MATH 532: Linear Algebra

Chapter 3: Matrix Algebra

Greg Fasshauer

Department of Applied Mathematics
Illinois Institute of Technology

Spring 2015

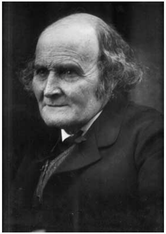
Outline

We will briefly go over some ideas from Chapters 1, 2 and the first half of Chapter 3 of the textbook [Mey00].

After that introduction we will start our real journey with Section 3.7 and the inverse of a matrix.

Linear algebra is an old subject

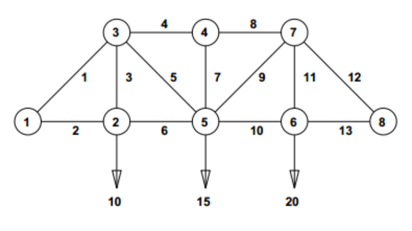
- The origins are attributed to the solution of systems of linear equations in China around 200 BC [NinBC, Chapter 8].
    - Look at Episode 2 (10:23–13:20) of *The Story of Maths*.
    - Look at [Yua12].
- In the West, the same algorithm became known as Gaussian elimination, named after Carl Friedrich Gauß (1755–1855).

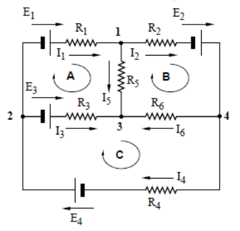



- “Modern” linear algebra is associated with Arthur Cayley (1821–1895), and many others after him.
- Recent developments have focused mostly on numerical linear algebra.

Linear algebra appears in many fields and guises:

- Numerical analysis: discretization of DEs [Mey00, Ch. 1.4]
- Mechanical/structural engineering: plane trusses [Mol08]
- Electrical engineering: electric circuits [Mey00, Ch. 2.6]





- Data science and statistics: regression

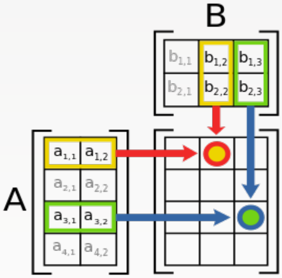
$$\min_{\boldsymbol{x}\in\mathbb{R}^m} \|A\boldsymbol{x}-\boldsymbol{b}\|_2 \quad \Longrightarrow \quad \boldsymbol{x} = (A^TA)^{-1}A^T\boldsymbol{b}$$

- Machine learning: regularization networks

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \left[L(\boldsymbol{b}, A\boldsymbol{x}) + \mu\|\boldsymbol{x}\|\right], \quad \text{e.g.,} \quad \min_{\boldsymbol{x}\in\mathbb{R}^n} \|A\boldsymbol{x}-\boldsymbol{b}\|_2 + \mu\boldsymbol{x}^TA\boldsymbol{x}$$

Different forms of matrix products

We all know how to multiply two matrices A and B:



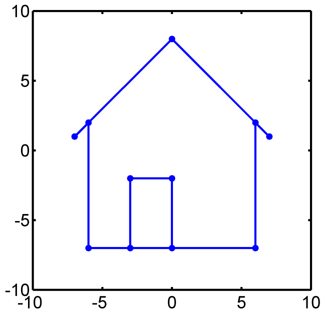
But why do we do it this way?

- Because Cayley said so.
- Because it works for systems of linear equations and for linear transformations, i.e., scalings, rotations, reflections and shear maps can be expressed as a matrix product.

Matrices as Linear Transformations

We illustrate properties of linear transformations (matrix multiplication by A) with the following “data”:

```
X = house
dot2dot(X)
```

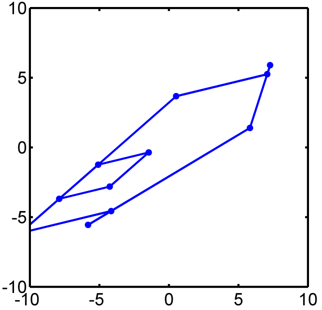


Straight lines are always mapped to straight lines.

```
A = rand(2,2)
dot2dot(A*X)
```

Sample matrix

$$A = \begin{bmatrix} 0.9357 & 0.7283 \\ 0.8187 & 0.1758 \end{bmatrix}$$

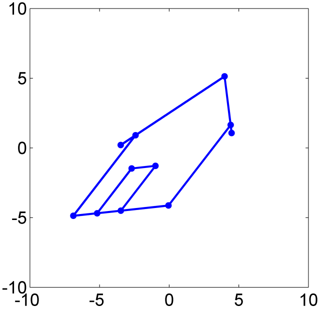


The transformation is orientation-preserving[1] if $\det A > 0$.

```
A = rand(2,2)
det(A)
dot2dot(A*X)
```

Sample matrix

$$A = \begin{bmatrix} 0.5694 & 0.4963 \\ 0.0614 & 0.6423 \end{bmatrix}$$



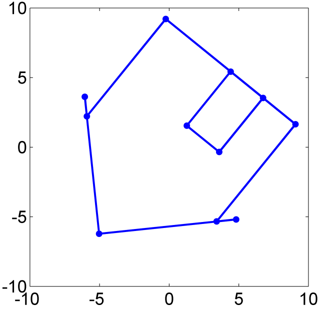
[1] The door always stays on the left.

The angles between straight lines are preserved if the matrix is orthogonal[2].

```
A = orth(rand(2,2));   % creates orthogonal matrix
A = A(:,randperm(2))   % randomly permute columns of A
det(A)
dot2dot(A*X)
```

Sample matrix

$$A = \begin{bmatrix} -0.7767 & -0.6299 \\ 0.6299 & -0.7767 \end{bmatrix}$$



[2] An orthogonal matrix A has det A $\pm$ 1 and represents either a rotation or a reflection.

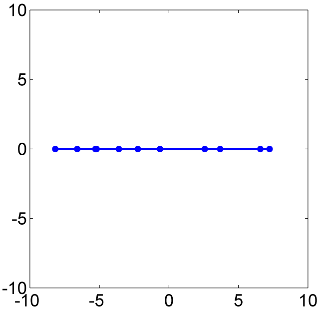
A linear transformation is invertible[3] only if $\det A \neq 0$.

```
a22 = randi(3,1,1)-2  % creates random {-1,0,1}
A = triu(rand(2,2)); A(2,2) = a22
det(A)
dot2dot(A*X)
```

Sample matrix

$$A = \begin{bmatrix} 0.9884 & 0.3209 \\ 0 & 0 \end{bmatrix}$$

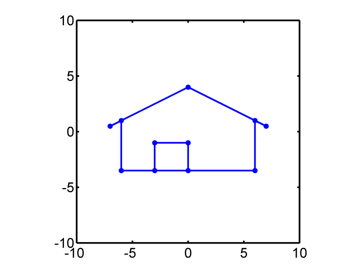
$$\det A = 0$$



[3]If the transformation is not invertible, then the 2D image collapses to a line or even a point.

