6.1 The LU Factorization without Pivoting

Given a matrix $A \in \mathbb{R}^{m \times n}$, a matrix factorization is an equation that expresses A as a product of two or more matrices. Here are some very powerful factorizations

A = LU LU Factorization for Square Matrix $A \in \mathbb{R}^{n \times n}$

$$A = LL^T$$
 Cholesky Factorization for Square, Symmetric Matrix $A = A^T \in \mathbb{R}^{n \times n}$

- A = QR QR Factorization for Rectangular Matrix $A \in \mathbb{R}^{m \times n}$
- $A = U\Sigma V^T$ Singular Value Decomposition (SVD) for Rectangular Matrix $A \in \mathbb{R}^{m \times n}$

As we will see, matrix factorizations are used to simplify the solutions to each of the following problems

ii. Nonsingular linear-systems problems:

Given a nonsingular matrix $A \in \mathbb{R}^{n \times n}$ and $\mathbf{b} \in \mathbb{R}^n$, find $\mathbf{x} \in \mathbb{R}^n$ such that

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

iii. Least-squares problems:

Given $A \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$, find

$$\min_{\mathbf{x}\in\mathbb{R}^n} \|A\mathbf{x}-\mathbf{b}\|_2$$

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

iv. Eigenvalue problems:

Given $A \in \mathbb{R}^{n \times n}$, find scalar λ and $n \times 1$ vector **x** such that

Notice that in each of these three problem types, we are given a matrix A and asked to find a vector \mathbf{x} with some special properties. The main idea behind matrix factorization is to spend some time and energy pre-processing the "given" matrix A into very special parts.

These parts (known as factors) can then be used to more easily compute solutions to our main problem types. Also, if done correctly, the factorizations lead to more accurate approximate solutions when we use computers to solve these problems. In this sense, you can think of matrix factorizations as an analysis problem, rather than a synthesis problem. Let's look at an analogy. Suppose you are given two scalars 11 and 70 and you are asked to find the product $11 \cdot 70$. We might call this a synthesis problem, also known as a forward problem, since we are asked to synthesize new information from given information. In order to solve this problem, we don't necessarily have to pre-process either of our two given inputs. Instead we can use brute force to find $11 \cdot 70 = 770$. Similarly, we can think of the matrix-matrix multiplication problem as a synthesis problem:

Given two pieces of data (a matrix matrices $A \in \mathbb{R}^{n \times n}$ and vector $\mathbf{x} \in \mathbb{R}^n$), combine this data together to create new data (the product)

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

where $\mathbf{b} \in \mathbb{R}^n$ is given by the matrix-vector product

$$\mathbf{b} = \sum_{k=1}^{n} x_k A(:,k)$$

Because of our definition, we do not need to process the two input matrices A or **x** to calculate the product. Instead, we can use brute force and find the product using vector operations.

Now, let's consider an analysis problem, also known as an inverse problem. Suppose we want to know the number x such that 22x = 770. In this case, we are given only one input, the operation that effects this input and the corresponding output. Using this information, we want to find our other input x. Of course, we could say that

$$x = \frac{770}{22}$$

and use our brute force division algorithm to find our answer. However, this is both frustrating and time consuming. The algorithm requires guess-and-check machinery and requires many arithmetic sub-steps. This leaves us yearning for a better way to do this. This is where factorizations come in.

If we spend a few minutes pre-processing the input $22 = 2 \cdot 11$ into unique factors, then we can simplify our problem. In particular, we see

$$x = \frac{770}{2 \cdot 11} = \frac{10}{2} \cdot \frac{77}{11} = 5 \cdot 7 = 35$$

We designed our factorization to give us much better insight and a quicker way to find the solution to our inverse problem.

This is how we will think about all of our matrix factorizations. For the case of the LU Factorization, we will be solving the linear systems problem.

Given a matrix matrices $A \in \mathbb{R}^{m \times n}$ and output vector $\mathbf{b} \in \mathbb{R}^{m}$), find the input vector(s) $\mathbf{x} \in \mathbb{R}^{n}$ such that

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

In general, it is not very easy to see which scalar weights on the columns of A create a linear combinations the results in **b**. This is partially due to the fact that A is not written in a form that makes these calculations easy. In particular, to get such a solution as is, we need to do a ton of work and arithmetic sub-steps to transform into RREF form. We are left craving a better way. This is where the LU Factorization comes in.

Definition 6.1: LU Factorization without Pivoting

Let $A \in \mathbb{R}^{n \times n}$ be a given, square, invertible matrix with non-zeros on the main diagonal elements. An LU factorization of A is given by

A = LU

where $U \in \mathbb{R}^{n \times n}$ is upper-triangular with nonzero diagonal elements. Also, L is unit lower-triangular with all of its diagonal entries equal to 1.

