Elementary Row Operations

- Consider the matrix

• Engineers intrested in $A \cdot \tilde{x}$

• mathmaticians intersted in A

$$E = \begin{bmatrix} \mathbf{e}_1^T \\ \alpha\,\mathbf{e}_1^T + \mathbf{e}_2^T \\ \mathbf{e}_3^T \\ \vdots \\ \mathbf{e}_n^T \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ \alpha & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}.$$

- E is the result of the elementary row operation $R_2 \to R_2 + \alpha\,R_1$ on the identity matrix.

Elementary Row Operations

- Note that

$$E\,A = \begin{bmatrix} \mathbf{e}_1^T \\ \alpha\,\mathbf{e}_1^T + \mathbf{e}_2^T \\ \mathbf{e}_3^T \\ \vdots \\ \mathbf{e}_n^T \end{bmatrix} A = \begin{bmatrix} \mathbf{e}_1^T\,A \\ (\alpha\,\mathbf{e}_1^T + \mathbf{e}_2^T)\,A \\ \mathbf{e}_3^T\,A \\ \vdots \\ \mathbf{e}_n^T\,A \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1^T \\ \alpha\,\mathbf{A}_1^T + \mathbf{A}_2^T \\ \mathbf{A}_3^T \\ \vdots \\ \mathbf{A}_n^T \end{bmatrix}$$

- Therefore $E\,A$ is the result of performing the elementary row operation $R_2 \to R_2 + \alpha\,R_1$ on A.
- Any other elementary row operation of this type $R_i \to R_i + \alpha\,R_j$ is obtained by replacing row i in the identity matrix by $\alpha\,\mathbf{e}_j^T + \mathbf{e}_i^T$. We will call this matrix $E_{ij}(\alpha)$.

Elementary Row Operations

- Therefore

$$E_{ij}(\alpha) = \begin{bmatrix} \mathbf{e}_1^T \\ \mathbf{e}_2^T \\ \vdots \\ \alpha\,\mathbf{e}_j^T + \mathbf{e}_i^T \\ \vdots \\ \mathbf{e}_n^T \end{bmatrix}.$$

  This matrix has 1's along the main diagonal and α in the (j, i) position. Note that for Gauss elimination only row operations with $j < i$ are used and so $E_{ij}(\alpha)$ is a **lower triangular matrix**.
- In order to "undo" this row operation we must add $-\alpha R_j$ to R_i; that is apply $E_{ij}(-\alpha)$. Therefore $E_{ij}(\alpha)$ is invertible with

$$E_{ij}^{-1}(\alpha) = E_{ij}(-\alpha).$$

Elementary Row Operations

- Interchanging rows i and j is accomplished by

$$E_{i\leftrightarrow j} = \begin{bmatrix} \mathbf{e}_1^T \\ \vdots \\ \mathbf{e}_{i-1}^T \\ \mathbf{e}_j^T \\ \mathbf{e}_{i+1}^T \\ \vdots \\ \mathbf{e}_{j-1}^T \\ \mathbf{e}_i^T \\ \mathbf{e}_{j+1}^T \\ \vdots \\ \mathbf{e}_n^T \end{bmatrix}.$$

  Clearly we have

$$E_{i\leftrightarrow j}^{-1} = E_{i\leftrightarrow j}.$$

Elementary Row Operations

- Finally multiplying a row by scalar is given by

$$E_i(\beta) = \begin{bmatrix} \mathbf{e}_1^T \\ \mathbf{e}_2^T \\ \vdots \\ \beta\, \mathbf{e}_i^T \\ \vdots \\ \mathbf{e}_n^T \end{bmatrix}.$$

Note that

$$E_i^{-1}(\beta) = E_i(\tfrac{1}{\beta}).$$

An Aside: Elementary Column Operations

- In the discussion on pivoting, it was mentioned that complete pivoting required column swaps. We can extend the concept of elementary row operations to column operations. However these operations **change the variables** in the associated system of linear equations.
- Recall

$$A\mathbf{x} = x_1\,\mathbf{a}_1 + x_2\,\mathbf{a}_2 + \cdots + x_n\,\mathbf{a}_n.$$