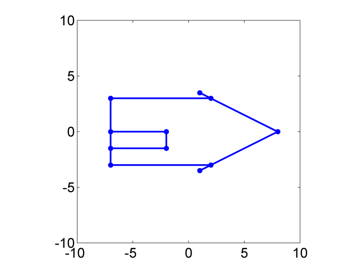
A diagonal matrix stretches the image or reverses its orientation.

$$A = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{2} \end{bmatrix}, \quad \det A = \frac{1}{2}$$



A anti-diagonal matrix in addition interchanges coordinates.

$$A = \begin{bmatrix} 0 & 1 \\ \frac{1}{2} & 0 \end{bmatrix}, \quad \det A = -\frac{1}{2}$$



The action of a diagonal matrix provides an interpretation of the effect of eigenvalues. Note that these matrices have orthogonal columns, but their determinant is not ± 1, so they are **not** orthogonal matrices. These matrices preserve right angles only.

Any rotation matrix can be expressed in terms of trigonometric functions:
The matrix

$$G(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

represents a counter-clockwise rotation by the angle θ (measured in radians).

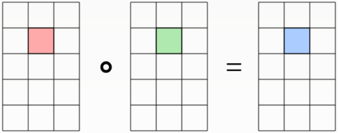
Look at `wiggle.m`.

Matrix multiplication: Why we do it the way we do it

- Because the most obvious way, i.e.,

$$[A \circ B]_{ij} = [A]_{ij}[B]_{ij},$$

known as Hadamard[4] (or Schur) product, doesn't work for linear systems and linear transformations.



- It's also defined only for matrices of the same size.
- But it does commute.

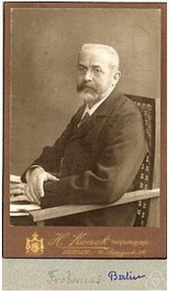
[4] Jacques Hadamard (1865–1963) and Issai Schur (1875–1941)

Matrix multiplication: Why we do it the way we do it

- Because the Frobenius[5] (inner) product,

$$\langle A, B \rangle_F = \sum_{i,j} [A]_{ij} [B]_{ij},$$

doesn't work for linear systems or linear transformations either.



- It is also requires $\text{size}(A) = \text{size}(B)$.
- It does, however, induce a useful matrix norm (see HW).

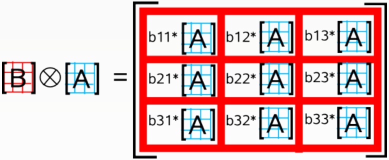
[5] Georg Frobenius (1849–1917)

Matrix multiplication: Why we do it the way we do it

- Because the Kronecker[6] product,

$$A \otimes B = \begin{pmatrix} [A]_{11}B & \cdots & [A]_{1n}B \\ \vdots & & \vdots \\ [A]_{m1}B & \cdots & [A]_{mn}B \end{pmatrix},$$

doesn't work for linear systems or linear transformations either.



- Works for matrices of arbitrary size, i.e., A is $m \times n$, B is $p \times q$.
- Ideal for working with tensor products $\rightsquigarrow$ multilinear algebra

[6] Leopold Kronecker (1823–1891)

Modern research on matrix multiplication

How to do them fast! Naive matrix multiplication of two $n \times n$ matrices requires $\mathcal{O}(n^3)$ operations (and must be at least $\mathcal{O}(n^2)$, since each element must be touched at least once)

- Special algorithms for general matrices:
  - Strassen's algorithm [Str69] $\mathcal{O}(n^{2.807})$,
  - Coppersmith–Winograd algorithm [CW90] $\mathcal{O}(n^{2.375})$,
  - Stothers' algorithm [DS13] $\mathcal{O}(n^{2.374})$,
  - Williams' algorithm [Wil14] $\mathcal{O}(n^{2.3729})$,
  - Le Gall's algorithm [LG14] $\mathcal{O}(n^{2.3728639})$.
  - A bet: `http://www.math.utah.edu/~pa/bet.html`
- Exploiting structure (banded, block, hierarchical) — often implied by application
- Using factorizations, into products of structured matrices
- Exploiting sparsity
- Exploiting new hardware

Definition (Matrix inverse)

For any $n \times n$ matrix A, the $n \times n$ matrix B that satisfies

$$\mathsf{AB} = \mathsf{I} \quad \text{and} \quad \mathsf{BA} = \mathsf{I}$$

is called the *inverse* of A.
We use the notation $\mathsf{B} = \mathsf{A}^{-1}$ to denote the inverse of A.

Terminology: If A^{-1} exists, then A is called nonsingular or invertible.

Remark

1. *The inverse of a matrix is unique. To verify, assume* B_1 *and* B_2 *are both inverses of* A*. Then*

$$\mathsf{B}_1 = \mathsf{B}_1\mathsf{I} = \mathsf{B}_1(\mathsf{AB}_2) = (\mathsf{B}_1 A)\mathsf{B}_2 = \mathsf{IB}_2 = \mathsf{B}_2.$$

2. *Sometimes one can find the notion of a left- and right-inverse. However, we consider only inverses of square matrices, so these notions don't apply (see also [Mey00, Ex. 3.7.2]).*

How to compute A^{-1}

- If we do it by hand, we use Gauss–Jordan elimination on $\left(A \mid I \right)$.
- If we do it by computer, we solve $AB = I$ for $B = A^{-1}$.
    - In MATLAB: `invA = A\eye(n)`

Example

Compute the inverse of

$$A = \begin{pmatrix} 2 & 2 & 6 \\ 2 & 1 & 7 \\ 2 & -6 & -7 \end{pmatrix}.$$

Solution

$$\left(\begin{array}{ccc|ccc} 2 & 2 & 6 & 1 & 0 & 0 \\ 2 & 1 & 7 & 0 & 1 & 0 \\ 2 & -6 & -7 & 0 & 0 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} 2 & 2 & 6 & 1 & 0 & 0 \\ 0 & -1 & 1 & -1 & 1 & 0 \\ 0 & -8 & -13 & -1 & 0 & 1 \end{array}\right)$$

$$\left(\begin{array}{ccc|ccc} 2 & 2 & 6 & 1 & 0 & 0 \\ 0 & -1 & 1 & -1 & 1 & 0 \\ 0 & 0 & -21 & 7 & -8 & 1 \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} 1 & 1 & 3 & \frac{1}{2} & 0 & 0 \\ 0 & -1 & 1 & -1 & 1 & 0 \\ 0 & 0 & 1 & -\frac{1}{3} & \frac{8}{21} & -\frac{1}{21} \end{array}\right)$$

Up to here this is Gaussian elimination

$$\left(\begin{array}{ccc|ccc} 1 & 1 & 0 & \frac{3}{2} & -\frac{8}{7} & \frac{1}{7} \\ 0 & 1 & 0 & \frac{2}{3} & -\frac{13}{21} & -\frac{1}{21} \\ 0 & 0 & 1 & -\frac{1}{3} & \frac{8}{21} & -\frac{1}{21} \end{array}\right) \longrightarrow \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & \frac{5}{6} & -\frac{11}{21} & \frac{4}{21} \\ 0 & 1 & 0 & \frac{2}{3} & -\frac{13}{21} & -\frac{1}{21} \\ 0 & 0 & 1 & -\frac{1}{3} & \frac{8}{21} & -\frac{1}{21} \end{array}\right)$$

Gauss–Jordan elimination is not good for solving linear systems, but useful for some theoretical purposes.

