Numerical Linear Algebra

The two principal problems in linear algebra are:

Linear system Given an $n \times n$ matrix A and an n-vector $\vec{b}$, determine $\vec{x} \in I\!R^n$ such that

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

Eigenvalue problem Given an $n \times n$ matrix A, find a scalar λ (an eigenvalue) and a *nonzero* vector $\vec{x}$ (an eigenvector) such that

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

Goals:

- to review some basic concepts from undergraduate linear algebra;
- to study the three basic decompositions of a matrix

  $A = LU$ (and its variants) $\qquad A = QR \qquad A = U\Sigma V^T$

  and understand what they tell us;
- to understand when a linear system is uniquely solvable;
- to be able to calculate and manipulate norms of vectors and matrices;
- given properties of the coefficient matrix A, determine the best algorithm to solve $A\vec{x} = \vec{b}$;
- to be able to compare the work required for two algorithms for solving a linear system;
- to understand what it means for a system to be ill-conditioned and the consequences resulting from this ill-conditioning;
- to be exposed to both direct and iterative methods for solving linear systems

and to understand when it is advantageous to use one or the other;

- to understand the importance of eigenvalues and eigenvectors and how to calculate them;
- to consider non-square systems where A is $m \times n$.

Some Important Results from Undergraduate Linear Algebra

References

We will briefly review some basic results from undergraduate linear algebra that we will need throughout our exploration of numerical linear algebra. If you are not familiar with these, then you should review a good undergraduate text. Recommended graduate texts which review this material and contain the material we will cover on numerical linear algebra are also listed below.

- G. Strange, *Linear Algebra* (undergraduate text)
- G.W. Stewart,*Introduction to Matrix Computations* (not very up-to-date but very readable)
- G. Golub and C. van Loan, *Matrix Computations*
- L. N. Trefethern and D. Bau, *Numerical Linear Algebra*

Vectors

- We will denote an n-vector as $\vec{x}$ and its components as x_i, $i = 1, \ldots, n$. We think of $\vec{x}$ as a column vector and $\vec{x}^T$ as a row vector.

- To add two vectors, they must have the same length and then we add corresponding entries. To multiply a vector by a scalar α, we multiply each entry by α. For n-vectors $\vec{x}, \vec{y}$ and scalar α

$$\vec{c} = \alpha\vec{x} + \vec{y} \quad \text{where} \quad c_i = \alpha x_i + y_i$$

- To take the dot or scalar or inner product of two n-vectors $\vec{x}$ and $\vec{y}$ we form

$$\sum_{i=1}^{n} x_i y_i$$

so the dot product of two vectors is a scalar. We denote the scalar product as

$$\vec{x} \cdot \vec{y} \quad \text{or} \quad \vec{x}^T\vec{y} \quad \text{or} \quad (\vec{x}, \vec{y})$$

- If two vectors have complex entries then their inner product is given by $\vec{x^*}^T\vec{y}$ where * denotes the complex conjugate.

- The standard Euclidean length of a vector is $\|\vec{x}\|_2 = [\vec{x}^T\vec{x}]^{1/2} = (x,x)^{1/2}$ (we will discuss this notation later)
- We also know that
$$\vec{x}^T\vec{y} = \|\vec{x}\|_2\|\vec{y}\|_2 \cos\theta$$
where θ is the angle between the vectors.
- Two vectors are called perpendicular or orthogonal if $\vec{x}^T\vec{y} = 0$.

Vector Spaces

- A vector or linear space V is a set of objects, which we will call vectors, for which addition and scalar multiplication are defined and satisfy the following properties.

(i) $x + y = y + x$
(ii) $x + (y + z) = (x + y) + z$
(iii) there exists a zero element $0 \in V$ such that $x + 0 = 0 + x = x$
(iv) for each $x \in V$ there exists $-x \in V$ such that $x + (-x) = 0$
(v) $1x = x$

(vi) $(\alpha+\beta)x = \alpha x + \beta x$, for scalars α, β

(vii) $\alpha(x+y) = \alpha x + \alpha y$

(viii) $(\alpha\beta)x = \alpha(\beta x)$

- We will use the notation $\mathbb{R}^n$ to denote the standard Euclidean vector space of dimension n; it consists of all real vectors of length n with the usual definitions of addition and scalar multiplication. So instead of saying $\vec{x}$ is an n-vector we will usually write $\vec{x} \in \mathbb{R}^n$. $\mathbb{R}^2$ is our standard two dimensional Euclidean space which consists of all points (or all vectors) (x_i, y_i).

- Other sets of objects (such as matrices, polynomials, etc) can also be "vectors" in a vector space.

- A set $S \subset V$ is called a subspace of V provided it is closed under addition and scalar multiplication, i.e., for every $x, y \in S$ and scalars α, β we have that $\alpha x + \beta y \in S$.

