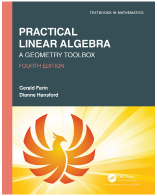
Practical Linear Algebra: A GEOMETRY TOOLBOX

Fourth Edition

Chapter 12: Gauss for Linear Systems

Gerald Farin & Dianne Hansford

A K Peters/CRC Press
www.farinhansford.com/books/pla

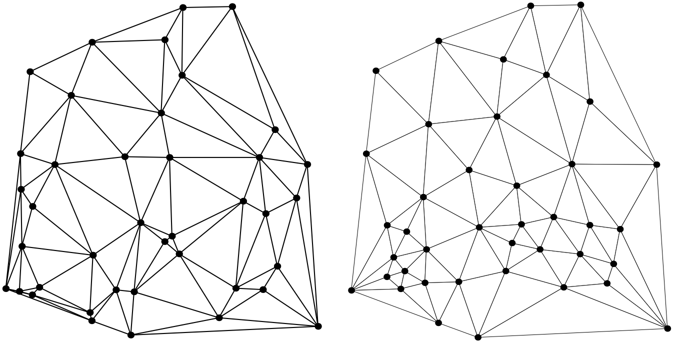


Outline

1. Introduction to Gauss for Linear Systems
2. The Problem
3. The Solution via Gauss Elimination
4. Homogeneous Linear Systems
5. Inverse Matrices
6. LU Decomposition
7. Determinants
8. Least Squares
9. Application: Fitting Data to a Femoral Head
10. WYSK

Introduction to Gauss for Linear Systems

Linear systems arise in virtually every area of science and engineering
Some as big as 1,000,000 equations in as many unknowns



Triangulation smoothing application
Left: "rough" triangulation Right: smoother triangulation

The Problem

Linear system: a set of linear equations

$$\begin{aligned} 3u_1 - 2u_2 - 10u_3 + u_4 &= 0 \\ u_1 - u_3 &= 4 \\ u_1 + u_2 - 2u_3 + 3u_4 &= 1 \\ u_2 + 2u_4 &= -4 \end{aligned}$$

Unknowns: $u_1, \ldots, u_4$
Number of equations = number of unknowns
4×4 linear system in matrix form:

$$\begin{bmatrix} 3 & -2 & -10 & 1 \\ 1 & 0 & -1 & 0 \\ 1 & 1 & -2 & 3 \\ 0 & 1 & 0 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 4 \\ 1 \\ -4 \end{bmatrix}$$

The Problem

General $n \times n$ linear system:

$$
\begin{aligned}
a_{1,1}u_1 + a_{1,2}u_2 + \ldots + a_{1,n}u_n &= b_1 \\
a_{2,1}u_1 + a_{2,2}u_2 + \ldots + a_{2,n}u_n &= b_2 \\
&\vdots \\
a_{n,1}u_1 + a_{n,2}u_2 + \ldots + a_{n,n}u_n &= b_n
\end{aligned}
$$

Matrix form:

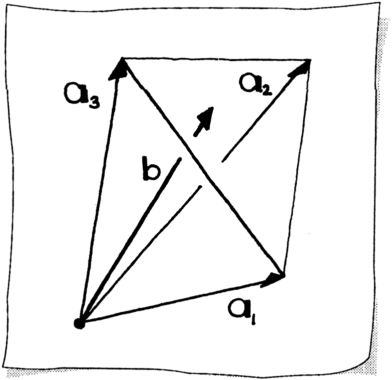
$$
\begin{bmatrix} a_{1,1} & a_{1,2} & \ldots & a_{1,n} \\ a_{2,1} & a_{2,2} & \ldots & a_{2,n} \\ & & \vdots & \\ a_{n,1} & a_{n,2} & \ldots & a_{n,n} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}
$$

$$
\begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \ldots & \mathbf{a}_n \end{bmatrix} \mathbf{u} = \mathbf{b} \qquad \Rightarrow \qquad A\mathbf{u} = \mathbf{b}
$$

A is called the coefficient matrix

The Problem

Underlying principles with a geometric interpretation

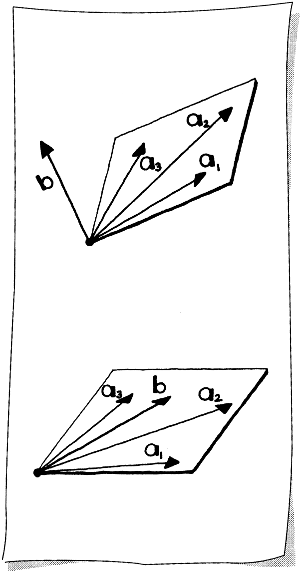


$$\begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3 \end{bmatrix} \mathbf{u} = \mathbf{b}$$

Write $\mathbf{b}$ as a linear combination of $\mathbf{a}_i$

If $\mathbf{a}_i$ truly 3D (form a tetrahedron)
$\Rightarrow$ unique solution

The Problem



$$\begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3 \end{bmatrix} \mathbf{u} = \mathbf{b}$$

If $\mathbf{a}_i$ all lie in a plane
then no unique solution

Top: no solution
Bottom: non-unique solution

The Problem

In general:

If the $\mathbf{a}_i$ have a *non-zero* n-dimensional volume
$\Rightarrow$ linear system is *uniquely solvable*