How to check if A is invertible

Theorem

For any $n \times n$ matrix A, the following statements are equivalent:

1. A^{-1} *exists*
2. $\operatorname{rank}(\mathsf{A}) = n$
3. *Gauss–Jordan elimination reduces* A *to* I
4. $\mathsf{A}\boldsymbol{x} = \mathbf{0}$ *has only the trivial solution* $\boldsymbol{x} = \mathbf{0}$
5. $\det(\mathsf{A}) \neq 0$
6. *Zero is not an eigenvalue of* A
7. *Zero is not a singular value of* A

Proof.

Items (1)–(4) are proved in [Mey00]. Items (5)–(7) are discussed later (but should probably be familiar concepts). □

Inverse of a matrix product

Theorem

If A *and* B *are invertible, then* AB *is also invertible and*

$$(AB)^{-1} = B^{-1}A^{-1}.$$

Proof.

Just use the definition to verify invertibility:

$$(AB)\,B^{-1}A^{-1} = A\underbrace{\left(BB^{-1}\right)}_{=I}A^{-1} = I$$

Since the inverse is unique we are done. □

Inverse of a matrix sum

A simple example shows that — just because A and B are invertible — the inverse of $A + B$ need not exist!

Example

Let

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \text{and} \qquad B = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$$

Then $A + B$ is the zero matrix, which is obviously not invertible.

Inverse of a matrix sum (cont.)

Moreover, the inverse is not a linear function.

Even in the scalar case we have (the breaking point in the education of many a young "mathematician"?):

Example

Let $a = 2$ and $b = 3$. Then

- $a + b = 5$, and so $(a + b)^{-1} = \frac{1}{5}$;
- $a^{-1} = \frac{1}{2}$ and $b^{-1} = \frac{1}{3}$.

And now we see/know that

$$(a + b)^{-1} \neq a^{-1} + b^{-1} \qquad \text{since} \qquad \frac{1}{5} \neq \frac{1}{2} + \frac{1}{3}.$$

So, how do we compute the inverse of $A + B$?

It can be done if one assumes that A and B are such that the inverse exists. The following theorem was proved only in 1981 [HS81].

Theorem (Henderson–Searle)

Suppose the $n \times n$ *matrix* A *is invertible, and let* C *be* $n \times p$, B *be* $p \times q$ *and* D *be* $q \times n$. *Also assume that* $(A + CBD)^{-1}$ *exists. Then*

1. $(A + CBD)^{-1} = A^{-1} - \left(I_n + A^{-1}CBD\right)^{-1} A^{-1}CBDA^{-1}$,
2. $(A + CBD)^{-1} = A^{-1} - A^{-1}\left(I_n + CBDA^{-1}\right)^{-1} CBDA^{-1}$,
3. $(A + CBD)^{-1} = A^{-1} - A^{-1}C\left(I_p + BDA^{-1}C\right)^{-1} BDA^{-1}$,
4. $(A + CBD)^{-1} = A^{-1} - A^{-1}CB\left(I_q + DA^{-1}CB\right)^{-1} DA^{-1}$,
5. $(A + CBD)^{-1} = A^{-1} - A^{-1}CBD\left(I_n + A^{-1}CBD\right)^{-1} A^{-1}$,
6. $(A + CBD)^{-1} = A^{-1} - A^{-1}CBDA^{-1}\left(I_n + CBDA^{-1}\right)^{-1}$.

Before we prove (part of) this theorem, let's see what this says about $(A+B)^{-1}$.

Corollary

In the theorem, let all matrices be $n \times n$ and let $C = D = I$. Then

1. $(A+B)^{-1} = A^{-1} - \left(I + A^{-1}B\right)^{-1} A^{-1}BA^{-1}$,
2. $(A+B)^{-1} = A^{-1} - A^{-1}\left(I + BA^{-1}\right)^{-1} BA^{-1}$,
3. $(A+B)^{-1} = A^{-1} - A^{-1}B\left(I + A^{-1}B\right)^{-1} A^{-1}$,
4. $(A+B)^{-1} = A^{-1} - A^{-1}BA^{-1}\left(I + BA^{-1}\right)^{-1}$.

Note that only four formulas are left.

To prove this theorem one needs

Lemma

Suppose A *is an* $n \times n$ *matrix such that* $\mathsf{I} + \mathsf{A}$ *is invertible. Then*

$$(\mathsf{I}+\mathsf{A})^{-1} = \mathsf{I} - \mathsf{A}(\mathsf{I}+\mathsf{A})^{-1} \tag{1a}$$

$$= \mathsf{I} - (\mathsf{I}+\mathsf{A})^{-1}\mathsf{A}. \tag{1b}$$

In particular,

$$\mathsf{A}(\mathsf{I}+\mathsf{A})^{-1} = (\mathsf{I}+\mathsf{A})^{-1}\mathsf{A}. \tag{2}$$

Proof.

(2) follows immediately from (1).
To prove (1), we start with

$$\mathsf{I} = (\mathsf{I}+\mathsf{A}) - \mathsf{A}.$$

Now multiply by $(\mathsf{I}+\mathsf{A})^{-1}$ from either the right (to get (1a)), or from the left (to get (1b)). □

Proof of the theorem.

We prove only the first identity.
We note that $I_n + A^{-1}CBD = A^{-1}(A + CBD)$, where both factors are invertible by assumption. Therefore $\left(I_n + A^{-1}CBD\right)^{-1}$ exists.
Then

$$\begin{aligned}
(A + CBD)^{-1} &= \left(A\left(I_n + A^{-1}CBD\right)\right)^{-1} && \text{from above} \\
&= \left(I_n + A^{-1}CBD\right)^{-1} A^{-1} && \text{since } (A\widetilde{B})^{-1} = \widetilde{B}^{-1}A^{-1} \\
&\overset{(1b)}{=} \left(I_n - \left(I_n + A^{-1}CBD\right)^{-1} A^{-1}CBD\right) A^{-1} && \widetilde{A} = A^{-1}CBD \\
&= A^{-1} - \left(I_n + A^{-1}CBD\right)^{-1} A^{-1}CBDA^{-1}.
\end{aligned}$$

Note that the other identities are not proven analogously. They require extra work. □

Sherman–Morrison formula

The following formula is older (from 1949–50), but can also be derived as a corollary from the Henderson–Searle theorem.

Corollary

Suppose that the $n \times n$ matrix A *is invertible, and also suppose that* $\alpha \in \mathbb{R}$ *and the column n-vectors* $\boldsymbol{c}$ *and* $\boldsymbol{d}$ *are such that* $1 + \alpha \boldsymbol{d}^T \mathsf{A}^{-1} \boldsymbol{c} \neq 0$*. Then* $\mathsf{A} + \alpha \boldsymbol{c}\boldsymbol{d}^T$ *is invertible and*

$$\left(\mathsf{A} + \alpha \boldsymbol{c}\boldsymbol{d}^T\right)^{-1} = \mathsf{A}^{-1} - \frac{\alpha \mathsf{A}^{-1} \boldsymbol{c}\boldsymbol{d}^T \mathsf{A}^{-1}}{1 + \alpha \boldsymbol{d}^T \mathsf{A}^{-1} \boldsymbol{c}}.$$

Note that $\alpha \boldsymbol{c}\boldsymbol{d}^T$ is a rank-1 update of A.

Remark

The Sherman–Morrison–Woodbury formula follows analogously and is stated in [Mey00].

Proof.

