

# Matrix Conditioning

$$\kappa(A) = \|A\|_2 \underbrace{\|A^{-1}\|_2}$$

conditioning =  $\max_{\text{true}} \max_{\text{err. input}} \left( \frac{\text{rel error out}}{\text{rel error in}} \right)$

# Conditioning

Now, let's study condition number of solving a linear system

$$Ax = b.$$

$$x = \hat{x} + \Delta x$$

out.

$$b = \hat{b} + \Delta b$$

inp.

$$\max_{x, \Delta x} \left( \frac{\|Ax\|_2 / \|x\|_2}{\|Ab\|_2 / \|b\|_2} \right) \leq \|A\|_2 \|A^{-1}\|_2$$

$$A \hat{x} = \hat{b}$$

distance  $\hat{x}$  and  $x$

**Demo:** Condition number visualized

**Demo:** Conditioning of  $2 \times 2$  Matrices

## More Properties of the Condition Number

What is  $\text{cond}(A^{-1})$ ?

$$\kappa(A) = \|A\| \|A^{-1}\| = \|A^{-1}\| \|A\| = \kappa(A^{-1})$$

What is the condition number of applying the matrix-vector multiplication  $Ax = b$ ? (i.e. now  $x$  is the input and  $b$  is the output)

Find  $b$ ,  $A^{-1}b = x$

condition number is  $\kappa(A)$

## Matrices with Great Conditioning (Part 1)

Give an example of a matrix that is *very* well-conditioned.  
(I.e. has a condition-number that's *good* for computation.)  
What is the best possible condition number of a matrix?

$$\kappa(A \cdot A^{-1}) = \kappa(I) = 1$$

$$\hookrightarrow \|I\| \cdot \|I\| = 1$$

||

$$\|A A^{-1}\| \cdot \|A^{-1} A\| \geq 1$$

$$1 = \|A A^{-1}\| \leq \|A\| \cdot \|A^{-1}\|$$

$$1 = \|A^{-1} A\| \leq \|A^{-1}\| \cdot \|A\|$$

## Matrices with Great Conditioning (Part 2)

What is the 2-norm condition number of an orthogonal matrix  $A$ ?

$$Q^T Q = I \Rightarrow Q^{-1} = Q^T$$

$$\|Q\|_2 = \max_{\|x\|_2=1} \sqrt{x^T \underbrace{Q^T Q}_I x} = \sqrt{x^T x} = \|x\|_2 = 1$$

$$\kappa_2(Q) = \|Q\|_2 \|Q^T\|_2 = 1 \cdot 1 = 1$$

↑                    ↑  
orthogonal       orthogonal

## In-class activity: Matrix Conditioning



$$\text{have} = \text{true} + \text{err}$$
$$\frac{\max \text{ true err}}{\max \text{ true err}} = \frac{\text{amplification of err}}{\text{amplification of true}}$$



# Outline

Python, Numpy, and Matplotlib  
Making Models with Polynomials  
Making Models with Monte Carlo  
Error, Accuracy and Convergence  
Floating Point  
Modeling the World with Arrays  
    The World in a Vector  
    What can Matrices Do?  
    Graphs  
    Sparsity  
Norms and Errors  
**The 'Undo' Button for Linear Operations: LU**  
Repeating Linear Operations:  
Eigenvalues and Steady States  
Eigenvalues: Applications

Approximate Undo: SVD and Least Squares  
SVD: Applications  
    Solving Funny-Shaped Linear Systems  
    Data Fitting  
    Norms and Condition Numbers  
    Low-Rank Approximation  
Interpolation  
Iteration and Convergence  
Solving One Equation  
Solving Many Equations  
Finding the Best: Optimization in 1D  
Optimization in  $n$  Dimensions



# Solving Systems of Equations

Want methods/algorithms to solve linear systems. Starting small, a kind of system that's easy to solve has a ... matrix.

$$2x_1 + 3x_2 = 7$$

$$4x_2 = 1$$

$$x_2 = \frac{1}{4}$$

$$\begin{pmatrix} 2 & 3 \\ 0 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 7 \\ 1 \end{pmatrix}$$

# Triangular Matrices

Solve

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ & a_{22} & a_{23} & a_{24} \\ & & a_{33} & a_{34} \\ & & & a_{44} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix}.$$

$$x_4 = b_4 / a_{44}$$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ & a_{22} & a_{23} \\ & & a_{33} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} b_1 - a_{14}x_4 \\ b_2 - a_{24}x_4 \\ b_3 - a_{34}x_4 \end{pmatrix}$$

**Demo:** Back-substitution

**In-class activity:** Forward-substitution

## General Matrices

What about non-triangular matrices?

LU decomposition = Gaussian elimination

Given  $n \times n$  matrix  $A$ , obtain lower triangular matrix  $L$  and upper triangular matrix  $U$  such that  $A = LU$ .

Is there some redundancy in this representation?

$$\text{rank}(L) = \sum_{i=1}^n i \geq \frac{n(n+1)}{2}$$

$$\text{rank}(L) \neq \text{rank}(U) = n(n+1)$$

$L$  has unit diagonal  $L_{ii} = 1$

## Using LU Decomposition to Solve Linear Systems

Given  $A = LU$ , how do we solve  $Ax = b$ ?

$$LUx = b$$

$\underbrace{\quad}_{y}$

$Ly = b \iff$  solve for  $y$  - fwd. subs.

$Ux = y \iff$  solve for  $x$  - fwd. subs.

## 2-by-2 LU Factorization (Gaussian Elimination)

Lets consider an example for  $n = 2$ .

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \approx \begin{pmatrix} 1 & \\ l_{21} & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} \\ & u_{22} \end{pmatrix}$$

$$a_{11} = 1 \cdot u_{11} \quad \text{so } u_{11} = a_{11}$$

$$a_{12} = 1 \cdot u_{12} \quad \text{so } u_{12} = a_{12}$$

first row of  $U$  = first row of  $A$

$$l_{21} \cdot u_{11} = a_{21} \Rightarrow l_{21} = a_{21}/u_{11}$$

$$a_{22} = [k_{21} \ 1] \begin{bmatrix} u_{12} \\ u_{22} \end{bmatrix}$$

$$= k_{21} u_{12} + u_{22}$$

$$u_{22} = a_{22} - \underbrace{k_{21} u_{12}}$$

Schur complement

## General LU Factorization (Gaussian Elimination)

$$A = \begin{bmatrix} a_{11} & \mathbf{a}_{12} \\ \mathbf{a}_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \mathbf{l}_{21} & L_{22} \end{bmatrix} \begin{bmatrix} u_{11} & \mathbf{u}_{12} \\ 0 & U_{22} \end{bmatrix}$$

$$[u_{11} \quad u_{12}] = [a_{11} \quad a_{12}]$$

$$l_{21} \cdot u_{11} = a_{21}$$

$$l_{21} = a_{21} / u_{11}$$

$$\bar{A}_{22} = A_{22} - l_{21} \cdot u_{12} \quad \bar{A} = A - \bar{L} \bar{U}$$

$$[L_{22}, u_{12}] = \text{LU-alg}(\bar{A}_{22})$$