Systems of Linear Equations and Gaussian Elimination
Last time: Tomography

What is a projection? Sum of values along a line.

If you like tomography, you’ll love EE123, EE145B, EECS261, EE225E!
Or research with profs: Miki Lustig, Chunlei Liu

What is a projection? Sum of values along a line.
Vectors are arrays of numbers
represents a SINGLE POINT in N-dimensional space

\[ \tilde{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \]

column vector

\[ x = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_N \end{bmatrix} \]

row vector

What are the dimensions of this vector?

\[ \tilde{x} \in \mathbb{R}^3 \]

3-dimensional vector

\[ \tilde{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \]
A matrix is a rectangular array of numbers

\[
A = \begin{bmatrix}
A_{11} & A_{12} & \cdots & A_{1n} \\
A_{21} & A_{22} & \cdots & A_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
A_{m1} & A_{m2} & \cdots & A_{mn}
\end{bmatrix}
\]

This is element (component) 2n of the matrix

What are the dimensions of \( A \)?

**m rows** and **n columns** means it is a **m x n** matrix
Some special types of matrices

**zero matrix**

\[
\bar{0} = \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{bmatrix}
\]

**diagonal matrix**

\[
A = \begin{bmatrix}
a_{11} & 0 & 0 & \cdots & 0 \\
0 & a_{22} & 0 & \cdots & 0 \\
0 & 0 & a_{33} & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & a_{nn}
\end{bmatrix}
\]

**identity matrix**

\[
I = \begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

**upper triangular matrix**

\[
A = \begin{bmatrix}
a_{11} & a_{12} & a_{13} & a_{14} \\
0 & a_{22} & a_{23} & a_{24} \\
0 & 0 & a_{33} & a_{34} \\
0 & 0 & 0 & a_{44}
\end{bmatrix}
\]
lately, i've just felt like nothing inside...

i can definitely identify with how you are feeling...

Association of Psychiatric Medicine

1 0 0
0 1 0
0 0 1
Ways of representing linear systems of equations

Can also be represented as:

\[ a_1 x_1 + a_2 x_2 = b_1 \]
\[ a_3 x_1 + a_4 x_4 = b_2 \]
\[ a_5 x_1 + a_5 x_3 = b_3 \]
\[ a_2 x_2 + a_4 x_4 = b_4 \]

Or:

\[
\begin{bmatrix}
 a & 0 & 0 & 0 \\
 0 & a & 0 & 0 \\
 0 & 0 & a & 0 \\
 0 & 0 & 0 & a \\
\end{bmatrix}
\begin{bmatrix}
 x_1 \\
 x_2 \\
 x_3 \\
 x_4 \\
\end{bmatrix}
= 
\begin{bmatrix}
 b_1 \\
 b_2 \\
 b_3 \\
 b_4 \\
\end{bmatrix}
\]

Or:

\[ A x = b \]
Today: Solving a linear system of equations

First, write in simple form:

\[
\begin{bmatrix}
1 & 4 & | & 6 \\
2 & -1 & | & 3 \\
\end{bmatrix}
\]

Don’t forget to put x in one column and y in another

Now solve it. How?

Start plugging equations into each other.... See what happens?

e.g.

1) Solve ① for x and plug into ②
2) 4 \times ② + ①
GOAL: to develop a **systematic** way of solving systems of equations with clear rules that *can be done by a computer*

(then I can be even lazier)
**Gaussian Elimination** for solving a linear system of equations

- Specifies the order in which you combine equations (rows) to "eliminate" (make zero) certain elements of the matrix

- Goal is to transform your system of equations into **upper triangular**

\[
A = \begin{bmatrix}
  a_{11} & a_{12} & a_{13} & a_{14} \\
  0 & a_{22} & a_{23} & a_{24} \\
  0 & 0 & a_{33} & a_{34} \\
  0 & 0 & 0 & a_{44}
\end{bmatrix}
\]

Diagonal elements are called **pivots**
The Gauss elim. way to solve is just a specific

\[ \begin{align*}
\text{(1)} & \quad x + 4y = 6 \\
\text{(2)} & \quad 2x - y = 3
\end{align*} \]

In simple form:

\[
\begin{bmatrix}
1 & 4 & 6 \\
2 & -1 & 3
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 & 4 & 6 \\
0 & -9 & -9
\end{bmatrix}
\]

Then plug it in, say to (1):

\[ x + 4(1) = 6 \]

\[ x = 2 \]

We 'eliminated' \( x \) to make this upper triangular.

\[
\begin{align*}
-9y &= -9 \\
y &= 1
\end{align*}
\]

Could also solve graphically:

(find \( x, y \) intercept)

Solution is (2, 1)
To be systematic, we should instead make matrix manipulation so that 3rd row pivot is 1:

\[
\begin{bmatrix}
1 & 3 & 2 & | & 5 \\
0 & 2 & 1 & | & 1 \\
0 & 0 & 1 & | & 3
\end{bmatrix}
\]