We use the fourth identity of the Henderson–Searle theorem with $\mathsf{B} = \alpha$, $\mathsf{C} = \boldsymbol{c}$ and $\mathsf{D} = \boldsymbol{d}^T$ (so that $p = q = 1$).
Then

$$(\mathsf{A} + \mathsf{CBD})^{-1} = \mathsf{A}^{-1} - \mathsf{A}^{-1}\mathsf{CB}\left(\mathsf{I}_q + \mathsf{DA}^{-1}\mathsf{CB}\right)^{-1}\mathsf{DA}^{-1}$$

becomes

$$\begin{aligned}\left(\mathsf{A} + \alpha\boldsymbol{c}\boldsymbol{d}^T\right)^{-1} &= \mathsf{A}^{-1} - \alpha\mathsf{A}^{-1}\boldsymbol{c}\left(1 + \alpha\boldsymbol{d}^T\mathsf{A}^{-1}\boldsymbol{c}\right)^{-1}\boldsymbol{d}^T\mathsf{A}^{-1} \\ &= \mathsf{A}^{-1} - \frac{\mathsf{A}^{-1}\boldsymbol{c}\boldsymbol{d}^T\mathsf{A}^{-1}}{1 + \alpha\boldsymbol{d}^T\mathsf{A}^{-1}\boldsymbol{c}}\end{aligned}$$

since $\boldsymbol{d}^T\mathsf{A}^{-1}\boldsymbol{c}$ is a scalar. □

If $A = I$, $\alpha = -1$ and $\boldsymbol{c}, \boldsymbol{d}$ such that $\boldsymbol{d}^T \boldsymbol{c} \neq 1$ in the Sherman–Morrison formula, then we get

$$\left(I - \boldsymbol{c}\boldsymbol{d}^T\right)^{-1} = I - \frac{\boldsymbol{c}\boldsymbol{d}^T}{\boldsymbol{d}^T\boldsymbol{c} - 1}.$$

- $I - \boldsymbol{c}\boldsymbol{d}^T$ is called an elementary matrix
- $\left(I - \boldsymbol{c}\boldsymbol{d}^T\right)^{-1}$ is also an elementary matrix, i.e.,

  the inverse of an elementary matrix is an elementary matrix.

We will use such elementary matrices in the next section.

Example

Assume we have worked hard to calculate A^{-1}, and now we change one entry, $[A]_{ij}$, of A. What is the new inverse?
Note that a change of α to $[A]_{ij}$ is given by $\alpha \boldsymbol{e}_i \boldsymbol{e}_j^T$.
We can apply the Sherman–Morrison formula with $\boldsymbol{c} = \boldsymbol{e}_i, \boldsymbol{d} = \boldsymbol{e}_j$:

$$\left(A + \alpha \boldsymbol{e}_i \boldsymbol{e}_j^T\right)^{-1} = A^{-1} - \frac{\alpha A^{-1} \boldsymbol{e}_i \boldsymbol{e}_j^T A^{-1}}{1 + \alpha \boldsymbol{e}_j^T A^{-1} \boldsymbol{e}_i} = A^{-1} - \alpha \frac{[A^{-1}]_{*i} [A^{-1}]_{j*}}{1 + \alpha [A^{-1}]_{ji}}.$$

Note that there's no need to recompute the entire inverse (an $\mathcal{O}(n^3)$ effort). All we need to compute is one outer product, two scalar multiplications and a division (which is $\mathcal{O}(n^2)$).

Elementary Matrices and Equivalence

Our goals for the next two sections are to

- obtain a matrix factorization of a nonsingular $n \times n$ matrix A into elementary matrices,
- obtain a representation of Gaussian elimination as a matrix factorization.

The basic operations used in Gaussian elimination are

Type I: interchange of row i with row j,

Type II: multiplication of row i by $\alpha \neq 0$,

Type III: addition of α times row i to row j.

All of these are row operations and can be represented by left-multiplication by an elementary matrix.

Remark

Right-multiplication will result in similar column operations.

Example: Let A be a 3×3 matrix.

1. Interchange of row 2 with row 3 of A, accomplished as $E_1 A$, where

$$E_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

2. Multiplication of row 2 by $\alpha \neq 0$, accomplished as $E_2 A$, where

$$E_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

3. Addition of α times row 2 to row 3, accomplished as $E_3 A$, where

$$E_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \alpha & 1 \end{pmatrix}$$

Example (cont.)

Recall that elementary matrices are of the form $\mathsf{I} - \boldsymbol{c}\boldsymbol{d}^T$.

1. E_1 can be written as

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} = \mathsf{I} - \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix} = \mathsf{I} - \begin{pmatrix} 0 \\ 1 \\ -1 \end{pmatrix} \begin{pmatrix} 0 & 1 & -1 \end{pmatrix}$$
$$= \mathsf{I} - (\boldsymbol{e}_2 - \boldsymbol{e}_3)(\boldsymbol{e}_2 - \boldsymbol{e}_3)^T$$

2. E_2 can be written as

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} = \mathsf{I} - (1-\alpha) \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} = \mathsf{I} - (1-\alpha)\boldsymbol{e}_2\boldsymbol{e}_2^T$$

3. E_3 can be written as

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \alpha & 1 \end{pmatrix} = \mathsf{I} + \alpha \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} = \mathsf{I} + \alpha\boldsymbol{e}_3\boldsymbol{e}_2^T$$

Example

Gaussian elimination with elementary matrices

Earlier we had

$$\begin{pmatrix} 2 & 2 & 6 \\ 2 & 1 & 7 \\ 2 & -6 & -7 \end{pmatrix} \xrightarrow[-R_1+R_3]{-R_1+R_2} \begin{pmatrix} 2 & 2 & 6 \\ 0 & -1 & 1 \\ 0 & -8 & -13 \end{pmatrix} \xrightarrow{-8R_2+R_3} \begin{pmatrix} 2 & 2 & 6 \\ 0 & -1 & 1 \\ 0 & 0 & -21 \end{pmatrix}$$

So

$$\begin{pmatrix} 2 & 2 & 6 \\ 0 & -1 & 1 \\ 0 & 0 & -21 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -8 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 & 2 & 6 \\ 2 & 1 & 7 \\ 2 & -6 & -7 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 7 & -8 & 1 \end{pmatrix} \begin{pmatrix} 2 & 2 & 6 \\ 2 & 1 & 7 \\ 2 & -6 & -7 \end{pmatrix}$$

Note that this is of the form $U = LA$.

Theorem

A matrix A *is nonsingular if and only if it is the product of elementary matrices of types I–III.*

Proof.

"$\Longrightarrow$": If A is nonsingular, then Gauss–Jordan elimination produces

$$\mathsf{E}_k \cdots \mathsf{E}_2 \mathsf{E}_1 \mathsf{A} = \mathsf{I}.$$

Now, since the inverse of an elementary matrix is an elementary matrix (of the same type) we have

$$\mathsf{A} = \mathsf{E}_1^{-1} \mathsf{E}_2^{-1} \cdots \mathsf{E}_k^{-1} \qquad \text{as desired.}$$

"$\Longleftarrow$": Assume $\mathsf{A} = \mathsf{E}_1 \mathsf{E}_2 \cdots \mathsf{E}_k$. Then A is nonsingular since elementary matrices are nonsingular, and so is their product. □

Equivalent matrices

Definition

Two matrices A and B are called equivalent, i.e., $A \sim B$, if

$$PAQ = B$$

for some nonsingular matrices P and Q.