- Consider a set of vectors $\{\vec{v}_i\}$, $i = 1, \ldots, n$ in the vector space V

– The vectors are linearly independent if there exist constants C_i such that

$$\sum_{j=1}^{n} C_j \vec{v}_j = \vec{0} \quad \Longrightarrow \quad C_j = 0, \forall j$$

otherwise they are linearly dependent. Note that this says that the only way we can combine linearly independent vectors and get the zero vector is if all coefficients are zero. If a set of vectors $\{\vec{v}_i\}_{i=1}^{N}$ are linear dependent this says that there is some i where $C_i \neq 0$ and thus

$$\vec{v}_i = \sum_{\substack{j=1 \\ j \neq i}}^{N} \frac{C_j}{C_i} \vec{v}_j$$

i.e., $\vec{v}_i$ is a linear combination of the other vectors.

– The vectors $\{\vec{v}_j\}_{j=1}^{M} \in V$ span V if any $\vec{w} \in V$ can be written as a linear combination of the $\vec{v}_j$, i.e.,

$$\vec{w} = \sum_{j=1}^{M} C_j \vec{v}_j$$

– The vectors $\{\vec{v}_j\}_{j=1}^N \in V$ form a basis for V if they are linearly independent and span V.

Exercise Determine if each set of vectors in $\mathbb{R}^3$ (i) is linearly independent or linear dependent, (ii) spans $\mathbb{R}^3$, (iii) forms a basis for $\mathbb{R}^3$.

$$S_1 = \left\{(2,4,-1)^T, (1,2,5)^T\right\}, \quad S_2 = \left\{(1,2,3)^T, (0,1,1)^T, (1,4,5)^T\right\},$$

$$S_3 = \left\{(2,4,-1)^T, (1,2,5)^T, (1,0,0)^T, (0,1,0)^T\right\}$$

- If the dimension of the basis is N then we say that V is finite dimensional; otherwise it is an infinite dimensional linear space.
- All bases for a finite dimensional space have the same number of elements; we call this the dimension of V.

Example Let V be the set of all polynomials of degree ≤ 3 with the usual definitions of addition and scalar multiplication.

Then V is a vector space and the set of vectors

$$\{1, x, x^2, x^3\}$$

form a basis because they are obviously linearly independent and span the space.

The vectors $\{1+x, x^2+x^3\}$ do not form a basis because we need 4 for a basis. The vectors are linearly independent but they do not span V; e.g., there are no constants C_1, C_2 such that

$$C_1(1+x) + C_2(x^2+x^3) = x$$

The vectors $\{1, x, x^2, x^3, 2x^2-1\}$ do not form a basis because they are not linearly independent although they span V. To see this note that there are constants, not all zero, so that we can combine the vectors and get zero, e.g.,

$$(1)1 + (0)x + (-2)x^2 + (0)x^3 + (1)\big(2x^2 - 1\big) = 0$$

Exercise Let V be the plane in $\mathbb{R}^3$ that pass through the origin, i.e., such that $x_1 + x_2 + x_3 = 0$. Justify that V is a subspace of $\mathbb{R}^3$, give its dimension and a basis.

Exercise Give an example of an infinite dimensional vector space.

Matrices

- We will denote the entries of a matrix A as A_{ij} or a_{ij} where i denotes the ith row and j denotes the jth column. For a matrix with m rows and n columns we have

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{pmatrix}$$

- We will assume that the entries of A are real unless specifically stated otherwise.

- Matrix addition is defined if the matrices have the same number of rows and columns; in this case

$$C = A + B \quad \text{where } c_{ij} = a_{ij} + b_{ij}$$

- Scalar multiplication is defined by multiplying each entry of the matrix by the

scalar

$$C = \alpha A \quad \text{where } c_{ij} = \alpha a_{ij}$$

- The set of all $m \times n$ matrices with these definitions of addition and scalar multiplication form a vector space.

- For the product of a matrix and a vector to be defined, the vector must have the same dimension as the number of columns of A. If A is $m \times n$ and $\vec{x} \in I\!R^n$ then the result is a vector $\vec{b} \in I\!R^m$

$$\vec{b} = A\vec{x} \quad \text{where } b_i = \sum_{j=1}^{n} a_{ij}x_j \quad i = 1, 2, \ldots, m$$

  Note that we can also view the product $A\vec{x}$ as

$$\vec{b} = x_1\vec{a}_1 + x_2\vec{a}_2 + \cdots + x_n\vec{a}_n$$

  where $\vec{a}_i$ denotes the ith column of A. This interpretation of $\vec{b}$ being a linear combination of the columns of A will be useful.

- The quantity $\vec{x}\vec{x}^T$ for $\vec{x} \in I\!R^n$ is called the outer product of two vectors (as opposed to the inner product) and the result is an $n \times n$ matrix.

- For the product of two matrices A and B to be defined, the number of columns of A must be the same as the number of rows of B. If A is $m \times p$ and B is $p \times n$ then the result is an $m \times n$ matrix given by

$$C = AB \quad \text{where } c_{ij} = \sum_{k=1}^{p} a_{ik} b_{kj}$$

  Note that if $C = AB$ then BA might not even be defined. However, if A and B are square then both AB and BA are defined but in general, $AB \neq BA$, i.e., matrix multiplication is NOT commutative.

