$$\hat{A} = \left[\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & a_{1,n+1} \\ a_{21} & a_{22} & \cdots & a_{2n} & a_{2,n+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & a_{n,n+1} \end{array}\right] \longrightarrow \left[\begin{array}{cccc|c} * & * & \cdots & * & * \\ & * & \cdots & * & * \\ & & \ddots & \vdots & \vdots \\ & & & * & * \end{array}\right]$$

using the following row operation

$$-\frac{a_{ji}}{a_{ii}} R_i + R_j \to R_j, \qquad j = i+1, i+2, \ldots, n$$

provided $a_{ii} \neq 0$.

# Gauss Elimination

- Backward substitution

$$x_n = \frac{a_{n,n+1}}{a_{nn}}$$

$$x_i = \frac{a_{i,n+1} - \displaystyle\sum_{j=i+1}^{n} a_{ij}x_j}{a_{ii}}, \qquad i = n-1, n-2, \ldots, 2, 1$$

# Computational Cost

*Q.* How many flops (floating point operations) are needed?

- **Elimination step.**

  divisions in $m_{ji} = \frac{a_{ji}}{a_{ii}}$: $n - i$

  multiplications in $m_{ji}R_i + R_j$: $(n-i)(n-i+1)$

  additions in $m_{ji}R_i + R_j$: $(n-i)(n-i+1)$

  Summing the flops for each $i$:

$$\sum_{i=1}^{n-1} (n-i)(2n-2i+3),$$

  which is of order $O(n^3)$.

# Computational Cost

- Backward substitution.

divisions in $x_n$: $1$

for each $i = 1, \ldots, n-1$:

multiplications in summation: $n - i$

division: $1$

additions in summation: $n - i - 1$

subtraction $1$

Summing the flops:

$$1 + \sum_{i=1}^{n-1} \underbrace{(n-i)+1}_{\text{multiplications/divisions}} + \underbrace{(n-i-1)+1}_{\text{additions/subtractions}}$$

which is of order $O(n^2)$.

# Gauss Elimination

- If $A$ is singular (non-invertible) and has rank $r$, the elimination process will terminate after $r$ steps.

  In this case, the linear system is solvable if and only if

  $$y_{r+1} = \cdots = y_n = 0.$$

  The solution can be found by arbitrarily choosing

  $$x_{r+1}, \ldots, x_n$$

# Gauss Elimination

- To control the influence of rounding errors, keep $\frac{a_{ji}}{a_{ii}}$ small, i.e., require pivot element $a_{ii}$ to be large.
  - Partial Pivoting. Determine the smallest $p \geq k$ such that

    $$|a_{pk}| = \max_{k \leq i \leq n} |a_{ik}|.$$

    Then, interchange rows $p$ and $k$.
  - Complete Pivoting. Both rows and columns are reordered such that $a_{ii}$ has the maximal absolute value in the $(n-i+1) \times (n-i+1)$ matrix remaining for the $i$th elimination step.

# Hands-on Activity

Gauss-Jordan Method circumvents the backward substitution method in Gauss elimination as follows.

- Elimination step. Reduce $\hat{A} := [A \,|\, y]$ to a diagonal system

$$\hat{A} = \left[\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & a_{1,n+1} \\ a_{21} & a_{22} & \cdots & a_{2n} & a_{2,n+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & a_{n,n+1} \end{array}\right] \longrightarrow \left[\begin{array}{cccc|c} * & & & & * \\ & * & & & * \\ & & \ddots & & \vdots \\ & & & * & * \end{array}\right]$$

- Evaluation

$$x_i = \frac{a_{i,n+1}}{a_{ii}}, \quad i = 1, 2, \ldots, n$$

Write a Python code for Gauss-Jordan method with partial pivoting. Using your code, solve the given linear system.

– BREAK –

10:00 – 10:30