Moreover, A and B are row equivalent, i.e., $A \overset{\text{row}}{\sim} B$, if $PA = B$, and A and B are column equivalent, i.e., $A \overset{\text{col}}{\sim} B$, if $AQ = B$

Remark

Note that P *performs row operations, and* Q *performs column operations on* A.

The following theorem ensures that row operations preserve column relations (an analogous theorem holds for column operations).

Theorem

If $A \overset{row}{\sim} B$, *then*

$$[B]_{*k} = \sum_{j=1}^{n} \alpha_j [B]_{*j} \quad \iff \quad [A]_{*k} = \sum_{j=1}^{n} \alpha_j [A]_{*j}.$$

Before we prove the theorem, we state

Corollary

Since $A \overset{row}{\sim} E_A$[a], *the nonbasic columns of* A *are the same linear combinations of the basic columns of* A *as those of* E_A.

[a] Here E_A is the unique row-reduced echelon form of A (produced via Gauss–Jordan elimination). This equivalence is proved in [Mey00] with a rather long and technical proof.

Example (for the corollary)

$$A = \begin{pmatrix} 1 & 2 & 2 & 3 & 1 \\ 2 & 4 & 4 & 6 & 2 \\ 3 & 6 & 6 & 9 & 6 \\ 1 & 2 & 4 & 5 & 3 \end{pmatrix} \quad \xrightarrow{\text{G–J}} \quad E_A = \begin{pmatrix} \textcircled{1} & 2 & 0 & 1 & 0 \\ 0 & 0 & \textcircled{1} & 1 & 0 \\ 0 & 0 & 0 & 0 & \textcircled{1} \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Since columns 1, 3 and 5 of E_A are basic columns, the same holds for A and

$$\begin{aligned} [E_A]_{*2} &= 2[E_A]_{*1} & \iff & \quad [A]_{*2} = 2[A]_{*1} \\ [E_A]_{*4} &= [E_A]_{*1} + [E_A]_{*3} & \iff & \quad [A]_{*4} = [A]_{*1} + [A]_{*3} \end{aligned}$$

Proof of theorem.

The definition of $A \overset{\text{row}}{\sim} B$ implies the existence of a nonsingular P so that $PA = B$. Then,

$$[B]_{*j} = [PA]_{*j} = P[A]_{*j}. \tag{3}$$

Therefore, if $[A]_{*k} = \sum_{j=1}^{n} \alpha_j [A]_{*j}$, then

$$P[A]_{*k} = \sum_{j=1}^{n} \alpha_j P[A]_{*j} \quad \overset{(3)}{\Longleftrightarrow} \quad [B]_{*k} = \sum_{j=1}^{n} \alpha_j [B]_{*j}.$$

To prove the reverse implication we multiply the identity $[B]_{*k} = \ldots$ by P^{-1}. □

We just saw that row operations reduce A to row-echelon form E_A.

Row and column operations reduce A to rank-normal form.

Theorem

If A *is an* $n \times n$ *matrix with* $\operatorname{rank}(A) = r$*, then*

$$A \sim N_r = \begin{pmatrix} I_r & 0 \\ 0 & 0 \end{pmatrix}.$$

N_r *is called the rank-normal form of* A.

Proof.

We already know $A \overset{\text{row}}{\sim} E_A$, so that $PA = E_A$ with P nonsingular.
Now, if $\operatorname{rank}(A) = r$, then E_A has r basic (unit) columns, and we can reorder the columns of E_A via an appropriate nonsingular Q_1, so that

$$PAQ_1 = E_AQ_1 = \begin{pmatrix} I_r & J \\ 0 & 0 \end{pmatrix}$$

for an appropriate matrix J.
Finally, define $Q_2 = \begin{pmatrix} I_r & -J \\ 0 & I \end{pmatrix}$ so that

$$PAQ_1Q_2 = E_AQ_1Q_2 = \begin{pmatrix} I_r & J \\ 0 & 0 \end{pmatrix} \begin{pmatrix} I_r & -J \\ 0 & I \end{pmatrix} = \begin{pmatrix} I_r & 0 \\ 0 & 0 \end{pmatrix}$$

and $PA\underbrace{Q_1Q_2}_{=Q} = N_r$, i.e., $A \sim N_r$. □

Block matrix version

Corollary

If $\operatorname{rank}(A) = r$ *and* $\operatorname{rank}(B) = s$, *then* $\operatorname{rank}\begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix} = r + s$.

Proof.

Just note that $A \sim N_r$ and $B \sim N_s$ so that

$$\begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix} \sim \begin{pmatrix} N_r & 0 \\ 0 & N_s \end{pmatrix},$$

where $P = \begin{pmatrix} P_r & 0 \\ 0 & P_s \end{pmatrix}$ and $Q = \begin{pmatrix} Q_r & 0 \\ 0 & Q_s \end{pmatrix}$. □

Theorem

Let A *and* B *be* $n \times n$ *matrices. Then*

1. $A \sim B \iff \text{rank}(A) = \text{rank}(B)$,
2. $A \overset{row}{\sim} B \iff E_A = E_B$,
3. $A \overset{col}{\sim} B \iff E_{A^T} = E_{B^T}$,

so that multiplication by a nonsingular matrix does not change rank.

Proof of (1).

"$\Longrightarrow$": Assume $A \sim B$ with $\text{rank}(A) = r$, $\text{rank}(B) = s$. Then

$$N_r \sim A \sim B \sim N_s \quad \text{so that} \quad N_r \sim N_s \text{ and } r = s.$$

"$\Longleftarrow$": Assume $\text{rank}(A) = \text{rank}(B) = r$. Then

$$A \sim N_r \text{ and } B \sim N_r \quad \text{so that} \quad A \sim N_r \sim B.$$

□

Proof of (2).

"$\Longrightarrow$": Assume $A \overset{\text{row}}{\sim} B$. We know

$$A \overset{\text{row}}{\sim} E_A \quad \text{so that} \quad B \overset{\text{row}}{\sim} A \overset{\text{row}}{\sim} E_A.$$

However, we also have $B \overset{\text{row}}{\sim} E_B$ and uniqueness of the row echelon form gives us $E_A = E_B$.

"$\Longleftarrow$": Assume $E_A = E_B$. Then

$$A \overset{\text{row}}{\sim} E_A = E_B \overset{\text{row}}{\sim} B.$$

□

Proof of (3).

This follows from (2) using the transpose since

$$A \overset{\text{col}}{\sim} B \iff AQ = B \iff (AQ)^T = B^T$$
$$\iff Q^T A^T = B^T \iff A^T \overset{\text{row}}{\sim} B^T.$$

□

Theorem (Row-rank = column rank = rank)

For any $m \times n$ matrix A *we have* $\operatorname{rank}(\mathsf{A}) = \operatorname{rank}(\mathsf{A}^T)$.

Proof.

Let $\operatorname{rank}(\mathsf{A}) = r$ and P, Q nonsingular such that

$$\mathsf{PAQ} = \mathsf{N}_r = \begin{pmatrix} \mathsf{I}_r & \mathsf{0}_{r\times n-r} \\ \mathsf{0}_{m-r\times r} & \mathsf{0}_{m-r\times n-r} \end{pmatrix}.$$

Then

$$(\mathsf{PAQ})^T = \mathsf{N}_r^T \quad \Longleftrightarrow \quad \mathsf{Q}^T\mathsf{A}^T\mathsf{P}^T = \mathsf{N}_r^T$$

so that $\mathsf{A}^T \sim \mathsf{N}_r^T$.