- We will often compute the quantity $\vec{x}^T A \vec{x}$ where A is $n \times n$ and $\vec{x} \in \mathbb{R}^n$. Note that this quantity is a scalar and is formed by

$$\vec{x}^T \begin{pmatrix} \sum_{j=1}^{n} a_{1j} x_j \\ \sum_{j=1}^{n} a_{2j} x_j \\ \vdots \\ \sum_{j=1}^{n} a_{nj} x_j \end{pmatrix} = \sum_{i=1}^{n} \Big(\sum_{j=1}^{n} a_{ij} x_j \Big) x_i$$

- Some matrices have special names due to their pattern of zeros.
  - A diagonal matrix has all entries $a_{ij} = 0$ for all $i \neq j$; the diagonal matrix

with all 1's on the diagonal is called the identity matrix

– An upper triangular matrix A has entries $a_{ij} = 0$ for all $i > j$.

– A lower triangular matrix A has entries $a_{ij} = 0$ for all $j > i$.

– If A is $m \times n$ then we sometimes use the terminology upper or lower trapezoidal.

– A tridiagonal matrix A has entries $a_{ij} = 0$ for all $|i - j| > 1$.

– A lower bidiagonal matrix A has entries $a_{ij} = 0$ for all $j > i$ or $i > j + 1$; analogous definition for upper bidiagonal.

– A matrix is upper Hessenberg if $a_{ij} = 0$ for $i > j + 1$.

– A banded matrix of bandwidth $2p+1$ has $a_{ij} = 0$ for $|i-j| > p$; for example a tridiagonal matrix has bandwidth 3. A matrix may have a different lower and upper bandwidth; in this case the total bandwidth is $p + q + 1$ where p is the lower bandwidth and q the upper.

– A general sparse matrix is assumed to have the majority of its entries $=0$; however they may not be arranged in any pattern.

- The inverse of an $n \times n$ matrix A (if it exists) is a matrix, denoted A^{-1} such that
$$AA^{-1} = A^{-1}A = I$$
where I denotes the $n \times n$ identity matrix.

- If A^{-1} exists then we say that A is invertible or nonsingular; otherwise we say that A is singular. We only talk about invertibility for square matrices; there will be something called a pseudo inverse for rectangular matrices.

- To find the inverse of the product of two square invertible matrices we have
$$(AB)^{-1} = B^{-1}A^{-1} \quad \text{because } AB(B^{-1}A^{-1}) = AIA^{-1} = I$$

- A^T will denote the transpose of a matrix; that is, $\left(A^T\right)_{ij} = A_{ji}$. Recall that to take the transpose of the product of two matrices you must reverse the order
$$(AB)^T = B^T A^T$$

- If A is a matrix that possibly has complex entries then instead of taking its transpose we take the complex conjugate and then the transpose; we write

$A^H = (A^*)^T$ where $*$ denotes the complex conjugate; i.e., everywhere there is an i (i.e., $\sqrt{-1}$) replace with $-i$.

- Some matrices are classified due to their inherent properties.
  - A real matrix is symmetric if $A = A^T$.
  - A real matrix is skew-symmetric if $A = -A^T$.
  - A complex matrix is Hermitian if $A = A^H$ where A^H denotes the transpose of the complex conjugate of A, i.e., $A^{*T} = A^H$.
  - A real matrix is positive definite if $\vec{x}^T A\vec{x} > 0$ for all $\vec{x} \neq \vec{0}$.
  - A real matrix is positive semi-definite if $\vec{x}^T A\vec{x} \geq 0$ for all $\vec{x}$.
  - A complex matrix is positive definite if $\vec{x^*}^T A\vec{x} > 0$ for all $\vec{x} \neq \vec{0}$.
  - A real matrix is orthogonal if $AA^T = I$ (i.e., its inverse is its transpose).
  - A complex matrix is unitary if $AA^H = I$ (i.e., its inverse is its complex conjugate transpose).
  - A matrix is (strictly) diagonally dominant if $|a_{ii}| > \sum_{i \neq j} |a_{ij}|$ for all $i =$

$1, 2, \ldots, n$.

Exercise Consider the matrix $B = A^T A$. What are the properties of B?

Exercise Let V be the vector space of all 3×3 symmetric matrices. What is the dimension of V? What is a basis for V?

Exercise If a matrix is both upper and lower triangular, what is it?

Exercise If a real symmetric matrix is orthogonal, what is its inverse?

Exercise If a real matrix is positive definite, can you say anything about its diagonal entries?

Exercise Give an example (not the zero matrix) of a square matrix that satisfies $A^2 = A$; here A^2 is shorthand notation for the product AA. Such a matrix is called idempotent.

Exercise Give an example (not the zero matrix) of a square matrix A that

satisfies A^k is the zero matrix for some integer k. Such a matrix is called nilpotent.

Four Important subspaces

There are four important spaces associated with a matrix. These spaces are related and are essential in analyzing certain algorithms. We summarize them here for a general $m \times n$ matrix A, state some of their consequences and then look at how they are related.