If $\mathbf{a}_i$ span a k-dimensional *subspace* ($k < n$)
$\Rightarrow$ non-unique solutions only exist if $\mathbf{b}$ is itself in that subspace

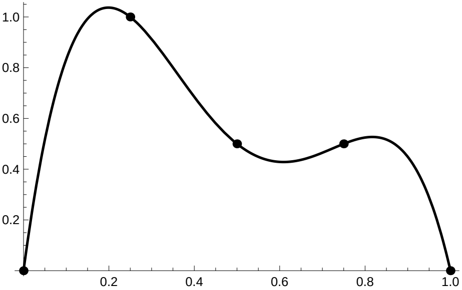
A linear system is called consistent if *at least* one solution exits

The Problem

Example: Polynomial Interpolation

Given: observations $p(t_i) = 0, 1, 0.5, 0.5, 0$ at $t_i = 0, 0.25, 0.5, 0.75, 1$ seconds

Find: a polynomial $p(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3 + c_4 t^4$
that interpolates data $\Rightarrow$ estimate values between observations



$$\begin{bmatrix} 1 & t_0 & t_0^2 & t_0^3 & t_0^4 \\ 1 & t_1 & t_1^2 & t_1^3 & t_1^4 \\ & & \vdots & & \\ 1 & t_4 & t_4^2 & t_4^3 & t_4^4 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_4 \end{bmatrix} = \begin{bmatrix} p(t_0) \\ p(t_1) \\ \vdots \\ p(t_4) \end{bmatrix}$$

The Solution via Gauss Elimination

Gauss elimination = forward elimination + back substitution

Review a 2×2 example: $A\mathbf{u} = \mathbf{b}$ $\begin{bmatrix} 2 & 4 \\ 1 & 6 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \end{bmatrix}$

Forward elimination transform system to *upper triangular* with a shear

$$S_1 A\mathbf{u} = S_1\mathbf{b} \qquad S_1 = \begin{bmatrix} 1 & 0 \\ -1/2 & 1 \end{bmatrix} \qquad \Rightarrow \qquad \begin{bmatrix} 2 & 4 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \end{bmatrix}$$

Corresponds to *elementary row operations*

$$\mathrm{row}_1 \leftarrow \mathrm{row}_1 \quad \text{and} \quad \mathrm{row}_2 \leftarrow \mathrm{row}_2 - \frac{1}{2}\mathrm{row}_1$$

The Solution via Gauss Elimination

Apply *back substitution* to upper triangular system

$$\begin{bmatrix} 2 & 4 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \end{bmatrix}$$

$$u_2 = \frac{1}{4} \times 2 = \frac{1}{2}$$
$$u_1 = \frac{1}{2}(4 - 4u_2) = 1$$

Can interpret this step as a scaling:

$$S_2 S_1 A\mathbf{u} = S_2 S_1 \mathbf{b} \qquad S_2 = \begin{bmatrix} 1/2 & 0 \\ 0 & 1/4 \end{bmatrix} \quad \Rightarrow \quad \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 1/2 \end{bmatrix}$$

The Solution via Gauss Elimination

Pivoting revisited:

$$\begin{bmatrix} 1 & 6 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \end{bmatrix} \quad \Rightarrow \quad \begin{bmatrix} 2 & 4 \\ 1 & 6 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \end{bmatrix}$$

Equations reordered so *pivot element* $a_{1,1}$ largest in first column
Row exchange can be represented as a *permutation matrix*

$$P_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \qquad P_1 A\mathbf{u} = P_1 \mathbf{b}$$

Then – Gauss elimination as before:

$$S_2 S_1 P_1 A\mathbf{u} = S_2 S_1 P_1 \mathbf{b}$$

The Solution via Gauss Elimination

Example:

$$\begin{bmatrix} 2 & -2 & 0 \\ 4 & 0 & -2 \\ 4 & 2 & -4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 4 \\ -2 \\ 0 \end{bmatrix}$$

$$P_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \Rightarrow \quad \begin{bmatrix} 4 & 0 & -2 \\ 2 & -2 & 0 \\ 4 & 2 & -4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} -2 \\ 4 \\ 0 \end{bmatrix}$$

Zero entries in the first column

$$\mathrm{row}_2 \leftarrow \mathrm{row}_2 - \frac{1}{2}\mathrm{row}_1 \qquad \mathrm{row}_3 \leftarrow \mathrm{row}_3 - \mathrm{row}_1$$

$$\text{shear} \quad G_1 = \begin{bmatrix} 1 & 0 & 0 \\ -1/2 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix} \quad \Rightarrow \quad \begin{bmatrix} 4 & 0 & -2 \\ 0 & -2 & 1 \\ 0 & 2 & -2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} -2 \\ 5 \\ 2 \end{bmatrix}$$

G_1 called a Gauss matrix

The Solution via Gauss Elimination

Example continued:

No pivoting necessary: $P_2 = I$
Zero last element in second column:

$$\mathrm{row}_3 \leftarrow \mathrm{row}_3 + \mathrm{row}_2$$

$$G_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \qquad \Rightarrow \qquad \begin{bmatrix} 4 & 0 & -2 \\ 0 & -2 & 1 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} -2 \\ 5 \\ 7 \end{bmatrix}$$