- A column swap

$$C_i \leftrightarrow C_j$$

will interchange the variables x_i and x_j in the associated linear system.
- Let $\tilde{A}$ be obtained from A by multiplying a column by a non-zero constant

$$C_i \to \beta\, C_i.$$

We have

$$\tilde{A}\,\tilde{\mathbf{x}} = \tilde{x}_1\,\mathbf{a}_1 + \cdots + \tilde{x}_i\,\beta\mathbf{a}_i + \cdots + \tilde{x}_n\,\mathbf{a}_n.$$

$\beta\tilde{x}_i = x_i$

This is equivalent to the original system $A\mathbf{x} = \mathbf{b}$ if we identify $\beta\tilde{x}_i = \cancel{\beta}\, x_i$ (with $\tilde{x}_k = x_k$ for all $k \neq i$). Thus this column operation **scales** the variable x_i; $x_i \to \frac{1}{\beta}\, x_i$.

An Aside: Elementary Column Operations

- Let $\tilde{A}$ be obtained from A by adding a multiple of one column to another

$$C_i \to C_i + \alpha\, C_j.$$

We have

$$\tilde{A}\,\tilde{\mathbf{x}} = \tilde{x}_1\,\mathbf{a}_1 + \cdots + \tilde{x}_i\,(\mathbf{a}_i + \alpha\,\mathbf{a}_j) + \cdots + \tilde{x}_n\,\mathbf{a}_n.$$

This is equivalent to the original system if we identify $\tilde{x}_j = x_j - \alpha\, x_i$ (with $\tilde{x}_k = x_k$ for all $k \neq j$).

An Aside: Elementary Column Operations

- An elementary column operation on A is equivalent to an elementary row operation on the transpose A^T.
- Therefore a column swap on A will be given by

$$\left(E_{i\leftrightarrow j}\,A^T\right)^T = A\,E_{i\leftrightarrow j}^T = A\,E_{i\leftrightarrow j}$$

since $E_{i\leftrightarrow j}^T = E_{i\leftrightarrow j}$.
- A row swap results from **pre-multiplying** by $E_{i\leftrightarrow j}$. A column swap results from **post-multiplying** by $E_{i\leftrightarrow j}$.
- Note $E_i(\beta)^T = E_i(\beta)$. **Pre-multiplying** by $E_i(\beta)$ scales row i. **Post-multiplying** by $E_i(\beta)$ scales column i.
- Finally, **pre-multiplying** by $E_{ij}(\alpha)$ results in $R_i \to R_i + \alpha\, R_j$ while **post-multiplying** by $E_{ij}(\alpha)^T$ results in $C_i \to C_i + \alpha\, C_j$.

- post-multiply = Column's
- pre-multiply = Rows

An Aside: Elementary Column Operations

- Let E be any elementary matrix. The effect of a elementary column operation on a system of equations is

$$\left(A\,E^{T}\right)\tilde{\mathbf{x}} = A\left(E^{T}\,\tilde{\mathbf{x}}\right)$$

and so we identify

$$E^{T}\,\tilde{\mathbf{x}} = \mathbf{x}$$

or, since E is easily invertible,

$$\tilde{\mathbf{x}} = E^{-T}\,\mathbf{x}$$

we have equivalent systems of equations.

LU Decomposition

- In Gauss elimination we replace the coefficient matrix A by an **upper triangular matrix** U, the so-called **row reduced echelon form (rref)** of A. The solution of the original system $A\,\mathbf{x} = \mathbf{b}$ is found by solving $U\,\mathbf{x} = \tilde{\mathbf{b}}$ with

$$\left[A \mid \mathbf{b}\right] \to \left[U \mid \tilde{\mathbf{b}}\right]$$

using elementary row operations. This second system is solved by back substitution.

- In order to process the first column of A

$$A \longrightarrow \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & * & * & \cdots & * \\ 0 & * & * & \cdots & * \\ \vdots & \vdots & \vdots & \ddots & \vdots \end{bmatrix} = U^{(1)}$$

we perform the elementary row operations

$$R_i \to R_i - \frac{a_{i1}}{a_{11}}\,R_1.$$

for $i = 2,\ 3,\ \ldots$. These row operations are equivalent to

LU Decomposition

- multiplication by elementary matrices of the form $E_{i1}(\alpha_i)$. Therefore