1. The column space (or equivalently the range) of A is all linear combinations of the columns of A. We denote this by $\mathcal{R}(A)$.
   - By definition (because it contains all linear combinations and is thus closed under addition and scalar multiplication) the column space is a subspace of $\mathbb{R}^m$.
   - An equivalent statement to $A\vec{x} = \vec{b}$ being solvable is that $\vec{b}$ is in the range or column space of A.
2. The null space of A, denoted $\mathcal{N}(A)$ is the set of all vectors $\vec{z} \in \mathbb{R}^n$ such that $A\vec{z} = \vec{0}$.
   - The null space is a subspace of $\mathbb{R}^n$ because it consists of vectors in $\mathbb{R}^n$

and is closed under addition and scalar multiplication.

$$A\vec{y} = 0, A\vec{z} = 0 \implies A(\alpha\vec{y} + \beta\vec{z}) = \alpha A\vec{y} + \beta A\vec{z} = 0$$

3. The row space of A is the span of the rows of A and is thus a subspace of $\mathbb{R}^n$.

   - The row space of A is the same as the column space of A^T so it is often denoted $\mathcal{R}(A^T)$.

4. The null space of A^T, $\mathcal{N}(A^T)$, is a subspace of $\mathbb{R}^n$ and consists of all $\vec{z} \in \mathbb{R}^n$ such that $A^T\vec{z} = \vec{0}$. This space is often call the left null space of A because if we take the transpose of $A^T\vec{z} = \vec{0}$ we get $\vec{z}^T A = \vec{0}$.

- We call the rank of a matrix the number of the number of linearly independent rows or columns of A (they are the same), i.e., the rank is the dimension of the range and it is also the dimension of the row space of A.

- The dimensions of each of these spaces are related and are given in the following theorem. It is so important that it is called the Fundamental Theorem of Linear Algebra.

Fundamental Theorem of Linear Algebra, Part I. Let A be an $m \times n$ matrix. Then the following conditions hold.

(i) The $\mathcal{R}(A)$ is the column space of A and is a subspace of $\mathbb{R}^m$. The dimension, $\dim(\mathcal{R}(A))$, is the rank r and $r \leq m$.

(ii) The null space of A, $\mathcal{N}(A)$, is a subspace of $\mathbb{R}^n$ and has dimension $n - r$ where r is the rank of A

(iii) The row space of A is a subspace of $\mathbb{R}^n$ and is the column space of A^T, $\mathcal{R}(A^T)$ and has dimension r.

(iv) The $\mathcal{N}(A^T)$ is the left null space of A and is a subspace of $\mathbb{R}^m$ whose dimension is $m - r$.

- Note that two of these spaces are subspaces of $\mathbb{R}^n$ and two of $\mathbb{R}^m$. These spaces are related in another way which we review here.
- Recall that two vectors $\vec{x}, \vec{y}$ are orthogonal provided their inner product is zero. We can make an analogous definition for spaces.
- Two vector spaces V, W are orthogonal provided $\vec{v}^T \vec{w} = 0$ for any $\vec{v} \in V$ and

$\vec{w} \in W$.

- If two spaces are orthogonal then the only vector they have in common is the zero vector. If every $\vec{v} \in V$ is orthogonal to each basis vector of W then it is orthogonal to all of W because every other vector in W can be written as a linear combination of the basis vectors. Specifically if $\vec{w}_i \in W$, $i = 1, \ldots, n$ form a basis for W and $\vec{v}^T \vec{w}_i = 0$ for all i then

$$\vec{p} = \sum_{i=1}^{n} c_i \vec{w}_i \implies \vec{v}^T \vec{p} = \vec{v}^T \left(\sum_{i=1}^{n} c_i \vec{w}_i \right) = \sum_{i=1}^{n} c_i (\vec{v}^T \vec{w}_i) = \sum_{i=1}^{n} c_i (0) = 0$$

- The null space of A, $\mathcal{N}(A)$ and the row space of A are orthogonal spaces.

- The left null space of A and the column space of A, $\mathcal{R}(A)$ are orthogonal spaces.

- This says that every vector in the null space of A is perpendicular to every vector in the row space of A; however something stronger is actually true. The fact is that every vector in $\mathbb{R}^n$ which is perpendicular to the row space of A is in the null space of A; that is, the null space contains *every* vector in $\mathbb{R}^n$ which is orthogonal to row space of A. The analogous condition holds

for the left null space.

- Let V be a given subspace of $\mathbb{R}^n$. Then the set of all vectors in $\mathbb{R}^n$ which are orthogonal to V is called the orthogonal complement of V and is denoted $V^\perp$.

Fundamental Theorem of Linear Algebra, Part II.

$$\mathcal{N}(A) = \Big(\mathcal{R}(A^T)\Big)^\perp \qquad \mathcal{R}(A^T) = \Big(\mathcal{N}(A)\Big)^\perp$$

$$\mathcal{N}(A^T) = \Big(\mathcal{R}(A)\Big)^\perp \qquad \mathcal{R}(A) = \Big(\mathcal{N}(A^T)\Big)^\perp$$

- This theorem says that for an $m \times n$ matrix A we can write $\vec{x} \in \mathbb{R}^n$ as the sum of something in the null space of A plus something in its orthogonal complement, $\mathcal{R}(A^T)$

$$\vec{x} = \vec{w} + \vec{z} \quad \text{where } \vec{z} \in \mathcal{N}(A),\ \vec{w} \in \mathcal{R}(A^T)$$