Finally,

$$\operatorname{rank}(\mathsf{A}^T) = \operatorname{rank}(\mathsf{N}_r^T) = \begin{pmatrix} \mathsf{I}_r & \mathsf{0}_{r\times m-r} \\ \mathsf{0}_{n-r\times r} & \mathsf{0}_{n-r\times m-r} \end{pmatrix} = r.$$

□

LU Factorization/Decomposition

Recall our earlier example with the matrix

$$A = \begin{pmatrix} 2 & 2 & 6 \\ 2 & 1 & 7 \\ 2 & -6 & -7 \end{pmatrix}$$

Gaussian elimination (with the multipliers as below) leads to

$$\underbrace{\underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -8 & 1 \end{pmatrix}}_{E_3} \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}}_{=E_2} \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{=E_1}}_{= \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 7 & -8 & 1 \end{pmatrix}} A = \underbrace{\begin{pmatrix} 2 & 2 & 6 \\ 0 & -1 & 1 \\ 0 & 0 & -21 \end{pmatrix}}_{=U}$$

lower triangular

We would, however, like a factorization of the form $A = LU$.

What we need is the inverse of the lower triangular matrix $\begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 7 & -8 & 1 \end{pmatrix}$, i.e.,

$$\begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 7 & -8 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 8 & 1 \end{pmatrix} = \mathsf{L}.$$

Note that the entries below the diagonal in L correspond to the negatives of the multipliers in $\mathsf{E}_1, \mathsf{E}_2, \mathsf{E}_3$.

If we remember that the inverse of a (lower) triangular matrix is (lower) triangular then we can be optimistic about this approach working in general.

General Discussion

Consider the $n \times n$ lower-triangular elementary matrix

$$\mathsf{T}_k = \mathsf{I} - \boldsymbol{c}_k \boldsymbol{e}_k^T,$$

where

$$\boldsymbol{c}_k = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \mu_{k+1} \\ \vdots \\ \mu_n \end{pmatrix}, \quad \textit{i.e.}, \qquad \boldsymbol{c}_k \boldsymbol{e}_k^T = \begin{pmatrix} 0 & \cdots & 0 & \boldsymbol{c}_k & 0 & \cdots & 0 \end{pmatrix}_{n \times n}$$

or

$$\mathsf{T}_k = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & 0 & \\ & & 1 & & & \\ & & -\mu_{k+1} & 1 & & \\ & 0 & \vdots & & \ddots & \\ & & -\mu_n & & & 1 \end{pmatrix}.$$

To compute the inverse of T_k we use the Sherman–Morrison formula:

$$\mathsf{T}_k^{-1} = \left(\mathsf{I} - \boldsymbol{c}_k \boldsymbol{e}_k^T\right)^{-1}$$
$$= \mathsf{I} - \frac{\boldsymbol{c}_k \boldsymbol{e}_k^T}{\boldsymbol{e}_k^T \boldsymbol{c}_k - 1}$$

This simplifies because $\boldsymbol{e}_k^T \boldsymbol{c}_k = 0$.

Thus,

$$\mathsf{T}_k^{-1} = \mathsf{I} + \boldsymbol{c}_k \boldsymbol{e}_k^T$$

and we see that we always get the negatives of the multipliers $\mu_{k+1}, \ldots, \mu_n$ below the diagonal in the k^{th} column of T_k.

We now consider what happens during the k^{th} step of Gaussian elimination, i.e., we start with

$$A_{k-1} = \begin{pmatrix} \star & \star & \cdots & \alpha_1 & \star & \cdots & \star \\ 0 & \star & & \alpha_2 & & & \\ \vdots & \ddots & \ddots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 0 & \alpha_k & \star & \cdots & \star \\ 0 & \cdots & 0 & \alpha_{k+1} & \star & \cdots & \star \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \alpha_n & \star & \cdots & \star \end{pmatrix}$$

and take the vector of multipliers to be

$$\boldsymbol{c}_k = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \alpha_{k+1}/\alpha_k \\ \vdots \\ \alpha_n/\alpha_k \end{pmatrix}$$

The next stage of Gaussian elimination produces

$$\begin{aligned}
\mathsf{A}_k &= \mathsf{T}_k \mathsf{A}_{k-1} = \left(\mathsf{I} - \boldsymbol{c}_k \boldsymbol{e}_k^T\right) \mathsf{A}_{k-1} \\
&= \mathsf{A}_{k-1} - \boldsymbol{c}_k \underbrace{\boldsymbol{e}_k^T \mathsf{A}_{k-1}}_{=(\mathsf{A}_{k-1})_{k*}} \\
&= \mathsf{A}_{k-1} - \begin{pmatrix} 0 & \cdots & 0 & \alpha_k \boldsymbol{c}_k & \star & \cdots & \star \end{pmatrix}_{n \times n} \\
&= \begin{pmatrix}
\star & \star & \cdots & \alpha_1 & \star & \cdots & \star \\
0 & \star & & \alpha_2 & & & \\
\vdots & \ddots & \ddots & \vdots & \vdots & & \vdots \\
0 & \cdots & 0 & \alpha_k & \star & \cdots & \star \\
0 & \cdots & 0 & 0 & \star & \cdots & \star \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & 0 & \star & \cdots & \star
\end{pmatrix}
\end{aligned}$$

If we assume that everything is nice enough so that no row interchanges are required, then we end up with

$$\mathsf{T}_{n-1} \cdots \mathsf{T}_k \cdots \mathsf{T}_2 \mathsf{T}_1 \mathsf{A} = \mathsf{U}$$

or

$$\mathsf{A} = \mathsf{T}_1^{-1} \mathsf{T}_2^{-1} \cdots \mathsf{T}_k^{-1} \cdots \mathsf{T}_{n-1}^{-1} \mathsf{U}.$$

From above we remember that

$$\mathsf{T}_k^{-1} = \mathsf{I} + \boldsymbol{c}_k \boldsymbol{e}_k^T$$

Therefore, using $\mathsf{T}_k^{-1} = \mathsf{I} + \boldsymbol{c}_k \boldsymbol{e}_k^T$, we have

$$\mathsf{T}_1^{-1}\mathsf{T}_2^{-1}\cdots\mathsf{T}_{n-1}^{-1} = \left(\mathsf{I} + \boldsymbol{c}_1\boldsymbol{e}_1^T\right)\left(\mathsf{I} + \boldsymbol{c}_2\boldsymbol{e}_2^T\right)\cdots\left(\mathsf{I} + \boldsymbol{c}_{n-1}\boldsymbol{e}_{n-1}^T\right)$$
$$= \left(\mathsf{I} + \boldsymbol{c}_1\boldsymbol{e}_1^T + \boldsymbol{c}_2\boldsymbol{e}_2^T + \boldsymbol{c}_1 \underbrace{\boldsymbol{e}_1^T\boldsymbol{c}_2}_{=0} \boldsymbol{e}_2^T\right)\cdots\left(\mathsf{I} + \boldsymbol{c}_{n-1}\boldsymbol{e}_{n-1}^T\right)$$

and since in general $\boldsymbol{e}_j^T\boldsymbol{c}_k = 0$ whenever $j \le k$ this yields

$$\mathsf{T}_1^{-1}\mathsf{T}_2^{-1}\cdots\mathsf{T}_{n-1}^{-1} = \mathsf{I} + \boldsymbol{c}_1\boldsymbol{e}_1^T + \boldsymbol{c}_2\boldsymbol{e}_2^T + \ldots + \boldsymbol{c}_{n-1}\boldsymbol{e}_{n-1}^T$$

where

$$\boldsymbol{c}_k\boldsymbol{e}_k^T = \begin{pmatrix} 0 & \cdots & 0 & \boldsymbol{c}_k & 0 & \cdots & 0 \end{pmatrix}_{n\times n} = \begin{pmatrix} & & & 0 & & & \\ & & & \vdots & & & \\ 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ & & & \alpha_{k+1}/\alpha_k & & & \\ & & & \vdots & & & \\ & & & \alpha_n/\alpha_k & & & \end{pmatrix}$$

