

Math 2270-1

Notes of 09/18/19

Matrix Factorizations

- A **factorization** of a matrix A is obtained by expressing A as a product of several (usually 2 or 3) factors.
- There are very many widely used matrix factorizations and in this class we will encounter just a few of them.
- To begin with, suppose A is a (square) invertible matrix and you want to solve the linear system

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

for several right hand sides

$$\mathbf{b} = \mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_p,$$

giving solutions

$$\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_p,$$

where

$$A\mathbf{x}_i = \mathbf{b}_i, \quad i = 1, \dots, p.$$



For the moment, and for simplicity, let us assume that the pivots are all non-zero, i.e.,

we do not need to apply row interchanges.
We'll discuss pivoting later.

- We know several ways of solving those linear systems.
1. Solve each system independently and from scratch. That is obviously wasteful!
 2. Augment the matrix A with all right hand sides, to get the augmented matrix

$$[A \quad \mathbf{b}_1 \quad \mathbf{b}_2 \quad \dots \quad \mathbf{b}_p]$$

and reduce to the reduced row echelon form

$$[I \quad \mathbf{x}_1 \quad \mathbf{x}_2 \quad \dots \quad \mathbf{x}_p]$$

But note that we need to know all right hand sides before solving any of the linear systems. In applications we often get a new right hand side only after solving the most recent of those systems.

3. Compute A^{-1} and then get

$$\mathbf{x}_i = A^{-1}\mathbf{b}_i$$

when we need it.

4. Apply row operations to reduce A to (unreduced) row echelon form and keep track of the multipliers. Then, using the stored multipliers, process each right hand side, and use back substitution.



where do we store the multipliers?

- Our focus today is on an alternative approach. Factor A as

$$A = LU \quad (1)$$

where

L is **unit lower triangular**, i.e.,

$$l_{ii} = 1, \quad i = 1, 2, \dots, n \quad \text{and} \quad j < i \quad \implies \quad a_{ij} = 0$$

U is **upper triangular**, i.e.,

$$l_{ii} = 1, \quad i = 1, 2, \dots, n \quad \text{and} \quad j > i \quad \implies \quad a_{ij} = 0$$

- In other words, denoting possibly non-zero entries by x , L and U are of the form:

$$L = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ x & 1 & 0 & \dots & 0 & 0 \\ x & x & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ x & x & x & \dots & 1 & 0 \\ x & x & x & \dots & x & 1 \end{bmatrix}$$

and

$$U = \begin{bmatrix} x & x & x & \dots & x & x \\ 0 & x & x & \dots & x & x \\ 0 & 0 & x & \dots & x & x \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & x & x \\ 0 & 0 & 0 & \dots & 0 & x \end{bmatrix}$$

- Given the matrix factorization (1) we solve the system $Ax = LUx = b$ in two steps:

1. solve: $Lz = b$ Forward Substitution
2. solve: $Ux = z$ Backward Substitution

- Example: Let

$$A = \begin{bmatrix} 1 & -1 & 1 \\ 2 & 0 & 1 \\ 4 & 2 & 4 \end{bmatrix}, L = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 4 & 3 & 1 \end{bmatrix}, \quad \text{and} \quad U = \begin{bmatrix} 1 & -1 & 1 \\ 0 & 2 & -1 \\ 0 & 0 & 3 \end{bmatrix}$$

- Then $A = LU$:

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 4 & 3 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1 & 1 \\ 0 & 2 & -1 \\ 0 & 0 & 3 \end{bmatrix} = U$$

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

- moreover, let

$$\mathbf{x} = \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}$$

- Then

$$A\mathbf{x} = \begin{bmatrix} 4 \\ 3 \\ 4 \end{bmatrix}$$

- Find \mathbf{x} by using the LU factorization.

- OK, and how would we have done it the old way?



Actually, **Reduction to r.e.f. computes the LU factorization!**

- the two procedures are equivalent!
- The standard argument to show this is a mess that involves “elementary matrices” and their inverses, and generous use of groups of three dots . . .
- See textbook for an argument along those lines.
- However, Gil Strang of MIT came up with a beautifully simple argument.
- Consider the evolution of the working array during Gaussian Elimination, for a 4×4 system.
- The letter x denotes an entry in the working array. The symbol \otimes denotes an entry that is final and no longer changed in the process. The letter m denotes the multipliers, stored in the lower part of the working array.

- We get

$$\begin{aligned}
 \begin{bmatrix} \otimes & \otimes & \otimes & \otimes \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \end{bmatrix} &\longrightarrow \begin{bmatrix} \otimes & \otimes & \otimes & \otimes \\ m_{21} & \otimes & \otimes & \otimes \\ m_{31} & x & x & x \\ m_{41} & x & x & x \end{bmatrix} \\
 &\longrightarrow \begin{bmatrix} \otimes & \otimes & \otimes & \otimes \\ m_{21} & \otimes & \otimes & \otimes \\ m_{31} & m_{32} & \otimes & \otimes \\ m_{41} & m_{42} & x & x \end{bmatrix} \\
 &\longrightarrow \begin{bmatrix} \otimes & \otimes & \otimes & \otimes \\ m_{21} & \otimes & \otimes & \otimes \\ m_{31} & m_{32} & \otimes & \otimes \\ m_{41} & m_{42} & m_{43} & \otimes \end{bmatrix}
 \end{aligned}$$