Also we can write $\vec{y} \in I\!R^m$ as the sum of something in the left null space of A and something in its orthogonal complement, $\mathcal{R}(A)$

$$\vec{y} = \vec{w} + \vec{z} \quad \text{where } \vec{z} \in \mathcal{N}(A^T),\ \vec{w} \in \mathcal{R}(A)$$

- We already know that $A\vec{x} = \vec{b}$ has a solution if $\vec{b} \in \mathcal{R}(A)$. This second equality also says that $A\vec{x} = \vec{b}$ has a solution if $\vec{b}$ is orthogonal to every vector in $\mathcal{N}(A^T)$; i.e., if $A^T\vec{w} = \vec{0}$, then $\vec{b}^T\vec{w} = 0$.
- We already know that $A\vec{x} = \vec{b}$ has a solution if the dimension of $\mathcal{N}(A)$ is zero. The first equality also says that $A\vec{x} = \vec{b}$ has a solution if the range of A^T is all of $I\!R^n$.
- In the next theorem we summarize these results for the solvability of $A\vec{x} = \vec{b}$ where A is a square matrix.

Exercise Find the dimension of each of the four fundamental spaces for the matrix

$$A = \begin{pmatrix} 2 & 3 & -1 \\ 0 & 4 & 6 \end{pmatrix}$$

Give a basis for each space. What is the rank of A?

Exercise Find the dimension of each of the four fundamental spaces for the matrix

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

Give a basis for each space. What is the rank of A? (Hint: row reduce matrix for row space)

Exercise For the matrix A in the previous exercise demonstrate that $\mathcal{N}(A)$ is orthogonal to $\mathcal{R}(A^T)$. Also demonstrate that $\mathcal{N}(A^T)$ is orthogonal to the range of A.

Let A be an $n \times n$ invertible matrix. Then the following statements are equivalent.

- A^{-1} exists
- the determinant of A, $\det A$, is $\neq 0$
- the solution to $A\vec{x} = \vec{b}$ is unique
- the only solution to $A\vec{x} = \vec{0}$ is $\vec{x} = \vec{0}$
- the rank of A is n; the rank of A^T is n
- the dimension of the column space of A is n; the column space is a basis for $\mathbb{R}^n$
- the dimension of the row space of A is n; ; the row space is a basis for $\mathbb{R}^n$
- the null space of $A = \{\vec{0}\}$
- the dimension of the null space of A is 0
- the dimension of the left null space of A is 0
- A has no zero eigenvalues

We will use these statements freely so you should review any that you are not familiar with.

With our definitions and basic facts out of the way, we now turn to solving linear systems.

Solving linear systems

Problem Given an $n \times n$ matrix A and an n-vector $\vec{b}$, determine $\vec{x} \in I\!R^n$ such that $A\vec{x} = \vec{b}$

Recall that the inverse of an $n \times n$ matrix (if it exists) is a matrix, denoted A^{-1} such that

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

where I denotes the $n \times n$ identity matrix.

This means that if we have A^{-1} in hand, then we can find the solution with a single matrix times vector multiplication. This is very useful for analysis but we will see that it does NOT result in a good algorithm to implement.

There are two main classes of algorithms for solving $A\vec{x} = \vec{b}$.

Direct Methods: If these methods are implemented using exact arithmetic then they determine the exact solution in a finite number of steps; i.e., we have exact formulas to follow to obtain the solution.

Iterative Methods: These methods start with an initial guess $\vec{x}^0$ for $\vec{x}$ and determine a sequence of iterates $\vec{x}^k$ such that (hopefully) $\vec{x}^k \to \vec{x}$ as $k \to \infty$.

We will begin by looking at direct methods and delay iterative methods until after we have looked at the eigenvalue problem because eigenvalues play an important role in iterative methods.

Direct Methods for Solving Linear Systems

Gaussian elimination (GE) is a direct method for solving $A\vec{x} = \vec{b}$ that you were probably introduced to in algebra, even though you may not have called it that. For example, if you had three equations in three unknowns then you were taught to eliminate one unknown from one equation and that unknown plus another from a second equation. Then you solved for one unknown from the equation with only one, used it in the equation with two unknowns and finally used the last equation to complete the solution.

Example To solve the linear system

$$\begin{aligned} 2x_1 + x_2 + x_3 &= 1 \\ 4x_1 - 6x_2 &= 2 \\ -2x_1 + 7x_2 + 2x_3 &= 3 \end{aligned}$$

we can add the first and third equations to eliminate x_1 to get $8x_2 + 3x_3 = 4$. Now combining the first and the second one (multiply first by -2 and add to

second) we get $-8x_2 - 2x_3 = 0$. Adding this to $8x_2 + 3x_3 = 4$ gives $x_3 = 4$; now we have that $8x_2 = -8$ or $x_2 = -1$ and finally $2x_1 - 1 + 4 = 1$ implies $x_1 = -1$. Thus the solution is $\vec{x} = (-1, -1, 4)^T$.