The Solution via Gauss Elimination

Example continued:
Matrix in upper triangular form — ready for back substitution:

$$u_3 = \frac{1}{-1}(7) \qquad u_2 = \frac{1}{-2}(5 - u_3) \qquad u_1 = \frac{1}{4}(-2 + 2u_3)$$

(Implicitly incorporates a scaling matrix)
Solution

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} -4 \\ -6 \\ -7 \end{bmatrix}$$

Original equations:

$$\begin{bmatrix} 2 & -2 & 0 \\ 4 & 0 & -2 \\ 4 & 2 & -4 \end{bmatrix} \begin{bmatrix} -4 \\ -6 \\ -7 \end{bmatrix} = \begin{bmatrix} 4 \\ -2 \\ 0 \end{bmatrix}$$

The Solution via Gauss Elimination

Summary:

Gauss elimination $=$
forward elimination (pivoting and shears)
$+$ back substitution (scaling)

Elementary row operations of Gauss elimination:

- Pivoting results in the exchange of two rows
- Shears result in adding a multiple of one row to another
- Scaling results in multiplying a row by a scalar

The Solution via Gauss Elimination

Algorithm: Gauss Elimination with Pivoting

Given: $n \times n$ coefficient matrix A and $n \times 1$ vector $\mathbf{b}$

$$A\mathbf{u} = \mathbf{b}$$

Find: unknowns $u_1, \ldots, u_n$ of $n \times 1$ vector $\mathbf{u}$

The Solution via Gauss Elimination

Initialize the $n \times n$ matrix $G = I$
For $j = 1, \ldots, n-1$ (j counts columns)

Pivoting step:
Find element in largest absolute value in column j
from $a_{j,j}$ to $a_{n,j}$; this is element $a_{r,j}$
If $r > j$, exchange equations r and j
If $a_{j,j} = 0$, the system is not solvable

Forward elimination step for column j:
For $i = j+1, \ldots, n$ (elements below diagonal of column j)
Construct the *multiplier* $g_{i,j} = a_{i,j}/a_{j,j}$
$a_{i,j} = 0$
For $k = j+1, \ldots, n$ (each element in row i after column j)
$a_{i,k} = a_{i,k} - g_{i,j}a_{j,k}$
$b_i = b_i - g_{i,j}b_j$

All elements below diagonal set to zero $\Rightarrow$ matrix is *upper triangular*

The Solution via Gauss Elimination

Back substitution:
$u_n = b_n / a_{n,n}$
For $j = n-1, \ldots, 1$

$$u_j = \frac{1}{a_{j,j}}(b_j - a_{j,j+1}u_{j+1} - \ldots - a_{j,n}u_n)$$

Programming environment: convenient to form *augmented matrix*
A augmented with the vector $\mathbf{b}$

$$\begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & b_1 \\ a_{2,1} & a_{2,2} & a_{2,3} & b_2 \\ a_{3,1} & a_{3,2} & a_{3,3} & b_3 \end{bmatrix}$$

Then the k steps run to $n+1$
— no need for the extra line for the b_i element

The Solution via Gauss Elimination

Forward elimination steps written in matrix form:
To produce zeroes under $a_{j,j}$ use

$$G_j = \begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & 1 & & & \\ & & & -g_{j+1,j} & 1 & & \\ & & & \vdots & & \ddots & \\ & & & -g_{n,j} & & & 1 \end{bmatrix}$$

Elements $-g_{i,j}$ of G_j are multipliers
G_j is called a Gauss matrix

$$G = G_{n-1}P_{n-1} \cdot \ldots \cdot G_2 \cdot P_2 \cdot G_1 \cdot P_1 \quad \text{then} \quad GA\mathbf{u} = G\mathbf{b}$$

If no pivoting is required: possible to store $g_{i,j}$ in the zero elements of A
For efficiency: do not to (explicitly) multiply A and $\mathbf{b}$ by G

The Solution via Gauss Elimination

Scaling to achieve *row echelon form*

$$\begin{bmatrix} 2 & 2 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 6 \\ 1 \\ 6 \end{bmatrix} \quad \Rightarrow \quad \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 3 \\ -1 \\ 3 \end{bmatrix}$$

If matrix is *rank deficient* (rank $< n$)
$\Rightarrow$ rows with all zeroes should be the last rows

More efficient to do the scaling as part of back substitution

The Solution via Gauss Elimination

Gauss elimination requires $O(n^3)$ operations
$\Rightarrow$ an estimated number of n^3 operations

Algorithm is suitable for a system with thousands of equations
Not suitable for a system with millions of equations

When the system is very large
often times many matrix elements are zero — sparse linear system

Iterative methods are a better approach (discussed in next chapter)

Homogeneous Linear Systems

$$A\mathbf{u} = \mathbf{0}$$

Trivial solution is always an option — but of little interest

How do we use Gauss elimination to find a nontrivial solution if it exists?