$$U^{(1)} = E_{n1}(\alpha_n) \cdots E_{31}(\alpha_3)\, E_{21}(\alpha_2)\, A$$

  with

$$\alpha_i = \frac{a_{i1}}{a_{11}}.$$

- Note that

$$E_{31}(\alpha_3)\, E_{21}(\alpha_2) = \begin{bmatrix} \mathbf{e}_1^T \\ \mathbf{e}_2^T \\ \alpha_3\, \mathbf{e}_1^T + \mathbf{e}_3^T \\ \vdots \\ \mathbf{e}_n^T \end{bmatrix} E_{21}(\alpha_2) = \begin{bmatrix} \mathbf{e}_1^T\, E_{21}(\alpha_2) \\ \mathbf{e}_2^T\, E_{21}(\alpha_2) \\ \left(\alpha_3\, \mathbf{e}_1^T + \mathbf{e}_3^T\right) E_{21}(\alpha_2) \\ \vdots \\ \mathbf{e}_n^T\, E_{21}(\alpha_2) \end{bmatrix}$$

LU Decomposition

- Thus

$$E_{n1}(\alpha_n) \cdots E_{31}(\alpha_3)\, E_{21}(\alpha_2) = \begin{bmatrix} \mathbf{e}_1^T \\ \alpha_2\, \mathbf{e}_1^T + \mathbf{e}_2^T \\ \alpha_3\, \mathbf{e}_1^T + \mathbf{e}_3^T \\ \vdots \\ \alpha_n\, \mathbf{e}_1^T + \mathbf{e}_n^T \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ \alpha_2 & 1 & 0 & \cdots & 0 \\ \alpha_3 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \alpha_n & 0 & 0 & \cdots & 1 \end{bmatrix} = L_1.$$

- We perform the *same* row operations on $\mathbf{b}$. Therefore, after the first stage of row reduction, we have

$$U^{(1)} = L_1\, A, \qquad \tilde{\mathbf{b}} = L_1 \mathbf{b}$$

  where L_1 is a **lower triangular matrix**.

LU Decomposition

- L_1 is invertible since

$$L_1^{-1} = E_{21}^{-1}(\alpha_2)\, E_{31}^{-1}(\alpha_3) \cdots E_{n1}^{-1}(\alpha_n) = E_{21}(-\alpha_2)\, E_{31}(-\alpha_3) \cdots E_{n1}(-\alpha_n)$$
$$= \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -\alpha_2 & 1 & 0 & \cdots & 0 \\ -\alpha_3 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -\alpha_n & 0 & 0 & \cdots & 1 \end{bmatrix}$$

- Our original linear system $A\,\mathbf{x} = \mathbf{b}$ is transformed to $L_1\,A\,\mathbf{x} = L_1\,\mathbf{b}$; that is

$$U^{(1)}\,\mathbf{x} = \tilde{\mathbf{b}}.$$

LU Decomposition

- We now repeat the procedure on the second column of $U^{(1)}$; that is, let

$$L_2 = E_{n2}(\alpha_{2n}) \cdots E_{42}(\alpha_{24})\, E_{32}(\alpha_{23})$$

with

$$\alpha_{2j} = \frac{U_{2j}^{(1)}}{U_{22}^{(1)}}$$

(note that the entries in $U^{(1)}$ are not the same as the equivalent entries in A).

- Define

$$U^{(2)} = L_2\, U^{(1)} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & U_{22}^{(1)} & U_{23}^{(1)} & \cdots & U_{2n}^{(1)} \\ 0 & 0 & * & \cdots & * \\ 0 & 0 & * & \cdots & * \\ \vdots & \vdots & \vdots & \ddots & \vdots \end{bmatrix}$$

LU Decomposition

- We proceed inductively. In order to obtain the k^{th} column of the rref of A, we have
$$U^{(k)} = L_k \, U^{(k-1)}$$
with
$$L_k = E_{nk}(\alpha_{kn}) \cdots E_{(k+2)k}(\alpha_{k(k+2)}) \, E_{k(k+1)}(\alpha_{k(k+1)})$$
and
$$\alpha_{kj} = \frac{U_{kj}^{(k-1)}}{U_{kk}^{(k-1)}}.$$
- Therefore the rref form of A (assuming that no row swaps are required) is
$$U \equiv U^{(n)} = L_n \, L_{n-1} \, \cdots \, L_2 \, L_1 \, A;$$
that is,
$$A = (L_n \, L_{n-1} \, \cdots \, L_2 \, L_1)^{-1} \, U = L_1^{-1} \, L_2^{-1} \, \cdots \, L_{n-1}^{-1} \, L_n^{-1} \, U.$$