This linear system can be written in matrix form as

$$\begin{pmatrix} 2 & 1 & 1 \\ 4 & -6 & 0 \\ -2 & 7 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

What we are actually doing when we eliminated variables is transforming the problem into an equivalent one that was easier to solve. For our example we transformed the linear system into

$$\begin{pmatrix} 2 & 1 & 1 \\ 0 & -8 & -2 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 4 \end{pmatrix}$$

In matrix notation, we are performing operations on $A\vec{x} = \vec{b}$ which preserved the solution, so that we transformed it into an upper triangular system $U\vec{x} = \vec{c}$.

Why is an upper triangular system easy to solve? Because it can be solved directly by noting that the last equation has only one unknown x_n; we solve for it and use it in the next to last equation which only has the unknowns x_n and x_{n-1}; we continue in this manner. This process is called a back solve for obvious reasons.

This is an example of a direct method and we can write explicit equations for the solution. Consider a general $n \times n$ upper triangular matrix U where we want to find the solution of the linear system $U\vec{x} = \vec{b}$

$$\begin{pmatrix} u_{11} & u_{12} & u_{13} & u_{14} & \cdots & u_{1n} \\ 0 & u_{22} & u_{23} & u_{24} & \cdots & u_{2n} \\ 0 & 0 & u_{33} & u_{34} & \cdots & u_{3n} \\ \vdots & \vdots & & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & u_{n-1,n-1} & u_{n-1,n} \\ 0 & 0 & 0 & \cdots & 0 & u_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_{n-1} \\ b_n \end{pmatrix}$$

To write the equations for x_i we note that $U\vec{x}$ is itself a vector so we just equate

entries on the vectors of the right and left sides of the equation. We have

$$x_n = \frac{b_n}{u_{nn}}$$

Then to obtain x_{n-1} we equate the $n-1$ component

$$u_{n-1,n-1}x_{n-1} + u_{n-1,n}x_n = b_{n-1} \implies x_{n-1} = \frac{b_{n-1} - u_{n-1,n}x_n}{u_{n-1,n-1}}$$

For x_{n-2} we equate the $n-2$ component

$$u_{n-2,n-2}x_{n-2} + u_{n-2,n-1}x_{n-1} + u_{n-2,n}x_n = b_{n-2}$$

$$\implies x_{n-2} = \frac{b_{n-2} - u_{n-2,n-1}x_{n-1} - u_{n-2,n}x_n}{u_{n-2,n-2}}$$

In general, we can find the ith component of $\vec{x}$ for $i < n$ by

$$x_i = \frac{b_i - \sum_{j=i+1}^{n} u_{i,j}x_j}{u_{ii}}$$

Back solve algorithm: Given an $n \times n$ nonsingular upper triangular matrix U with entries u_{ij} and an n-vector $\vec{b}$ with components b_i then the solution of $U\vec{x} = \vec{b}$ is given by the following algorithm.

Set $x_n = \dfrac{b_n}{u_{nn}}$

For $i = n-1, n-2, \ldots, 1$

$$x_i = \frac{b_i - \sum_{j=i+1}^{n} u_{i,j} x_j}{u_{ii}}$$

Exercise How do we know that $u_{ii} \neq 0$ for all i?

To implement this algorithm the matrix U and the right hand side $\vec{b}$ must be provided as input. Do we need to create a new vector $\vec{x}$? If you look at the equations carefully you will see that once we use b_i in the equation for x_i, it is never used again. This means that we can overwrite b_i. So to implement the algorithm our pseudo code description would look some thing like this:

$$b_n \leftarrow \frac{b_n}{u_{nn}}$$

For $i = n-1, n-2, \ldots, 1$

$$b_i \leftarrow \frac{b_i - \sum_{j=i+1}^{n} u_{i,j} b_j}{u_{ii}}$$

A useful thing for comparing algorithms is to compute the number of arithmetic operations that an algorithm requires. Clearly this will be a function of n. The following table gives a count for the number of additions and multiplications for each step.

step	# mult/div	# add/sub
n	1	0
$n-1$	2	1
$n-2$	3	2
$n-3$	4	3
$\vdots$	$\vdots$	$\vdots$
1	n	$n-1$
Total:	$\sum_{i=1}^{n} i$	$\sum_{i=1}^{n-1} i$

We want to get our sum in terms of an expression in n. Recall from calculus that

$$\sum_{i=1}^{m} i = \frac{m(m+1)}{2} \qquad \sum_{i=1}^{m} i^2 = \frac{m(m+1)(2m+1)}{6}$$

Thus we see that the method takes

$$\frac{n(n+1)}{2} = \frac{n^2}{2} + \frac{n}{2}$$ multiplications/divisions

and

$$\frac{n(n-1)}{2} = \frac{n^2}{2} - \frac{n}{2}$$ additions/subtractions

As n grows the dominant term is $\frac{n^2}{2}$ in each expression. We say that the method is order n^2 and write $\mathcal{O}(n^2)$. This notation means the growth in n is a constant times n^2.

Exercise If a method is $\mathcal{O}(n^2)$, then if n is doubled, is the work doubled? If not, how does it change?

Now we know how to solve the system $A\vec{x} = \vec{b}$ once we transform it to upper triangular form. So all we have to do is figure out how to systematically perform this transformation to upper triangular form *while preserving the solution*.