Suppose we have the LU Factorization A = LU of our given matrix. Then, we can transform the linear-systems problem $A\mathbf{x} = \mathbf{b}$ into to related problems



We can solve the linear systems problem $L\mathbf{y} = \mathbf{b}$ using forward substitution and $U\mathbf{x} = \mathbf{y}$ using backward substitution.

EXAMPLE 6.1.1

Let's look back at Example 5.1.4. In this problem, we worked to create a quadratic function model for the motion of a falling object using the linear-systems problem

1	3.0	9.00	$\begin{bmatrix} x_1 \end{bmatrix}$		3.000
1	3.3	10.89	x_2	2	2.559
1	3.6	12.96	x_3		1.236

We used elementary matrices to transform A into upper triangular form, as follows

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -2 & 1 \end{bmatrix}}_{L_2} \cdot \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}}_{L_1} \cdot \underbrace{\begin{bmatrix} 1 & 3.0 & 9.00 \\ 1 & 3.3 & 10.89 \\ 1 & 3.6 & 12.96 \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} 1 & 3.0 & 9.00 \\ 0 & 0.3 & 1.89 \\ 0 & 0 & 0.18 \end{bmatrix}}_{U}$$

We can write this transformation symbolically as

$$L_2 \cdot L_1 \cdot A = U$$

where L_i is a unit lower-triangular matrix and thus is invertible. Moreover, from our discussion of inverses, we see that

$$L_2^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix} \qquad \qquad L_1^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

With this, we multiply both sides of our transformed equation of $L_2\cdot L_1\cdot A=U$ by $(L_2\cdot L_1)^{-1}$ to find

$$\underbrace{\begin{bmatrix} 1 & 3.0 & 9.00 \\ 1 & 3.3 & 10.89 \\ 1 & 3.6 & 12.96 \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}}_{L_1^{-1}} \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix}}_{L_2^{-1}} \underbrace{\begin{bmatrix} 1 & 3.0 & 9.00 \\ 0 & 0.3 & 1.89 \\ 0 & 0 & 0.18 \end{bmatrix}}_{U}$$

Setting $L = L_1^{-1} \cdot L_2^{-1}$, we have A = LU is the LU Factorization of A with

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

We can use this LU Factorization to solve our linear systems problem by solving two equivalent problems

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 2 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 3.000 \\ 2.559 \\ 1.236 \end{bmatrix}, \qquad \begin{bmatrix} 1 & 3.0 & 9.00 \\ 0 & 0.3 & 1.89 \\ 0 & 0 & 0.18 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

Using forward substitution, we see that

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 3.000 \\ -0.441 \\ -0.882 \end{bmatrix}$$

We can substitute this vector into our matrix equation $U\mathbf{x} = \mathbf{y}$ to find our desired \mathbf{x} using backward substituion

$$\begin{bmatrix} x_1\\x_2\\x_3 \end{bmatrix} = \begin{bmatrix} -41.1\\29.4\\-4.9 \end{bmatrix}$$

This is the same solution we found in Example 5.1.4.

In our example above, we transformed the original matrix A into upper-triangular form by introducing zero entries into the strictly lower triangular portions of A. This algorithm will work in general to produce the LU Factorization of a given matrix A. Starting with column 1, then column 2 and continued through column (n-1), introduce zeros below the main diagonal elements by multiplying A on the left by a sequence of unit lower-triangular matrices in the form

$$L_{n-1} \cdot L_{n-2} \cdots L_2 \cdot L_1 \cdot A = U$$

The resulting upper-triangular matrix $U \in \mathbb{R}^{n \times n}$ is right factor from the LU factorization. Further, the unit lower-triangular matrix $L_k \in \mathbb{R}^{n \times n}$ introduces zeros in the *k*th column for k = 1, 2, ..., (n - 1). As we will see, we can produce the left factor *L* using the formula

$$L = L_1^{-1} \cdot L_2^{-1} \cdots L_{n-2}^{-1} \cdot L_{n-1}^{-1}$$

Because of the structure of each matrix L_k , we can construct the inverse L_k^{-1} very quickly. In this manner we produce the LU factorization of matrix A = LU.

EXAMPLE 6.1.2

Show example here. Ideal example comes from applications.