LU Decomposition

- Let
$$L = L_1^{-1} \, L_2^{-1} \, \cdots \, L_{n-1}^{-1} \, L_n^{-1}.$$
- Note that
$$L_k^{-1} = \begin{bmatrix} \mathbf{e}_1^T \\ \vdots \\ \mathbf{e}_k^T \\ -\alpha_{k(k+1)} \, \mathbf{e}_k^T + \mathbf{e}_{k+1}^T \\ \vdots \\ -\alpha_{kn} \, \mathbf{e}_k^T + \mathbf{e}_n^T \end{bmatrix}$$

LU DECOMPOSITION

- Therefore

$$L_1^{-1} L_2^{-1} = \begin{bmatrix} \mathbf{e}_1^T \\ -\alpha_2 \mathbf{e}_1^T + \mathbf{e}_2^T \\ -\alpha_3 \mathbf{e}_1^T + \mathbf{e}_3^T \\ \vdots \\ -\alpha_n \mathbf{e}_1^T + \mathbf{e}_n^T \end{bmatrix} \begin{bmatrix} \mathbf{e}_1^T \\ \mathbf{e}_2^T \\ -\alpha_{32} \mathbf{e}_2^T + \mathbf{e}_3^T \\ \vdots \\ -\alpha_{n2} \mathbf{e}_2^T + \mathbf{e}_n^T \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1^T \\ -\alpha_2 \mathbf{e}_1^T + \mathbf{e}_2^T \\ -\alpha_3 \mathbf{e}_1^T - \alpha_{32} \mathbf{e}_2^T + \mathbf{e}_3^T \\ \vdots \\ -\alpha_n \mathbf{e}_1^T - \alpha_{n2} \mathbf{e}_2^T + \mathbf{e}_n^T \end{bmatrix}$$

LU DECOMPOSITION

- Continuing this process we see that

$$L = \begin{bmatrix} \mathbf{e}_1^T \\ -\alpha_2 \mathbf{e}_1^T + \mathbf{e}_2^T \\ -\alpha_3 \mathbf{e}_1^T - \alpha_{32} \mathbf{e}_2^T + \mathbf{e}_3^T \\ -\alpha_4 \mathbf{e}_1^T - \alpha_{42} \mathbf{e}_2^T - \alpha_{43} \mathbf{e}_3^T + \mathbf{e}_4^T \\ \vdots \\ -\alpha_n \mathbf{e}_1^T - \alpha_{n2} \mathbf{e}_2^T - \alpha_{n3} \mathbf{e}_3^T - \cdots - \alpha_{n(n-1)} \mathbf{e}_{n-1}^T + \mathbf{e}_n^T \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ -\alpha_2 & 1 & 0 & 0 & \cdots & 0 \\ -\alpha_3 & -\alpha_{32} & 1 & 0 & \cdots & 0 \\ -\alpha_4 & -\alpha_{42} & -\alpha_{43} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ -\alpha_n & -\alpha_{n2} & -\alpha_{n3} & -\alpha_{n4} & \cdots & 1 \end{bmatrix}$$

LU DECOMPOSITION

- L is a **lower triangular matrix**. It comes for "free" when one does a row reduction. One only needs to keep a record of the row operations used.
- This is an example of a **matrix factorisation**; we can write

$$A = LU$$

as a product of a lower triangular matrix L and an upper triangular matrix U. We will see a number of factorisations in this course (diagonalisation, QR-factorisation, SVD).

LU DECOMPOSITION

Example

Find the LU decomposition of

$$A = \begin{bmatrix} 1 & 3 & -1 \\ 2 & 8 & 4 \\ -1 & 3 & 4 \end{bmatrix}.$$

Solution: The first stage of the row reduction of A uses the elementary row operations (note we are not using partial pivoting!)

$$R_2 \to -2R_1 + R_2, \qquad R_3 \to R_1 + R_3.$$

Therefore

$$A \to \begin{bmatrix} 1 & 3 & -1 \\ 0 & 2 & 6 \\ 0 & 6 & 3 \end{bmatrix} = U^{(1)} \qquad L = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & * & 1 \end{bmatrix}.$$

LU Decomposition

The next stage uses the elementary row operation

$$R_3 \to -3R_2 + R_3$$

Therefore

$$A \to \begin{bmatrix} 1 & 3 & -1 \\ 0 & 2 & 6 \\ 0 & 0 & -15 \end{bmatrix} = U \qquad L = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 3 & 1 \end{bmatrix}.$$

Check that $A = LU$.