Finally,

$$\mathsf{T}_1^{-1}\mathsf{T}_2^{-1}\cdots\mathsf{T}_{n-1}^{-1} = \mathsf{I} + \boldsymbol{c}_1\boldsymbol{e}_1^T + \boldsymbol{c}_2\boldsymbol{e}_2^T + \ldots + \boldsymbol{c}_{n-1}\boldsymbol{e}_{n-1}^T$$

$$= \begin{pmatrix} 1 & & & & \\ \ell_{2,1} & \ddots & & 0 & \\ & \ell_{3,2} & 1 & & \\ \vdots & \vdots & \ddots & \ddots & \\ \ell_{n,1} & \ell_{n,2} & \cdots & \ell_{n,n+1} & 1 \end{pmatrix} = \mathsf{L}$$

with

$$\ell_{i,k} = \alpha_i/\alpha_k, \quad i = k+1, \ldots, n,$$

due to the form of $\boldsymbol{c}_k\boldsymbol{e}_k^T$.

Remark

1. *The LU factorization obtained in this way is unique.*
2. *By not keeping track of the (known) 1s on the diagonal of* L *we can store — on a computer — the entries of both* L *and* U *in the space previously allocated for* A. *Thus, no additional memory is required.*

How to solve linear systems using the LU factorization

Consider the linear system

$$\mathsf{A}\boldsymbol{x} = \boldsymbol{b}.$$

To solve it we first compute the factorization $\mathsf{A} = \mathsf{LU}$, so that

$$\mathsf{A}\boldsymbol{x} = \boldsymbol{b} \quad \Longleftrightarrow \quad \mathsf{LU}\boldsymbol{x} = \boldsymbol{b}.$$

Now we

1. let $\boldsymbol{y} = \mathsf{U}\boldsymbol{x}$ and solve $\mathsf{L}\boldsymbol{y} = \boldsymbol{b}$ (easy and cheap since it is a lower-triangular system $\longrightarrow$ forward substitution);
2. solve $\mathsf{U}\boldsymbol{x} = \boldsymbol{y}$ (again easy and cheap since it is an upper-triangular system $\longrightarrow$ back substitution).

Solving multiple linear systems with the same A

The LU factorization is particular useful if only the right-hand side changes — but not the matrix A.

This is the case in data fitting when the measurements change, but not the basic model (i.e., the basis functions that are used and — if the basis depends on the measurement locations — the measurement locations).

The linear system can then be thought of as

$$AX = B,$$

and we compute the LU factorization of A only once, and then obtain each column of X by the forward-back substitution procedure above from the corresponding column in B.

This forward-back substitution procedure is embarrassingly parallel.

Remark

The multiple right-hand side approach is also the practical and efficient way to compute A^{-1} *— should we really have the need for this matrix.*

Namely, we solve

$$A\boldsymbol{x}_j = \boldsymbol{e}_j, \qquad j = 1, \ldots, n.$$

Since $\boldsymbol{e}_j$ *is the* j^{th} *column of* I *this implies that* $\boldsymbol{x}_j$ *is the* j^{th} *column of* A^{-1}*, i.e.,*

$$\boldsymbol{x}_j = (A^{-1})_{:j} \quad \Longleftrightarrow \quad X = A^{-1}.$$

Major limitation of the basic LU factorization

So far we have assumed that Gaussian elimination does not require any row interchanges. This assumption is, of course, in general not realistic.

Even for a nonsingular matrix A, LU factorization will fail due to a division by zero error if we encounter a zero pivot during Gaussian elimination. This is not something that can immediately be predicted by looking at A.

How do we overcome this problem?

We look for a row (below the current pivot row) to swap places with so that there no longer is a zero pivot.

How do we do this in our matrix formulation?

We multiply (from the left) by an appropriate permutation matrix.

Partial Pivoting

Since the choice of pivot is not unique we declare that we always pick that row that produces the largest pivot.

Example

Consider

$$A = \begin{pmatrix} 2 & 4 & 6 & -2 \\ 1 & 2 & 1 & 2 \\ 0 & 2 & 4 & 2 \\ -2 & 1 & 0 & 10 \end{pmatrix}$$

and use a permutation counter.

Permutation counter in rightmost column, multipliers in blue.

$$\left(\begin{array}{cccc|c} 2 & 4 & 6 & -2 & 1 \\ 1 & 2 & 1 & 2 & 2 \\ 0 & 2 & 4 & 2 & 3 \\ -2 & 1 & 0 & 10 & 4 \end{array}\right) \longrightarrow \left(\begin{array}{cccc|c} 2 & 4 & 6 & -2 & 1 \\ \frac{1}{2} & 0 & -2 & 3 & 2 \\ 0 & 2 & 4 & 2 & 3 \\ -1 & 5 & 6 & 8 & 4 \end{array}\right)$$

$$\longrightarrow \left(\begin{array}{cccc|c} 2 & 4 & 6 & -2 & 1 \\ -1 & 5 & 6 & 8 & 4 \\ 0 & 2 & 4 & 2 & 3 \\ \frac{1}{2} & 0 & -2 & 3 & 2 \end{array}\right) \longrightarrow \left(\begin{array}{cccc|c} 2 & 4 & 6 & -2 & 1 \\ -1 & 5 & 6 & 8 & 4 \\ 0 & \frac{2}{5} & \frac{8}{5} & -\frac{6}{5} & 3 \\ \frac{1}{2} & 0 & -2 & 3 & 2 \end{array}\right)$$

$$\longrightarrow \left(\begin{array}{cccc|c} 2 & 4 & 6 & -2 & 1 \\ -1 & 5 & 6 & 8 & 4 \\ \frac{1}{2} & 0 & -2 & 3 & 2 \\ 0 & \frac{2}{5} & \frac{8}{5} & -\frac{6}{5} & 3 \end{array}\right) \longrightarrow \left(\begin{array}{cccc|c} 2 & 4 & 6 & -2 & 1 \\ -1 & 5 & 6 & 8 & 4 \\ \frac{1}{2} & 0 & -2 & 3 & 2 \\ 0 & \frac{2}{5} & -\frac{4}{5} & \frac{6}{5} & 3 \end{array}\right)$$

Therefore, we end up with the pivoted LU factorization

$$PA = LU,$$

where

$$L = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ \frac{1}{2} & 0 & 1 & 0 \\ 0 & \frac{2}{5} & -\frac{4}{5} & 1 \end{pmatrix}, \quad U = \begin{pmatrix} 2 & 4 & 6 & -2 \\ 0 & 5 & 6 & 8 \\ 0 & 0 & -2 & 3 \\ 0 & 0 & 0 & \frac{6}{5} \end{pmatrix}, \quad P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Remark

The messy details of the general derivation can be found in the book.