Nontrivial solution $\mathbf{u} \Rightarrow c\mathbf{u}$ are solutions as well

The answer: slightly modify the back substitution step

Homogeneous Linear Systems

Example: rank one matrix

$$\begin{bmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \\ 1 & 2 & 3 \end{bmatrix} \mathbf{u} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \qquad \Rightarrow \qquad \begin{bmatrix} 1 & 2 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{u} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

For each *zero row* of the transformed system
set the corresponding u_i — the free variables — to one:

$$\mathbf{u} = \begin{bmatrix} -5 \\ 1 \\ 1 \end{bmatrix}$$

All vectors $c\mathbf{u}$ are solutions

Homogeneous Linear Systems

Previous example: 3×3 rank one matrix
— Two dimensional null space
— Number of free variables $=$ dimension of the null space

Systematically construct two vectors $\mathbf{u}_1, \mathbf{u}_2$ that span the null space
— Set one of the free variables to one and the other to zero

$$\mathbf{u}_1 = \begin{bmatrix} -3 \\ 0 \\ 1 \end{bmatrix} \quad \text{and} \quad \mathbf{u}_2 = \begin{bmatrix} -2 \\ 1 \\ 0 \end{bmatrix}$$

All linear combinations of elements of null space are also in null space
Example: $\mathbf{u} = 1\mathbf{u}_1 + 1\mathbf{u}_2$

Homogeneous Linear Systems

Column pivoting

Example: homogeneous system from an eigenvector problem

$$\begin{bmatrix} 0 & 6 & 3 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{u} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Leads to $0u_3 = 0$ and $2u_3 = 0$ — instead apply column exchanges:

$$\begin{bmatrix} 6 & 3 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_2 \\ u_3 \\ u_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Set the free variable: $u_1 = 1$ — then back substitution

$$\text{Solution: all vectors} \quad c \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Inverse Matrices

Inverse of a square matrix A "undoes" A's action

$$AA^{-1} = I$$

			1	0	−1
			3	1	−3
			1	2	−2
−4	2	−1	1	0	0
−3	1	0	0	1	0
−5	2	−1	0	0	1

Inverse Matrices

How to compute the inverse of an $n \times n$ matrix A?

Vectors $\overline{\mathbf{a}}_i$ and $\mathbf{e}_i$ are $n \times 1$
Vector $\mathbf{e}_i$: zero entries except ith component equals 1

$$A \begin{bmatrix} \overline{\mathbf{a}}_1 & \dots & \overline{\mathbf{a}}_n \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1 & \dots & \mathbf{e}_n \end{bmatrix}$$

n linear systems:

$$A\overline{\mathbf{a}}_1 = \mathbf{e}_1, \quad \dots, \quad A\overline{\mathbf{a}}_n = \mathbf{e}_n$$

Solve with with Gauss elimination:
— Apply forward elimination to A and to each of the $\mathbf{e}_i$
— Back substitution to solve for each $\overline{\mathbf{a}}_i \Rightarrow A^{-1}$
— More economical to use LU decomposition – next section

Inverse Matrices

Inverse matrices are primarily a theoretical concept

Inverse suggests to solve $A\mathbf{v} = \mathbf{b}$ via $\mathbf{v} = A^{-1}\mathbf{b}$
Don't do that! – very expensive

Gauss elimination or LU decomposition is much cheaper:
— Explicitly forming inverse:

- forward elimination
- n back substitution algorithms
- matrix-vector multiplication

— Gauss elimination:

- forward elimination
- 1 back substitution algorithm

Inverse Matrices

Inverse exists if matrix is $n \times n$ and rank n — *full rank*
$\Rightarrow$ Action of A does not reduce dimensionality
$\Rightarrow$ All columns are linearly independent

Is A invertible?
Perform Gauss elimination
— A upper triangular with all nonzero diagonal elements $\Rightarrow$ invertible
— Otherwise: A is singular

Matrix rank review:
— Matrix does not reduce dimensionality $\Rightarrow$ rank n or full rank
— Matrix reduces dimensionality by $k \Rightarrow$ rank $n - k$
— $n \times n$ identity matrix has rank n
— Zero matrix has rank 0

Inverse Matrices

Apply forward elimination to achieve row echelon form:

$$M_1 = \begin{bmatrix} 1 & 3 & -3 & 0 \\ 0 & 3 & 3 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad \text{rank } 2$$

$$M_2 = \begin{bmatrix} 1 & 3 & -3 & 0 \\ 0 & 3 & 3 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad \text{rank } 3$$

$$M_3 = \begin{bmatrix} 1 & 3 & -3 & 0 \\ 0 & 3 & 3 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} \qquad \text{rank } 4$$

Inverse Matrices

Compute the inverse of the $n \times n$ Gauss matrix G_j

$$G_j = \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & -g_{j+1,j} & 1 & \\ & & & \vdots & & \ddots & \\ & & & -g_{n,j} & & & 1 \end{bmatrix} \quad G_j^{-1} = \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & g_{j+1,j} & 1 & \\ & & & \vdots & & \ddots & \\ & & & g_{n,j} & & & 1 \end{bmatrix}$$

G_j is a shear $\Rightarrow$ G_j^{-1} "undoes" G_j

Suppose $k \neq 0$ and kA is an invertible matrix: $(kA)^{-1} = \frac{1}{k}A^{-1}$
A and B are invertible then AB is invertible

LU Decomposition

Forward elimination (no pivoting) in terms of Gauss matrices:

$$G_{n-1} \cdot \ldots \cdot G_1 \cdot A = U$$
$$A = G_1^{-1} \cdot \ldots \cdot G_{n-1}^{-1} U$$