Solving $A\mathbf{x} = \mathbf{b}$ using a LU Decomposition

- Suppose we wish to solve $A\mathbf{x} = \mathbf{b}$. We first do a row reduction **on** A (not on the augmented matrix $[A \mid \mathbf{b}]$) to obtain

$$A = LU.$$

- Our system now reads

$$LU\mathbf{x} = \mathbf{b}.$$

- Since L is invertible, we could try to solve

$$U\mathbf{x} = L^{-1}\mathbf{b}$$

  by back substitution. However that would require the computation of L^{-1}. While elementary matrices are easy to invert, the product in reverse order is not so easy to compute (see lab problems).

Solving $A\mathbf{x} = \mathbf{b}$ using a LU Decomposition

- Instead, we introduce a new unknown $\mathbf{y}$

$$\mathbf{y} = U\mathbf{x}.$$

- We split the problem into two linear systems. We first solve

$$L\mathbf{y} = \mathbf{b}$$

and **then** solve

$$U\mathbf{x} = \mathbf{y}$$

to obtain the solution.

- The second problem is a standard back substitution. Since L is also triangular, we can solve the first problem by **forward substitution** (that is start at the first row and work our way down).

LU Decomposition

Example

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

$$A = \begin{bmatrix} 1 & 3 & -1 \\ 2 & 8 & 4 \\ -1 & 3 & 4 \end{bmatrix} \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} -4 \\ 2 \\ 4 \end{bmatrix}$$

Solution: From the previous example, we have

$$A = LU = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 3 & 1 \end{bmatrix} \begin{bmatrix} 1 & 3 & -1 \\ 0 & 2 & 6 \\ 0 & 0 & -15 \end{bmatrix}$$

We first solve the system

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

LU Decomposition

We do this by *forward* substitution; that is we start from the first row and work our way down. Thus

$$\begin{aligned} y_1 &= -4 \\ y_2 &= 2 - 2y_1 = 10 \\ y_3 &= 4 + y_1 - 3y_2 = -30. \end{aligned}$$

We now solve the system

$$U\mathbf{x} = \begin{bmatrix} 1 & 3 & -1 \\ 0 & 2 & 6 \\ 0 & 0 & -15 \end{bmatrix} \mathbf{x} = \begin{bmatrix} -4 \\ 10 \\ -30 \end{bmatrix} = \mathbf{y}$$

by back substitution. Thus

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

Why $A\mathbf{x} = \mathbf{b}$ using a LU Decomposition?

- Using a LU decomposition to solve a system of equations requires a row reduction and **both** a back and forward substitution. Thus we appear to have a flop count
$$\mathcal{O}(\tfrac{2}{3}n^3) + 2\mathcal{O}(n^2)$$
(clearly the flop count for a forward substitution is the same as for a back substitution). It appears that we are doing a little extra work. However the row reduction is on the matrix A and not the augmented matrix. This saves us $\mathcal{O}(n^2)$ flops and so there is the same amount of work.
- So why use LU decompositions? In many situations, one is faced with solving $A\mathbf{x} = \mathbf{b}_i$ where A is fixed but there are many $\mathbf{b}_i$ (for example, numerically solving a partial differential equation). With a LU decomposition, one does the row reduction only **once**. If there are m different right hand sides, the flop count would be
$$\mathcal{O}(\tfrac{2}{3}n^3) + 2m\mathcal{O}(n^2).$$
However if we naively solve the equations in full each time, the flop count would be
$$m\mathcal{O}(\tfrac{2}{3}n^3).$$

LU Decomposition

Example

Solve $A^3\,\mathbf{x} = \mathbf{b}$ with

$$A = \begin{bmatrix} 1 & 2 & -1 \\ -2 & -5 & 3 \\ -1 & -3 & 0 \end{bmatrix} \qquad \text{and} \qquad \mathbf{b} = \begin{bmatrix} 9 \\ -17 \\ -44 \end{bmatrix}$$

Solution: We really do not want to compute A^3 since this would require

$$\mathcal{O}(4n^3)$$

flops ($2n^3$ to compute $A^2 = A\,A$ and $2n^3$ to compute $A^3 = A\,A^2$) before we begin the row reduction. Therefore it would require

$$\mathcal{O}(\tfrac{14}{3}n^3)$$

to compute the solution this way.