You probably remember how to do this with equations. We simply use what are called *elementary row operations*. If we have a system of linear equations and we multiply any *equation* by a constant, then the solution is preserved. If we multiply one equation by a constant and add to another, then the solution is preserved. We now want to describe this in terms of elementary transformation matrices /Gauss transformation matrices.

A Gauss transformation matrix $\mathcal{M}^k$ is a unit lower triangular matrix which has

nonzero entries (except for the ones on the diagonal) only in the kth column below the diagonal. We will use it to perform elementary row operations on a given matrix. For example,

$$\mathcal{M}^2 = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & m_{32}^2 & 1 & 0 & \cdots & 0 \\ 0 & m_{42}^2 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \ddots & \\ 0 & m_{n2}^2 & 0 & 0 & \cdots & 1 \end{pmatrix}$$

Example Lets return to our linear system

$$\begin{pmatrix} 2 & 1 & 1 \\ 4 & -6 & 0 \\ -2 & 7 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

which we transformed into the upper triangular system

$$\begin{pmatrix} 2 & 1 & 1 \\ 0 & -8 & -2 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 4 \end{pmatrix}$$

We want to describe our procedure in terms of premultiplying by Gauss transformation matrices. The first step was to eliminate x_1 from the second and third equations by taking multiples of the first equation and adding. We want to construct a Gauss transformation matrix $\mathcal{M}^1$ which accomplishes this when we premultiply A by it:

$$\mathcal{M}^1 A = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 \\ 4 & -6 & 0 \\ -2 & 7 & 2 \end{pmatrix} = \begin{pmatrix} 2 & 1 & 1 \\ 0 & -8 & -2 \\ 0 & 8 & 3 \end{pmatrix}$$

How did we choose the entries of the Gauss transformation matrix $\mathcal{M}^1$? We chose the entries so that we could "zero out" the entries in A below the main diagonal in the first column, i.e., our first step towards converting the system to an upper triangular one. Of course if we multiply the left hand side by $\mathcal{M}^1$ we

must do so to the right hand side. Specifically we have

$$\mathcal{M}^1_{21} = \frac{-a_{21}}{a_{11}} \qquad \mathcal{M}^1_{1,3} = \frac{-a_{13}}{a_{11}}$$

Note that all entries in the modified A excluding the first row (and first column) must be computed.

For the last step choose $\mathcal{M}^2$ such that $\mathcal{M}^2\mathcal{M}^1 A$ is upper triangular. Using the same reasoning as before

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 \\ 0 & -8 & -2 \\ 0 & 8 & 3 \end{pmatrix} = \begin{pmatrix} 2 & 1 & 1 \\ 0 & -8 & -2 \\ 0 & 0 & 1 \end{pmatrix}$$

When we form $\mathcal{M}^2\mathcal{M}^1\vec{b}$ then we have successfully transformed the original system into an upper triangular system which can be solved by our back solve algorithm. We have $\mathcal{M}^2\mathcal{M}^1\vec{b} = (1, 0, 4)^T$ so we have transformed our system into an equivalent upper triangular system.

For a general $n \times n$ matrix A we construct Gauss transformation matrices $\mathcal{M}^k$,

$k = 1, \ldots, p$ such that

$$\mathcal{M}^p \mathcal{M}^{p-1} \cdots \mathcal{M}^2 \mathcal{M}^1 A = U \qquad p \leq n-1$$

Now each $\mathcal{M}^k$ is unit lower triangular and is thus invertible. In fact its inverse is easy to write down. For example, the inverse of $\mathcal{M}^1$ in the example above is just

$$\mathcal{M}^1 = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \qquad (\mathcal{M}^1)^{-1} \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}$$

All we have to do to find the inverse of $\mathcal{M}^k$ is multiply the entries below the diagonal in the kth column by -1. (You should convince yourselves of this!) So another way to write our expression is

$$A = [\mathcal{M}^1]^{-1} [\mathcal{M}^2]^{-1} \cdots [\mathcal{M}^p]^{-1} U$$

Now the next thing you should convince yourselves of is that when we multiply two unit lower triangular matrices then the result is a unit lower triangular matrix. So when we transform our original system into an upper triangular system, we can view the process as factoring A into the product of a unit lower triangular matrix and an upper triangular matrix.

For our example above we have

$$\begin{pmatrix} 2 & 1 & 1 \\ 4 & -6 & 0 \\ -2 & 7 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 \\ 0 & -8 & -2 \\ 0 & 0 & 1 \end{pmatrix}$$

Note how easy it was to find the product of $[\mathcal{M}^1]^{-1}[\mathcal{M}^2]^{-1}$.

We now ask ourselves if it is always possible to construct the Gauss transformation matrices which transform an invertible matrix into an upper triangular one. Recall our formulas for $\mathcal{M}^1$

$$\left(\mathcal{M}^1\right)_{i1} = -\frac{a_{i1}}{a_{11}}$$

so if $a_{11} = 0$, then our procedure fails. Can this ever happen for an invertible matrix?