For k = 1, 2, ..., (n - 1), let's consider at the beginning of the kth step of our reduction algorithm. We begin with the matrix A. We then apply a series of lower triangular matrices to produce an updated matrix

$$L_{k-1}\cdots L_1 \cdot A = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1,k-1} & u_{1k} & u_{1,k+1} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2,k-1} & u_{2k} & u_{2,k+1} & \cdots & u_{2n} \\ \vdots & \ddots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & u_{k-1,k-1} & u_{k-1,k} & u_{k-1,k+1} & \cdots & u_{k-1,n} \\ 0 & \cdots & 0 & 0 & u_{k,k} & u_{k,k+1} & \cdots & x_{k,n} \\ 0 & \cdots & 0 & 0 & x_{k+1,k} & x_{k+1,k+1} & \cdots & x_{k+1,n} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & x_{nk} & x_{n,k+1} & \cdots & x_{nn} \end{bmatrix}$$

Let's focus on introducing zeros in the kth column of this matrix. In particular, at the beginning of the kth step let's introduce call the kth column \mathbf{x}_k . We want to choose matrix L_k such that

$$L_{k}\mathbf{x}_{k} = \begin{bmatrix} u_{1k} \\ \vdots \\ u_{kk} \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \text{where} \quad \mathbf{x}_{k} = \begin{bmatrix} u_{1k} \\ \vdots \\ u_{k,k} \\ x_{k+1,k} \\ \vdots \\ x_{nk} \end{bmatrix}$$

In order to accomplish this, we add $-\ell_{jk}$ times row k to row j, where we choose the coefficient

$$\ell_{jk} = \frac{x_{jk}}{u_{kk}}$$

for all $k < j \le n$. We see that we can write the matrix L_k as a Gauss transformation

$$L_{k} = I_{n} - \boldsymbol{\tau}_{k} \mathbf{e}_{k}^{T} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \ddots & \vdots & \vdots & \ddots & \vdots & 0 \\ \vdots & \ddots & \ddots & 0 & 0 & \cdots & 0 & 0 \\ 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & \cdots & 0 & -\ell_{k+1,k} & 1 & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & -\ell_{k+2,k} & 0 & \ddots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ 0 & \cdots & 0 & -\ell_{n,k} & 0 & \cdots & 0 & 1 \end{bmatrix}, \quad \text{where } \boldsymbol{\tau}_{k} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \ell_{k+1,k} \\ \vdots \\ \ell_{n,k} \end{bmatrix}$$

Notice that we can now calculate the inverse

$$L_k^{-1} = I_n + \boldsymbol{\tau}_k \mathbf{e}_k^T$$

by simply changing the sign in front of the τ_k . This is a generalization of the inverse of a shear matrix

$$\left(S_{ik}(c)\right)^{-1} = S_{ik}(-c)$$

Lesson 16: The LU Factorization- Suggested Problems List

For Problems 1 - 3, consider the following model for a 4-mass, 5-spring chain. Note that positive positions and positive displacements are marked in the downward direction. Assume the acceleration due to earth's gravity is $g = 9.8m/s^2$. Also assume that the mass of each spring is zero and that these springs satisfy the ideal version of Hooke's law exactly.



1. Recall our model for the mass-spring chain above is given by

$$M\ddot{\mathbf{u}}(t) + K\mathbf{u}(t) = \mathbf{f}_e(t).$$

Find the stiffness matrix K that results from this model.

- 2. Find the LU factorization of K = LU.
- 3. For each of the mass vectors $\mathbf{m} \in \mathbb{R}^4$ given below, solve the associated square linear-systems problems $K \cdot \mathbf{u} = \mathbf{f}_e$ using the LU factorization of K. Explicitly show the forward and backward substitution steps in each case.

(a)
$$\begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{bmatrix} = \begin{bmatrix} 0.025 \\ 0.050 \\ 0.025 \end{bmatrix}$$
 (b) $\begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{bmatrix} = \begin{bmatrix} 0.050 \\ 0.100 \\ 0.100 \\ 0.050 \end{bmatrix}$ (c) $\begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{bmatrix} = \begin{bmatrix} 0.050 \\ 0.025 \\ 0.025 \\ 0.050 \end{bmatrix}$

4. Solve each of the following linear-system problem by transforming the matrix equation $A\mathbf{x} = \mathbf{b}$ into an equivalent equation $U\mathbf{x} = \mathbf{y}$ by multiplying both sides of the equation by appropriate unit lower-triangular matrices and then applying the backward substitution algorithm.

A.
$$\begin{bmatrix} 5 & 2 & 3 & 6 \\ -10 & -7 & -5 & -10 \\ -5 & -11 & 1 & 4 \\ 10 & 16 & 0 & -2 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} -11 \\ 20 \\ 7 \\ -14 \end{bmatrix}$$

B.
$$\begin{bmatrix} 5 & 2 & 3 & 6 \\ -10 & -7 & -5 & -10 \\ -5 & -11 & 1 & 4 \\ 10 & 16 & 0 & -2 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} -10 \\ 17 \\ -11 \\ 10 \end{bmatrix}$$

C.
$$\begin{bmatrix} 5 & 2 & 3 & 6 \\ -10 & -7 & -5 & -10 \\ -5 & -11 & 1 & 4 \\ 10 & 16 & 0 & -2 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 16 \\ -23 \\ 21 \\ -20 \end{bmatrix}$$