LU Decomposition

Instead we view this system $A\left(A^2\right)\mathbf{x} = \mathbf{b}$ and rewrite it as

$$A\,\mathbf{y} = \mathbf{b}$$

with $\mathbf{y} = A^2\,\mathbf{x} = A\,(A)\,\mathbf{x}$. We now rewrite this system as

$$A\,\mathbf{z} = \mathbf{y}$$

with $\mathbf{z} = A\,\mathbf{x}$. We obtain *three* systems to solve

$$\begin{aligned} A\,\mathbf{y} &= \mathbf{b} \\ A\,\mathbf{z} &= \mathbf{y} \\ A\,\mathbf{x} &= \mathbf{z} \end{aligned}$$

to obtain $\mathbf{x}$. This is already more efficient with a flop count $\mathcal{O}(2n^3)$. However the coefficient matrix in each of these systems is the same. Therefore we only need to do the row reduction once and so, using LU decomposition, the flop count reduces to $\mathcal{O}(\frac{2}{3}n^3)$.

LU Decomposition

The first stage is to compute the LU decomposition of A:

$$A = \begin{bmatrix} 1 & 2 & -1 \\ -2 & -5 & 3 \\ -1 & -3 & 0 \end{bmatrix} \to \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 1 \\ 0 & -1 & -1 \end{bmatrix} \qquad L = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -1 & * & 1 \end{bmatrix}$$

$$\to \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 1 \\ 0 & 0 & -2 \end{bmatrix} \qquad L = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -1 & 1 & 1 \end{bmatrix}$$

Next we solve $A\,\mathbf{y} = \mathbf{b}$ via $L\,\tilde{\mathbf{y}} = \mathbf{b}$ followed by $U\,\mathbf{y} = \tilde{\mathbf{y}}$:

$$[L \mid \mathbf{b}] = \left[\begin{array}{ccc|c} 1 & 0 & 0 & 9 \\ -2 & 1 & 0 & -17 \\ -1 & 1 & 1 & -44 \end{array}\right] \qquad \tilde{\mathbf{y}} = \begin{bmatrix} 9 \\ 1 \\ -36 \end{bmatrix}$$

$$[U \mid \tilde{\mathbf{y}}] = \left[\begin{array}{ccc|c} 1 & 2 & -1 & 9 \\ 0 & -1 & 1 & 1 \\ 0 & 0 & -2 & -36 \end{array}\right] \qquad \mathbf{y} = \begin{bmatrix} -7 \\ 17 \\ 18 \end{bmatrix}$$

LU Decomposition

We now solve $A\,\mathbf{z} = \mathbf{y}$ via $L\,\tilde{\mathbf{z}} = \mathbf{y}$ followed by $U\,\mathbf{z} = \tilde{\mathbf{z}}$:

$$[L \mid \mathbf{y}] = \left[\begin{array}{ccc|c} 1 & 0 & 0 & -7 \\ -2 & 1 & 0 & 17 \\ -1 & 1 & 1 & 18 \end{array}\right] \qquad \tilde{\mathbf{z}} = \begin{bmatrix} -7 \\ 3 \\ 8 \end{bmatrix}$$

$$[U \mid \tilde{\mathbf{z}}] = \left[\begin{array}{ccc|c} 1 & 2 & -1 & -7 \\ 0 & -1 & 1 & 3 \\ 0 & 0 & -2 & 8 \end{array}\right] \qquad \mathbf{z} = \begin{bmatrix} 3 \\ -7 \\ -4 \end{bmatrix}$$

Finally we solve $A\,\mathbf{x} = \mathbf{z}$ via $L\,\tilde{\mathbf{x}} = \mathbf{z}$ followed by $U\,\mathbf{x} = \tilde{\mathbf{x}}$:

$$[L \mid \mathbf{z}] = \left[\begin{array}{ccc|c} 1 & 0 & 0 & 3 \\ -2 & 1 & 0 & -7 \\ -1 & 1 & 1 & -4 \end{array}\right] \qquad \tilde{\mathbf{x}} = \begin{bmatrix} 3 \\ -1 \\ 0 \end{bmatrix}$$

$$[U \mid \tilde{\mathbf{x}}] = \left[\begin{array}{ccc|c} 1 & 2 & -1 & 3 \\ 0 & -1 & 1 & -1 \\ 0 & 0 & -2 & 0 \end{array}\right] \qquad \mathbf{x} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