Lower triangular matrix with elements $g_{i,j}$:

$$L = G_1^{-1} \cdot \ldots \cdot G_{n-1}^{-1} = \begin{bmatrix} 1 & & & \\ g_{2,1} & 1 & & \\ \vdots & \ddots & \ddots & \\ g_{n,1} & \cdots & g_{n,n-1} & 1 \end{bmatrix}$$

$A = LU \qquad \Rightarrow \qquad$ LU decomposition of A

Also called the triangular factorization of A

Every invertible matrix has such a decomposition
— pivoting might be necessary

LU Decomposition

$A = LU$ for 3×3 matrix:

$$
\begin{array}{ccc|ccc}
 & & & u_{1,1} & u_{1,2} & u_{1,3} \\
 & & & 0 & u_{2,2} & u_{2,3} \\
 & & & 0 & 0 & u_{3,3} \\
\hline
1 & 0 & 0 & a_{1,1} & a_{1,2} & a_{1,3} \\
l_{2,1} & 1 & 0 & a_{2,1} & a_{2,2} & a_{2,3} \\
l_{3,1} & l_{3,2} & 1 & a_{3,1} & a_{3,2} & a_{3,3}
\end{array}
$$

LU Decomposition

Given: $a_{i,j}$ **Find:** $l_{i,j}$ and $u_{i,j}$

Elements of A below diagonal:

$$a_{i,j} = l_{i,1}u_{1,j} + \ldots + l_{i,j-1}u_{j-1,j} + l_{i,j}u_{j,j}; \quad j < i$$

Elements of A on or above diagonal:

$$a_{i,j} = l_{i,1}u_{1,j} + \ldots + l_{i,i-1}u_{i-1,j} + l_{i,i}u_{i,j}; \quad j \geq i$$

$\Longrightarrow$

$$l_{i,j} = \frac{1}{u_{j,j}}(a_{i,j} - l_{i,1}u_{1,j} - \ldots - l_{i,j-1}u_{j-1,j}); \quad j < i$$

$$u_{i,j} = a_{i,j} - l_{i,1}u_{1,j} - \ldots - l_{i,i-1}u_{i-1,j}; \quad j \geq i$$

LU Decomposition

If A has a decomposition $A = LU$ then system can be written

$$LU\mathbf{u} = \mathbf{b}$$

Solving linear system is a two-step procedure:

$$\begin{aligned} L\mathbf{y} &= \mathbf{b} \qquad \text{where } \mathbf{y} = U\mathbf{u} \\ U\mathbf{u} &= \mathbf{y} \end{aligned}$$

The two systems are triangular and easy to solve:
— Forward substitution applied to L
— Back substitution applied to U

LU Decomposition

Given: Coefficient matrix A and right-hand side $\mathbf{b}$ of $A\mathbf{u} = \mathbf{b}$
Find: The unknowns $u_1, \ldots, u_n$ of $\mathbf{u}$
Algorithm:

Initialize L as the identity matrix and U as the zero matrix
Calculate the nonzero elements of L and U:
For $k = 1, \ldots, n$

$$u_{k,k} = a_{k,k} - l_{k,1}u_{1,k} - \ldots - l_{k,k-1}u_{k-1,k}$$

For $i = k+1, \ldots, n$

$$l_{i,k} = \frac{1}{u_{k,k}}(a_{i,k} - l_{i,1}u_{1,k} - \ldots - l_{i,k-1}u_{k-1,k})$$

For $j = k+1, \ldots, n$

$$u_{k,j} = a_{k,j} - l_{k,1}u_{1,j} - \ldots - l_{k,k-1}u_{k-1,j}$$

Using forward substitution solve L$\mathbf{y} = \mathbf{b}$.
Using back substitution solve U$\mathbf{u}$= $\mathbf{y}$

The $u_{k,k}$ term must not be zero $\Rightarrow$ requires pivoting or matrix is singular
L being filled column by column and U being filled row by row

LU Decomposition

Example: $$A = \begin{bmatrix} 2 & 2 & 4 \\ -1 & 2 & -3 \\ 1 & 2 & 2 \end{bmatrix} \quad \mathbf{u} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

First step: decompose A

$$k = 1:$$
$$u_{1,1} = a_{1,1} = 2$$
$$l_{2,1} = a_{2,1}/u_{1,1} = -1/2$$
$$l_{3,1} = a_{3,1}/u_{1,1} = 1/2$$
$$u_{1,2} = a_{1,2} = 2$$
$$u_{1,3} = a_{1,3} = 4$$

$$k = 2:$$
$$u_{2,2} = a_{2,2} - l_{2,1}u_{1,2} = 2 + 1 = 3$$
$$l_{3,2} = \frac{1}{u_{2,2}}[a_{3,2} - l_{3,1}u_{1,2}] = \frac{1}{3}[2 - 1] = 1/3$$
$$u_{2,3} = a_{2,3} - l_{2,1}u_{1,3} = -3 + 2 = -1$$

$$k = 3: \qquad u_{3,3} = a_{3,3} - l_{3,1}u_{1,3} - l_{3,2}u_{2,3} = 2 - 2 + 1/3 = 1/3$$

Check decomposition:

			2	2	4
			0	3	−1
			0	0	1/3
1	0	0	2	2	4
−1/2	1	0	−1	2	− 3
1/2	1/3	1	1	2	2

LU Decomposition

Next: solve $L\mathbf{y} = \mathbf{b}$ with forward substitution
— solving for y_1, then y_2, and then y_3

$$\mathbf{y} = \begin{bmatrix} 1 \\ 3/2 \\ 0 \end{bmatrix}$$

Last step: solve $U\mathbf{u} = \mathbf{y}$ with back substitution

$$\mathbf{u} = \begin{bmatrix} 0 \\ 1/2 \\ 0 \end{bmatrix}$$

Simple to check that solution correct: $\mathbf{a}_2$ is a multiple of $\mathbf{b}$

LU Decomposition

Suppose A is nonsingular, but in need of pivoting
— Permutation matrix P used to exchange row(s)
— System becomes $PA\mathbf{u} = P\mathbf{b}$ and find $PA = LU$

Major benefit of the LU decomposition: speed

Solving multiple linear systems with the same coefficient matrix
— Construct decomposition
— Perform the forward and backward substitutions for each right-hand side

Example: finding the inverse of a matrix

Chapter 8 3D Geometry: *scalar triple product* to measure volume in 3D
— Provided a geometric derivation of 3×3 *determinants*
Now: $n \times n$ determinants

Matrix A transformed to upper triangular U via forward elimination
— Sequence of shears and row exchanges
— Shears do not change volumes
— Row exchange changes the sign of the determinant
$\Rightarrow$ column vectors of U span same volume as A

$$\det A = (-1)^k (u_{1,1} \times \ldots \times u_{n,n})$$

where k is the number of row exchanges

One of the best (and most stable) methods for computing the determinant

Example from the Gauss Elimination Section – one row exchange ($k = 1$):

$$A = \begin{bmatrix} 2 & 2 & 0 \\ 1 & 1 & 2 \\ 2 & 1 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 6 \\ 9 \\ 7 \end{bmatrix} \quad \rightarrow \quad U = \begin{bmatrix} 2 & 2 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 6 \\ 1 \\ 6 \end{bmatrix}$$

Method 1: Cofactor expansion

$$\det A = 2 \begin{vmatrix} 1 & 2 \\ 1 & 1 \end{vmatrix} - 2 \begin{vmatrix} 1 & 2 \\ 2 & 1 \end{vmatrix} = 4$$

Method 2: Product of diagonal elements of U

$$\det A = (-1)^1 [2 \times -1 \times 2] = 4$$

Determinants

Cofactor expansion for $n \times n$ matrices

Choose any column or row of the matrix – for example entries $a_{1,j}$

$$\det A = a_{1,1}C_{1,1} + a_{1,2}C_{1,2} + \ldots + a_{1,n}C_{1,n}$$

where each cofactor is defined as

$$C_{i,j} = (-1)^{i+j} M_{i,j}$$

$M_{i,j}$ are called the minors
— Each is determinant with i^{th} row and j^{th} column removed
— Each is an $(n-1) \times (n-1)$ determinant
— Each computed by yet another cofactor expansion
Process repeated until reduced to 2×2 determinants
Technique also known as expansion by minors

Example:

$$A = \begin{bmatrix} 2 & 2 & 0 & 4 \\ 0 & -1 & 1 & 3 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix}$$

Choose the first column to form the cofactors
— Minimize number of non-zero cofactors

$$\det A = 2 \begin{vmatrix} -1 & 1 & 3 \\ 0 & 2 & 0 \\ 0 & 0 & 5 \end{vmatrix} = 2(-1) \begin{vmatrix} 2 & 0 \\ 0 & 5 \end{vmatrix} = 2(-1)(10) = -20$$

Since matrix is in upper triangular form — could also compute as

$$\det A = (-1)^0 (2 \times -1 \times 2 \times 5) = -20$$

Cofactor expansion is more a *theoretical tool* than a computational one
— Important theoretical role in the analysis of linear systems
— Advanced theorems involving cofactor expansion and the inverse

Computationally: Gauss elimination and the calculation of det U is superior

Revisit *Cramer's rule* – solution to $n \times n$ $A\mathbf{u} = \mathbf{b}$:
— Necessary that $\det A \neq 0$

$$u_1 = \frac{\det A_1}{\det A} \quad u_2 = \frac{\det A_2}{\det A} \quad \dots \quad u_n = \frac{\det A_n}{\det A}$$

where A_i is matrix obtained by replacing entries in the i^{th} column by $\mathbf{b}$
Cramer's rule is an important *theoretical tool*
— Only use it for 2×2 or 3×3 linear systems