- D. What do you notice about the unit lower-triangular matrices that you used to transform each of the systems in parts A, B, and C above? What do you notice about each of the matrices U that result from these transformations?
- 5. Consider the following matrix and vectors

$$A = \begin{bmatrix} 5 & 2 & 3 & 6 \\ -10 & -7 & -5 & -10 \\ -5 & -11 & 1 & 4 \\ 10 & 16 & 0 & -2 \end{bmatrix}, \quad \mathbf{b}_1 = \begin{bmatrix} -11 \\ 20 \\ 7 \\ -14 \end{bmatrix}, \quad \mathbf{b}_2 = \begin{bmatrix} -10 \\ 17 \\ -11 \\ 10 \end{bmatrix}, \quad \mathbf{b}_3 = \begin{bmatrix} 16 \\ -23 \\ 21 \\ -20 \end{bmatrix}.$$

- A. Find unit lower-triangular matrices $L_1, L_2, L_3 \in \mathbb{R}^{4 \times 4}$ such that $L_3 \cdot L_2 \cdot L_1 \cdot A = U$, where the matrix U is upper triangular matrix.
- B. Use your work in part a. to find the LU factorization of the matrix A.
- C. Use the LU factorization to solve the three linear systems problems: $A \cdot \mathbf{x} = \mathbf{b}_k$ for k = 1, 2, 3
- D. Compare your work in problem 1 with your work in problem 2. If you are solving a single nonsingular linear-systems problem with coefficient matrix A, is it necessary to find the LU Factorization of this matrix? What if you are solving many different nonsingular linear-systems problems with the same matrix A and a variety of different right-hand side vectors \mathbf{b}_k ? What is the benefit of solving these multiple systems by finding the LU Factorization of A (rather than transforming each system individually)?

6. Consider the following matrices:

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

- A. For k = 3, 4, 5, use a sequence of matrix multiplications to transform A_k into upper-triangular U_k . Specifically identify unit lower-triangular matrices $L_1, L_2, ..., L_t$ needed to make this transformation in each case.
- B. Find the LU factorization of the matrix ${\cal A}$ from above.
- 7. Let $\ell_{ik} \in \mathbb{R}$ for all $i, k \in \mathbb{N}$ and define the matrices $L_1, L_2, L_3 \in \mathbb{R}^{4 \times 4}$ be given by

$$\begin{split} L_{2} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ \ell_{21} & 1 & 0 & 0 \\ \ell_{31} & 0 & 1 & 0 \\ \ell_{41} & 0 & 0 & 1 \end{bmatrix} = I_{4} + \tau_{1} \cdot \mathbf{e}_{1}^{T}, \qquad \text{where } \tau_{1} = \begin{bmatrix} 0 \\ \ell_{21} \\ \ell_{31} \\ \ell_{41} \end{bmatrix}. \\ L_{2} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & \ell_{42} & 0 & 1 \end{bmatrix} = I_{4} + \tau_{2} \cdot \mathbf{e}_{2}^{T}, \qquad \text{where } \tau_{2} = \begin{bmatrix} 0 \\ 0 \\ \ell_{32} \\ \ell_{42} \end{bmatrix}. \\ L_{3} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \ell_{43} & 1 \end{bmatrix} = I_{4} + \tau_{3} \cdot \mathbf{e}_{3}^{T}, \qquad \text{where } \tau_{3} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \ell_{43} \end{bmatrix}. \\ \text{A. Prove } L_{k}^{-1} &= (I_{4} - \tau_{k} \cdot \mathbf{e}_{k}^{T}) \text{ for } k = 1, 2, 3. \\ \text{B. Prove } L_{k}^{-1} \cdot L_{j}^{-1} = I_{4} - \tau_{k} \cdot \mathbf{e}_{k}^{T} - \tau_{j} \cdot \mathbf{e}_{j}^{T} \text{ for all } j < k \\ \text{C. Use parts A and B to conclude the} \\ L &= L_{3}^{-1} \cdot L_{2}^{-1} \cdot L_{1}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -\ell_{21} & 1 & 0 & 0 \\ -\ell_{31} & -\ell_{32} & 1 & 0 \\ -\ell_{41} & -\ell_{42} & -\ell_{43} & 1 \end{bmatrix}. \end{split}$$

D. Is the following statement true or false. Justify your answer with explicit analysis and reasoning:

$$L_1 \cdot L_2 \cdot L_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \ell_{21} & 1 & 0 & 0 \\ \ell_{31} & \ell_{32} & 1 & 0 \\ \ell_{41} & \ell_{42} & \ell_{43} & 1 \end{bmatrix}.$$