LU Decomposition

- This example may seem artificial. However there are many real problems that require the solution of $A^k \mathbf{x} = \mathbf{b}$ (for example, Markov chain problems, discrete dynamical systems).
- Another use of LU decompositions is to evaluate a determinant. We have

$$\det A = \det L \, \det U.$$

  Both L and U are triangular and so their determinant is simply the product of the diagonal elements. Therefore $\det L = 1$ and

$$\det A = \det U = \prod_{k=1}^{n} U_{kk}.$$

- In order to compute the determinant from the cofactor expansion it requires

$$\mathcal{O}(n \cdot n!)$$

  flops. If we use the LU decomposition, it requires $\mathcal{O}(\frac{2}{3}n^3)$.

LU Decomposition

- For example, in order to compute the determinant of 100×100 matrix using a cofactor expansion, it requires approximately

$$0.9 \times 10^{160}$$

  flops! On the fastest supercomputer currently available that would take approximately 10^{134} *years* which is longer than the age of the universe. If we use the LU factorisation, it takes approximately

$$0.67 \times 10^{6}$$

  flops.
- Not surprisingly, MATLAB uses LU factorisations to evaluate determinants. In Lab 1 you may have been surprised that the MATLAB gave a non-integer answer for a 4×4 determinant whose entries were all integers. This is a result of a LU factorisation where the process involves the division by pivots which will give non-integer answers. It also shows that even on small problems, round-off error can accumulate.

Row Swaps and LU Decomposition

- Consider the row reduction

$$A = \begin{bmatrix} 1 & 3 & 2 \\ -2 & -6 & 1 \\ 3 & 5 & 7 \end{bmatrix} \to \begin{bmatrix} 1 & 3 & 2 \\ 0 & 0 & 5 \\ 0 & -4 & 1 \end{bmatrix} \qquad L = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 3 & * & 1 \end{bmatrix}.$$

- At this stage we need to do a row swap $R_2 \leftrightarrow R_3$. This is equivalent to multiplying by the elementary matrix $E_{2\leftrightarrow 3}$.
- Therefore

$$E_{2\leftrightarrow 3}\, A = \begin{bmatrix} 1 & 3 & 2 \\ 3 & 5 & 7 \\ -2 & -6 & 1 \end{bmatrix} \to \begin{bmatrix} 1 & 3 & 2 \\ 0 & -4 & 1 \\ 0 & 0 & 5 \end{bmatrix} \qquad L = \begin{bmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ -2 & * & 1 \end{bmatrix}$$

- Note that we have to swap the elements in the first column of L to reflect the row swap.
- After the row swap, no further row operations are needed. Thus

$$E_{2\leftrightarrow 3}\, A = \begin{bmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ -2 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 3 & 2 \\ 0 & -4 & 1 \\ 0 & 0 & 5 \end{bmatrix} = L\,U.$$

Row Swaps

Example

Find a LU decomposition of

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

using partial pivoting.

Solution: The first steps is to swap rows 1 and 4 and then reduce the first column; that is

$$E_{1\leftrightarrow 4}\, A = \begin{bmatrix} 2 & 2 & 0 & 2 \\ 0 & 1 & 2 & 0 \\ 1 & 3 & 4 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \to \begin{bmatrix} 2 & 2 & 0 & 2 \\ 0 & 1 & 2 & 0 \\ 0 & 2 & 4 & 0 \\ 0 & -1 & 0 & -1 \end{bmatrix} \qquad L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \frac{1}{2} & * & 1 & 0 \\ \frac{1}{2} & * & * & 1 \end{bmatrix}$$

Row Swaps

We now must swap rows 2 and 3

$$E_{2\leftrightarrow 3}\, E_{1\leftrightarrow 4}\, A \to \begin{bmatrix} 2 & 2 & 0 & 2 \\ 0 & 2 & 4 & 0 \\ 0 & 1 & 2 & 0 \\ 0 & -1 & 0 & -1 \end{bmatrix}$$

and adjust the *first column* of L to reflect this row swap

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ 0 & * & 1 & 0 \\ \frac{1}{2} & * & * & 1 \end{bmatrix}.$$

We now reduce the second column

$$E_{2\leftrightarrow 3}\, E_{1\leftrightarrow 4}\, A \to \begin{bmatrix} 2 & 2 & 0 & 2 \\ 0 & 2 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -1 \end{bmatrix}$$