Example of Cramer's rule:

$$A = \begin{bmatrix} 2 & 2 & 0 \\ 1 & 1 & 2 \\ 2 & 1 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 6 \\ 9 \\ 7 \end{bmatrix}$$

$$u_1 = \frac{\begin{vmatrix} 6 & 2 & 0 \\ 9 & 1 & 2 \\ 7 & 1 & 1 \end{vmatrix}}{\begin{vmatrix} 2 & 2 & 0 \\ 1 & 1 & 2 \\ 2 & 1 & 1 \end{vmatrix}} \quad u_2 = \frac{\begin{vmatrix} 2 & 6 & 0 \\ 1 & 9 & 2 \\ 2 & 7 & 1 \end{vmatrix}}{\begin{vmatrix} 2 & 2 & 0 \\ 1 & 1 & 2 \\ 2 & 1 & 1 \end{vmatrix}} \quad u_3 = \frac{\begin{vmatrix} 2 & 2 & 6 \\ 1 & 1 & 9 \\ 2 & 1 & 7 \end{vmatrix}}{\begin{vmatrix} 2 & 2 & 0 \\ 1 & 1 & 2 \\ 2 & 1 & 1 \end{vmatrix}}$$

$$u_1 = \frac{4}{4} = 1 \qquad u_2 = \frac{8}{4} = 2 \qquad u_3 = \frac{12}{4} = 3$$

Identical to solution found with Gauss elimination

Determinants

Determinant of a *positive definite matrix* is always positive
$\Rightarrow$ matrix is always nonsingular

Upper-left submatrices of an $n \times n$ matrix A are

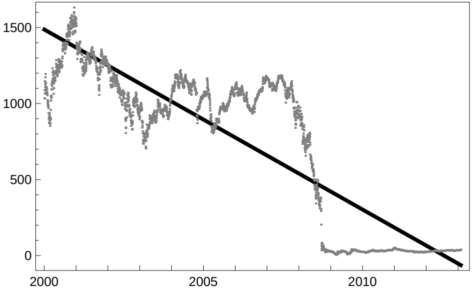
$$A_1 = \begin{bmatrix} a_{1,1} \end{bmatrix} \quad A_2 = \begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a2,2 \end{bmatrix} \quad \dots \quad A_n = A$$

(Different from A_i in Cramer's rule)
If A is positive definite then the determinants of all A_i are positive

Rules for working with determinants: see Chapter 9 Linear Maps in 3D

Least Squares

Presented with large amounts of data
— Need method to create a simpler view or synopsis of the data
— Example: graph of AIG's monthly average stock price over twelve years
A lot of activity in the price, but a clear declining trend



Mathematical tool to capture this: linear least squares approximation
— "Best fit" line or best approximating line

Least Squares

Linear least squares approximation also useful when analyzing experimental data

— Data can be “noisy”

- data capture method encounters error
- observation method lapse
- round-off from computations that generated the data

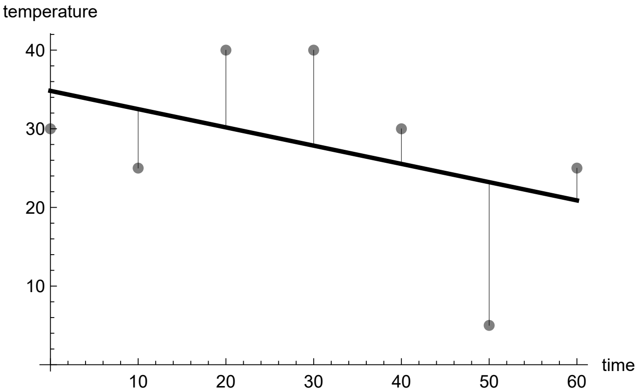
— Might want to

- make summary statements about data
- estimate values where data missing
- predict future values

Least Squares

Example: Experimental data of temperature (Celsius) over time (seconds)

$$\begin{bmatrix} \text{time} \\ \text{temperature} \end{bmatrix} \quad \begin{bmatrix} 0 \\ 30 \end{bmatrix} \quad \begin{bmatrix} 10 \\ 25 \end{bmatrix} \quad \begin{bmatrix} 20 \\ 40 \end{bmatrix} \quad \begin{bmatrix} 30 \\ 40 \end{bmatrix} \quad \begin{bmatrix} 40 \\ 30 \end{bmatrix} \quad \begin{bmatrix} 50 \\ 5 \end{bmatrix} \quad \begin{bmatrix} 60 \\ 25 \end{bmatrix}$$



Least Squares

Want to establish a simple linear relationship between the variables

$$\text{temperature} = u_1 \times \text{time} + u_2$$

Write down relationships between knowns and unknowns:

$$\begin{bmatrix} 0 & 1 \\ 10 & 1 \\ 20 & 1 \\ 30 & 1 \\ 40 & 1 \\ 50 & 1 \\ 60 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 30 \\ 25 \\ 40 \\ 40 \\ 30 \\ 5 \\ 25 \end{bmatrix} \qquad A\mathbf{u} = \mathbf{b}$$

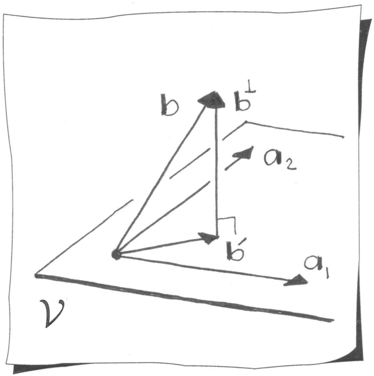
Overdetermined system of 7 equations in 2 unknowns