**Elimination Part**

Now we read off \( z = 3 \)

For the "plug in" part, now we need to back-substitute upwards from bottom:

\[
\begin{align*}
2y + 3 &= 1 \\
y &= -1
\end{align*}
\]

Then plug \( z = 3 \) and \( y = -1 \) into Row 1:

\[
\begin{align*}
x + 3y + 2z &= 5 \\
x + 3(-1) + 2(3) &= 5 \\
x &= 2
\end{align*}
\]

**To translate this into matrix manipulations**:

\[
\begin{align*}
\text{(Row 2 - Row 3)} & \quad \begin{bmatrix} 1 & 3 & 2 & | & 5 \\
0 & 2 & 1 & | & 1 \\
0 & 0 & 1 & | & 3 \end{bmatrix} \\
\text{divide Row 2 by 2} & \quad \begin{bmatrix} 1 & 3/2 & 1 & | & 5 \\
0 & 1 & 1/2 & | & 1/2 \\
0 & 0 & 1 & | & 3 \end{bmatrix} \\
\text{Finally, (Row 1 - 3Row 2 + 2Row 3)} & \quad \begin{bmatrix} 1 & 0 & -1 & | & 2 \\
0 & 1 & -1 & | & -1 \\
0 & 0 & 1 & | & 3 \end{bmatrix}
\end{align*}
\]

Read off \( x = 2 \), \( y = -1 \), \( z = 3 \).
What is allowed in Gaussian elimination?

• Linear combinations of equations (adding scalar multiples of rows to other rows)

• Multiply a row by a scalar

• Swap rows
Goals of Gaussian Elimination algorithm

• Equation with $i^{th}$ variable in the $i^{th}$ row

• Coefficient of the $i^{th}$ variable in the $i^{th}$ row becomes 1

• For rows $j=i+1$ and higher, subtract row $i$ times the entry in $(j,i)$ to cancel variable $i$
Gaussian elimination was part of the work of human computers

What might be the variables/measurements in calculating rocket trajectories?

Position, direction of motion, tilt, power/thrust, weight...
Will it always work?

In both cases, the number in the pivot position* being zero was a red flag! (*technically, it’s not called a pivot if zero)
Possible situations

- Unique solution
- Infinitely many solutions (underdetermined)
- No solution (inconsistent)

Is it possible to have exactly 2 solutions? No. consider graphically: two lines cannot intersect in exactly two places
Cats vs. Dogs

These measurements are different linear combinations of two images.

Can you guess what the measurements are?

Top: 0.6 (dog) + 0.4 (cat)
Bottom: 0.6 (cat) + 0.4 (dog)

Can I solve for both images from just these two linearly combined images? Just one? None? How many images do I need minimum?

Two images is enough if they’re linearly independent at each pixel!
Cats vs. Dogs

What are the ideal measurements? Depends. Maybe direct measurements of cat and dog...
Cats vs. Dogs: Direct measurements

Very easy to solve!
CATS vs. DOGS

How would you set up the linear system of equations?

\[
0.6 \begin{bmatrix} \text{dog} \\ \text{car} \end{bmatrix} + 0.4 \begin{bmatrix} \text{dog} \\ \text{car} \end{bmatrix} = \begin{bmatrix} \text{cat} \end{bmatrix}
\]

What are unknowns? **ALL pixels of DOG image**

**AND** **ALL pixels of CAT image**

i.e. \((300 \text{ pixels} \times 300 \text{ pixels}) \times 2\)

\[= 180,000 \text{ pixels} \]

Need to put them all into one vector:

\[
\begin{bmatrix} 90,000 \\ \text{Dog} \end{bmatrix} \quad \begin{bmatrix} 90,000 \\ \text{CAT} \end{bmatrix}
\]

e.g. \(\text{Rather scan (like in 2D tomography)}\)

Then just stick together:

\[
\begin{bmatrix} \text{DOG} \\ \text{CAT} \end{bmatrix}
\]

\(180,000 \times 1 \text{ vector} \)
My Research uses linear algebra

Computational Imaging: joint design of hardware and software

Image System Design

\[ \text{Algorithm} \]

\begin{align*}
\text{find } x \\
\text{such that } y = Ax
\end{align*}
Computational imaging pipeline
DiffuserCam: tape a diffuser onto a sensor
Lenses map a point to a point
Diffuser maps points to many points (linear combination!)

Point Spread Function (PSF)
Traditional cameras take direct measurements
Computational cameras can multiplex

\[ \text{measurement} = \text{object} \]
raw sensor data

recovered scene
raw sensor data

recovered scene
El cheapo version – ScotchTapeCam!

Raspberry Pi + sensor + Scotch tape = Scotch tape caustics

Reconstruction

https://waller-lab.github.io/DiffuserCam/