Row Swaps

and update L

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ 0 & \frac{1}{2} & 1 & 0 \\ \frac{1}{2} & -\frac{1}{2} & * & 1 \end{bmatrix}.$$

We now swap rows 3 and 4

$$E_{3\leftrightarrow 4}\, E_{2\leftrightarrow 3}\, E_{1\leftrightarrow 4}\, A \to \begin{bmatrix} 2 & 2 & 0 & 2 \\ 0 & 2 & 4 & 0 \\ 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

and update the *first two columns* of L

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 1 & 0 \\ 0 & \frac{1}{2} & * & 1 \end{bmatrix}.$$

Row Swaps

There are no more row operations required. In terms of row swaps, we have

$$E_{3\leftrightarrow 4}\, E_{2\leftrightarrow 3}\, E_{1\leftrightarrow 4};$$

that is,

$$\begin{aligned} 1 &\to 4 \to 3 \\ 2 &\to 3 \to 4 \\ 3 &\to 2 \\ 4 &\to 1 \end{aligned}$$

Therefore

$$P \equiv E_{3\leftrightarrow 4}\, E_{2\leftrightarrow 3}\, E_{1\leftrightarrow 4} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

Permutation Matrices

Finally we have

$$P\,A = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 1 & 0 \\ 0 & \frac{1}{2} & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 2 & 0 & 2 \\ 0 & 2 & 4 & 0 \\ 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 \end{bmatrix} = L\,U.$$

- P is a **Permutation Matrix**; its action is to reorder the rows (or columns) of A. The rows of a permutation matrix are unit row vectors; that is

$$\mathbf{P}_i^T = \mathbf{e}_j^T$$

for some j. Similiarly the columns of a permutation matrix are unit column vectors

$$\mathbf{p}_i = \mathbf{e}_k$$

for some k.

Permutation Matrices

- Note that

$$P^T P = \begin{bmatrix} \mathbf{p}_1^T \\ \vdots \\ \mathbf{p}_n^T \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 & \cdots & \mathbf{p}_n \end{bmatrix}$$

since the rows of P^T are the same as the columns of P. However each of the columns of P are unit vectors $\mathbf{e}_k$ (and the k is never repeated) we have

$$P^T P = I.$$

Thus

Theorem 1.2

If P is a permutation matrix then

$$P^{-1} = P^T.$$

LU Decomposition

- Since P is easy to invert, we write

$$A = P^T L U$$

This is known as a LU **factorisation** of A.
- It is achieved by a row reduction of the matrix A. Thus

Theorem 1.3

Let A be a square matrix. Then A has a LU factorisation.

- Note the LU factorisation is **NOT** unique. It depends on the row ordering that one chooses.
- One can find the LU factorisation as we did above by modifying L after each row swap or one can complete the reduction, noting the row swaps P, and then repeat the reduction on PA to compute L or use MATLAB!

MATLAB AND LU DECOMPOSITION

- The `lu` command in MATLAB computes tthe LU factorisation with partial pivoting.

```
>> A=[1 0 0 0; 0 1 2 0;1 3 4 1; 2 2 0 2]
A =
     1     0     0     0
     0     1     2     0
     1     3     4     1
     2     2     0     2

>> lu(A)
ans =
    2.0000    2.0000         0    2.0000
    0.5000    2.0000    4.0000         0
    0.5000   -0.5000    2.0000   -1.0000
         0    0.5000         0         0
```

MATLAB AND LU DECOMPOSITION

- In this case, MATLAB has displayed the LU factorisation in *compressed form*. Since U is upper triangular, it uses the lower triangular part to store L (it uses this form internally to reduce space requirements). In order to see the factors

```
>> [L,U] = lu(A)
L =
    0.5000   -0.5000    1.0000         0
         0    0.5000         0    1.0000
    0.5000    1.0000         0         0
    1.0000         0         0         0
U =
     2     2     0     2
     0     2     4     0
     0     0     2    -1
     0     0     0     0
```

- Here the first factor is $P^T L$.

Matlab and LU Decomposition

- To see all three factors, we

```
>> [L,U,P]=lu(A)
L =
    1.0000         0         0         0
    0.5000    1.0000         0         0
    0.5000   -0.5000    1.0000         0
         0    0.5000         0    1.0000
U =
     2     2     0     2
     0     2     4     0
     0     0     2    -1
     0     0     0     0
P =
     0     0     0     1
     0     0     1     0
     1     0     0     0
     0     1     0     0
```