— In general: will not have solutions; it is *inconsistent*

Unlikely that **b** lives in subspace $\mathcal{V}$ formed by columns of A

$\Rightarrow$ Find an approximate solution

Derivation of the least squares solution



Let $\mathbf{b}'$ be a vector in $\mathcal{V}$
(subspace formed by columns of A)

$$A\mathbf{u} = \mathbf{b}'$$

System is solvable (consistent)
— still overdetermined
(7 equations in 2 unknowns)

$$\mathbf{b} = \mathbf{b}' + \mathbf{b}^{\perp}$$

$\mathbf{b}'$ is closest to $\mathbf{b}$ and in $\mathcal{V}$

Least Squares

$\mathbf{b}^\perp$ is orthogonal to $\mathcal{V}$

$$\mathbf{a}_1^{\mathrm{T}}\mathbf{b}^\perp = 0 \quad \text{and} \quad \mathbf{a}_2^{\mathrm{T}}\mathbf{b}^\perp = 0 \qquad \Rightarrow \qquad A^{\mathrm{T}}\mathbf{b}^\perp = \mathbf{0}$$

$$\begin{aligned} \mathbf{b}^\perp = \mathbf{b} - \mathbf{b}' \quad \text{then } A^{\mathrm{T}}(\mathbf{b} - \mathbf{b}') &= \mathbf{0} \\ A^{\mathrm{T}}(\mathbf{b} - A\mathbf{u}) &= \mathbf{0} \\ A^{\mathrm{T}}\mathbf{b} - A^{\mathrm{T}}A\mathbf{u} &= \mathbf{0} \end{aligned}$$

Rearranging results in the normal equations

$$A^{\mathrm{T}}A\mathbf{u} = A^{\mathrm{T}}\mathbf{b}$$

Linear system with a square, symmetric matrix $A^{\mathrm{T}}A$
Solution to the new system minimizes the *error*

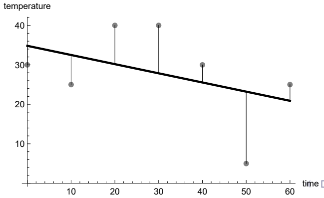
$$\|A\mathbf{u} - \mathbf{b}\|^2 \qquad \Rightarrow \qquad \textit{least squares solution}$$

Recall: $\mathbf{b}'$ is closest to $\mathbf{b}$ in $\mathcal{V} \Rightarrow$ minimizes $\|\mathbf{b}' - \mathbf{b}\|$

Continue Example — Form the normal equations

$$\begin{bmatrix} 0 & 1 \\ 10 & 1 \\ 20 & 1 \\ 30 & 1 \\ 40 & 1 \\ 50 & 1 \\ 60 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 30 \\ 25 \\ 40 \\ 40 \\ 30 \\ 5 \\ 25 \end{bmatrix} \quad \rightarrow \quad \begin{bmatrix} 9100 & 210 \\ 210 & 7 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 5200 \\ 195 \end{bmatrix}$$

Least squares solution $\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} -0.23 \\ 34.8 \end{bmatrix}$ line $x_2 = -0.23x_1 + 34.8$



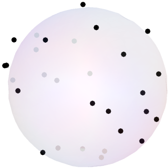
Real-world problem:
Data capture method fails due to some environmental condition
Want to remove data points if they seem outside the norm
— Such data called outliers
— Point six in Figure looks to be an outlier
— Least squares line provides a means for finding outliers

Least squares approximation can be used for *data compression*

Numerical problems can creep into the normal equations
— Particularly so when the $n \gg m$ in $n \times m$ matrix A
— Other methods to find least squares solution
Chapter 13: the Householder method
Chapter 16: SVD

Application: Fitting Data to a Femoral Head

Hip bone replacement:
— Remove an existing femoral head and replace it by a transplant
— Consists of new head and shaft for attaching to existing femur
— Data points collected from existing femoral head with MRI or PET
— Spherical fit is obtained
— Transplant is manufactured



Application: Fitting Data from a Femoral Head

Given: a set of 3D vectors $\mathbf{v}_1, \ldots, \mathbf{v}_L$
— approximately of equal length: $\rho_1, \ldots, \rho_L$
Find: a sphere (centered at the origin) with radius r closely fitting the $\mathbf{v}_i$

If all $\mathbf{v}_i$ on the desired sphere $r = \rho_1, \ldots, r = \rho_L$
In matrix form:

$$\begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \begin{bmatrix} r \end{bmatrix} = \begin{bmatrix} \rho_1 \\ \vdots \\ \rho_L \end{bmatrix}$$

A very overdetermined linear system — L equations in only 1 unknown r
Multiply both sides by $[1 \quad \ldots \quad 1]$ gives

$$Lr = \rho_1 + \ldots + \rho_L \qquad \Rightarrow \qquad r = \frac{\rho_1 + \ldots + \rho_L}{L}$$

Least squares solution is simply the average of the given radii

WYSK

- $n \times n$ linear system
- coefficient matrix
- consistent system
- subspace
- solvable system
- unsolvable system
- Gauss elimination
- upper triangular matrix
- forward elimination
- back substitution
- elementary row operation
- permutation matrix
- row echelon form
- pivoting
- Gauss matrix
- multiplier
- augmented matrix
- singular matrix
- matrix rank
- full rank
- rank deficient
- homogeneous linear system
- inverse matrix
- LU decomposition
- factorization
- forward substitution
- lower triangular matrix
- determinant
- cofactor expansion
- expansion by minors
- Cramer's rule
- overdetermined system
- least squares solution
- normal equations