Now we are ready to solve any nonsingular system $A\boldsymbol{x} = \boldsymbol{b}$.

We just perform LU factorization with partial pivoting.

Since $PA = LU$ we get

$$A\boldsymbol{x} = \boldsymbol{b} \iff PA\boldsymbol{x} = P\boldsymbol{b} \iff \boxed{LU\boldsymbol{x} = P\boldsymbol{b}}$$

Therefore, we can use exactly the same two-step procedure as before, but we must permute the right-hand side first.

Example

Solve $A\boldsymbol{x} = \boldsymbol{b}$, where

$$A = \begin{pmatrix} 2 & 4 & 6 & -2 \\ 1 & 2 & 1 & 2 \\ 0 & 2 & 4 & 2 \\ -2 & 1 & 0 & 10 \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 10 \end{pmatrix}.$$

We computed the pivoted LU factorization of A above and obtained a permutation vector $\boldsymbol{p} = \begin{pmatrix} 1 \\ 4 \\ 2 \\ 3 \end{pmatrix}$, so that $P\boldsymbol{b} = \begin{pmatrix} 0 \\ 10 \\ 1 \\ 2 \end{pmatrix}$.

Now we just need to solve

$$L \underbrace{U\boldsymbol{x}}_{=\boldsymbol{y}} = P\boldsymbol{b}.$$

Step 1: Solve $L\boldsymbol{y} = P\boldsymbol{b}$ (using augmented matrix notation)

$$\left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 10 \\ \frac{1}{2} & 0 & 1 & 0 & 1 \\ 0 & \frac{2}{5} & -\frac{4}{5} & 1 & 2 \end{array}\right) \implies \boldsymbol{y} = \begin{pmatrix} 0 \\ 10 \\ 1 \\ -\frac{6}{5} \end{pmatrix}$$

Step 2: Solve $U\boldsymbol{x} = \boldsymbol{y}$ (using augmented matrix notation)

$$\left(\begin{array}{cccc|c} 2 & 4 & 6 & -2 & 0 \\ 0 & 5 & 6 & 8 & 10 \\ 0 & 0 & -2 & 3 & 1 \\ 0 & 0 & 0 & \frac{6}{5} & -\frac{6}{5} \end{array}\right) \implies \boldsymbol{x} = \begin{pmatrix} 7 \\ 6 \\ -2 \\ -1 \end{pmatrix}$$

LU Factorization for Symmetric Matrices

We begin by creating a more symmetric version of the basic LU factorization for an arbitrary nonsingular $n \times n$ matrix A.

The trick is to factor out the diagonal of U, i.e.,

$$\mathsf{A} = \mathsf{LU} \quad \Longrightarrow \quad \mathsf{A} = \mathsf{LD}\widetilde{\mathsf{U}}$$

with

$$\mathsf{U} = \mathsf{D}\widetilde{\mathsf{U}} \quad \Longleftrightarrow$$

$$\begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & & \\ \vdots & \ddots & \ddots & \\ 0 & \cdots & 0 & u_{nn} \end{pmatrix} = \begin{pmatrix} u_{11} & & & \\ & u_{22} & & \\ & & \ddots & \\ & & & u_{nn} \end{pmatrix} \begin{pmatrix} 1 & \frac{u_{12}}{u_{11}} & \cdots & \frac{u_{1n}}{u_{11}} \\ 0 & 1 & \frac{u_{23}}{u_{22}} & \\ \vdots & \ddots & \ddots & \\ 0 & \cdots & 0 & 1 \end{pmatrix}.$$

Cholesky (see [BT14]) Factorization

If A is a symmetric matrix, then the LU factorization must be symmetric as well, i.e., $L = \widetilde{U}^T$, so that

$$A = \widetilde{U}^T D \widetilde{U}.$$

Moreover, if the entries of D are all positive (so that we can take square roots), then we can split $D = \sqrt{D}\sqrt{D}$ with

$$\sqrt{D} = \begin{pmatrix} \sqrt{u_{11}} & & & \\ & \sqrt{u_{22}} & & \\ & & \ddots & \\ & & & \sqrt{u_{nn}} \end{pmatrix}$$

This results in the Cholesky factorization of A

$$A = R^T R, \quad \text{with} \quad R = \sqrt{D}\widetilde{U},$$

where R is upper-triangular.

Definition

A symmetric (nonsingular) matrix A whose LU factorization has only positive pivot elements is called positive definite.

Theorem

A matrix A *is positive definite if and only if it has a unique Cholesky factorization* $A = R^T R$ *with* R *and upper-triangular matrix with positive diagonal entries*

Proof.

The implication

$$\mathsf{A} \text{ positive definite} \Longrightarrow \mathsf{A} = \mathsf{R}^T\mathsf{R}$$

follows from the discussion above.

"$\Longleftarrow$": Assume $\mathsf{A} = \mathsf{R}^T\mathsf{R}$ with $r_{ii} > 0$.

Factoring out r_{ii} produces

$$\mathsf{R} = \mathsf{D}\mathsf{U}, \quad \mathsf{D} = \operatorname{diag}(r_{11}, \ldots, r_{nn}).$$

So

$$\mathsf{A} = (\mathsf{D}\mathsf{U})^T\mathsf{D}\mathsf{U} = \mathsf{U}^T\mathsf{D}^2\mathsf{U} = \mathsf{L}\mathsf{D}^2\mathsf{L}^T$$

and we have an LU factorization with positive pivots.

Uniqueness follows from the uniqueness of the LU factorization. □

References I

[BT14] Claude Brezinski and Dominique Tournès, *André Louis Cholesky – Mathematician, Topographer and Army Officer*, Birkhäuser, 2014.

[CW90] Don Coppersmith and Shmuel Winograd, *Matrix multiplication via arithmetic progressions*, J. Symbol. Comput. **9** (1990), 251–280.

[DS13] A.M. Davie and A.J. Stothers, *Improved bound for complexity of matrix multiplication*, Proceedings of the Royal Society of Edinburgh **143A** (2013), 351–370.

[HS81] H. V. Henderson and S. R. Searle, *On deriving the inverse of a sum of matrices*, SIAM Review **23** (1981), 53–60.

[LG14] François Le Gall, *Powers of tensors and fast matrix multiplication*, Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation (ISSAC 2014), 2014.

[Mey00] Carl D. Meyer, *Matrix Analysis and Applied Linear Algebra*, SIAM, Philadelphia, PA, 2000.

References II

[Mol08] C. B. Moler, *Numerical Computing with MATLAB*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2008.

[NinBC] *Jiu Zhang Suan Shu — The Nine Chapters on the Mathematical Art*, China, approx. 200 BC.

[Str69] Volker Strassen, *Gaussian elimination is not optimal*, Numer. Math. **13** (1969), 354–356.

[Wil14] Virginia Williams, *Breaking the Coppersmith–Winograd barrier*, 2014.

[Yua12] Ya-xiang Yuan, *Jiu Zhang Suan Shu and the Gauss algorithm for linear equations*, Documenta Mathematica, Optimization Stories, Extra Volume ISMP, DMV, Bielefeld, 2012, pp. 9–14.