- Let's denote row i of a matrix A by $r_i(A)$ and consider in particular the third row of the working array. We get

$$r_3(U) = r_3(A) - m_{31}r_1(U) - m_{32}r_2(U)$$

- This can be rewritten as

$$r_3(A) = m_{31}r_1(U) + m_{32}r_2(U) + r_3(U)$$



That last equation is exactly what we get in the matrix multiplication $A = LU$!

$$\begin{array}{c}
\begin{bmatrix} 1 & 0 & 0 & 0 \\ m_{21} & 1 & 0 & 0 \\ m_{31} & m_{32} & 1 & 0 \\ m_{41} & m_{42} & m_{43} & 1 \end{bmatrix} \\
= L
\end{array}
\begin{array}{c}
\begin{bmatrix} r_1(U) \\ r_2(U) \\ r_3(U) \\ r_4(U) \end{bmatrix} \\
\begin{bmatrix} r_1(A) \\ r_2(A) \\ r_3(A) \\ r_4(A) \end{bmatrix}
\end{array}
\begin{array}{c}
= U \\
= A
\end{array}$$

- Clearly, this applies in general, not just to the third row of a 4×4 matrix!
- $A = LU$, L is unit lower triangular, U is upper triangular.

On Computing the Inverse



It usually is a bad idea to compute an inverse matrix, for these reasons:

- 0.** You don't need an explicit expression for the inverse to multiply with it, since $A^{-1}\mathbf{b}$ is the solution of $A\mathbf{x} = \mathbf{b}$. So instead of multiplying with A^{-1} you can just solve a linear system!
- 1.** Computing A^{-1} requires more (actually three times as much for large values of n) computational work than computing the LU factorization.
- 2.** A major problem in numerical procedures is the propagation of round-off errors, since numbers can be expressed only approximately on a computer. The inverse of a matrix is affected more than the LU factorization by such errors.
- 3.** Most importantly, but also most subtly, matrix inversion destroys sparsity, whereas the LU factorization has a better chance to preserve it. A matrix is sparse if most of its entries are zero.

Factorization and Block Matrices

- Let's combine the last two topics, partitioned matrices, and matrix factorizations.
- Suppose

$$A = \begin{matrix} & n-1 & 1 \\ n-1 & \hat{A} & \mathbf{c} \\ 1 & \mathbf{r}^T & a_{nn} \end{matrix}$$

and suppose we have computed the LU factorization

$$\hat{A} = \hat{L}\hat{U}$$

of \hat{A} . Without starting from scratch, can you compute the LU factorization of A ?

- Try

$$L = \begin{bmatrix} \hat{L} & 0 \\ \hat{\mathbf{y}}^T & 1 \end{bmatrix} \quad \text{and} \quad U = \begin{bmatrix} \hat{U} & 0 \\ \mathbf{x} & \xi \end{bmatrix}.$$

How do you pick $\mathbf{x} \in \mathbb{R}^{n-1}$, $\mathbf{y} \in \mathbb{R}^{n-1}$, and $\xi \in \mathbb{R}$? What can go wrong?

- **Example and Exercise:** The $n \times n$ matrix

$$A = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix} \quad (2)$$

occurs frequently in the solution of second order ordinary differential equations.

- a. Show that

$$A^{-1} = [x_{ij}] \quad (3)$$

where

$$x_{ij} = \begin{cases} \frac{(n+1-j)i}{n+1} & i = 1, 2, \dots, j \\ \frac{(n+1-i)j}{n+1} & i = j+1, \dots, n. \end{cases} \quad (4)$$

- b. Show that

$$A = LU$$

where $L = [l_{ij}]$ is unit lower triangular and $U = [u_{ij}]$ is upper triangular and

$$l_{ii} = 1, \quad l_{i,i-1} = -\frac{i-1}{i}, \quad u_{ii} = \frac{i+1}{i}, \quad \text{and} \quad u_{i,i+1} = -1. \quad (5)$$

- c. Discuss the differences between solving $Ax = b$ by multiplying with the inverse, and by applying backward and forward substitution to the LU factorization.

- Example. In the case $n = 10$ we get

$$A^{-1} = \frac{1}{11} \begin{bmatrix} 10 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \\ 9 & 18 & 16 & 14 & 12 & 10 & 8 & 6 & 4 & 2 \\ 8 & 16 & 24 & 21 & 18 & 15 & 12 & 9 & 6 & 3 \\ 7 & 14 & 21 & 28 & 24 & 20 & 16 & 12 & 8 & 4 \\ 6 & 12 & 18 & 24 & 30 & 25 & 20 & 15 & 10 & 5 \\ 5 & 10 & 15 & 20 & 25 & 30 & 24 & 18 & 12 & 6 \\ 4 & 8 & 12 & 16 & 20 & 24 & 28 & 21 & 14 & 7 \\ 3 & 6 & 9 & 12 & 15 & 18 & 21 & 24 & 16 & 8 \\ 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 & 9 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{bmatrix}$$