In the construction of $\mathcal{M}^1$ the term a_{11} is called a pivot. When we construct $\mathcal{M}^2$ we need the pivot $a^1_{22} \neq 0$ where the superscript 1 denotes the fact that it is an element of A after the first step; i.e., of $\mathcal{M}^1 A$. At any point in the procedure we may hit a zero pivot. If we are performing the calculations by hand, we simply interchange equations. We can do the same here (i.e., interchange rows

of the matrix.) We can describe this procedure by premultiplying the matrix by a permutation matrix.

A permutation matrix is a matrix formed by rearranging the rows (or columns) of the identity matrix.

Exercise Find the permutation matrix P which interchanges the first and second rows of

$$A = \begin{pmatrix} 0 & 1 & 1 \\ 4 & -6 & 0 \\ -2 & 7 & 2 \end{pmatrix}$$

In general, we now have

$$\mathcal{M}^p\mathcal{P}^p\mathcal{M}^{p-1}\mathcal{P}^{p-1}\cdots\mathcal{M}^2\mathcal{P}^2\mathcal{M}^1\mathcal{P}^1A = U \qquad p \leq n-1$$

where $\mathcal{P}^k$ denotes the permutation matrix at the kth step.

We have seen that not all invertible matrices have an LU factorization. However, the following is true.

LU Factorization Theorem Let A be an $n \times n$ matrix. There exists a permutation P such that

$$PA = LU$$

where L is a unit lower triangular matrix and U is an upper triangular matrix. Once P is specified, L and U are unique.

If we are given an LU factorization of a matrix A, can we use it to solve $A\vec{x} = \vec{b}$?

The answer is yes.

$$A = LU \implies LU\vec{x} = \vec{b}$$

so we can first solve the lower triangular system $L\vec{y} = \vec{b}$ and then the upper triangular system $U\vec{x} = \vec{y}$. Now we have seen that an upper triangular system requires $\mathcal{O}(n^2)$ operations and it's not hard to believe that a lower triangular system requires the same. Shortly we will demonstrate that the factorization

$A = LU$ requires $\mathcal{O}(n^3)$ operations so a back solve and a forward solve are "cheap" compared with obtaining the factorization.

Why would we want to solve a linear system by an LU factorization instead of the standard GE approach? On paper they are equivalent. If we only want to solve one linear system, then it doesn't matter which approach we use. If we want to solve $AX = B$ where B is $n \times p$, i.e., we have p systems with the same coefficient matrix, then it doesn't matter which routine we use. However, in practice, we often have to solve $A\vec{x} = \vec{b}$ and then use the solution $\vec{x}$ to form the next right hand side so we don't have all the right hand sides in hand at one time. In this case, LU factorization can be advantageous because we factor $A = LU$ once ($\mathcal{O}(n^3)$ operations) and then for each right hand side we do one forward solve and one back solve so we do an additional $\mathcal{O}(pn^2)$ operations yielding a total of $\mathcal{O}(n^3)+\mathcal{O}(pn^2)$. This should be compared to p applications of GE which is $\mathcal{O}(pn^3)$.

Instead of finding L from the Gauss transformation matrices, we can find L and U directly by simply writing $A = LU$ and equating entries; thus we will have explicit equations which we can program. Consider the matrix equation $A = LU$

written as

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \ell_{21} & 1 & 0 & \cdots & 0 \\ \ell_{31} & \ell_{32} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \ell_{n1} & \ell_{n2} & \ell_{n3} & \cdots & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ 0 & u_{22} & u_{23} & \cdots & u_{2n} \\ 0 & 0 & u_{33} & \cdots & u_{3n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & u_{nn} \end{pmatrix}$$

Now equating the $(1,1)$ entry gives

$$a_{11} = 1 \cdot u_{11} \implies u_{11} = a_{11}$$

In fact, if we equate each entry of the first row of A, i.e., a_{1j} we get

$$u_{1j} = a_{1j} \quad \text{for } j = 1, \ldots, n.$$

Now we move to the second row and look at the (2,1) entry to get $a_{21} = \ell_{21} \cdot u_{11}$ implies $\ell_{21} = a_{21}/u_{11}$. Now we can determine the remaining terms in the first column of L by

$$\ell_{i1} = a_{i1}/u_{11} \quad \text{for } i = 2, \ldots, n.$$

We now find the second row of U. Equating the (2,2) entry gives $a_{22} = \ell_{21}u_{12} + u_{22}$ implies $u_{22} = a_{22} - \ell_{21}u_{12}$. In general

$$u_{2j} = a_{2j} - \ell_{21}u_{1j} \quad \text{for } j = 2, \ldots, n.$$

For the second column of L we have

$$a_{i2} = \ell_{i1}u_{12} + \ell_{i2}u_{22} \quad i = 3, \ldots, n$$

so that

$$\ell_{i2} = \frac{a_{i2} - \ell_{i1}u_{12}}{u_{22}} \quad i = 3, \ldots, n$$

Continuing in this manner, we get the following algorithm.

Let A be a given $n \times n$ matrix. Then if no pivoting is needed, the LU factorization of A into a unit lower triangular matrix L with entries ℓ_{ij} and an upper triangular matrix U with entries u_{ij} is given by the following algorithm for LU factorization.

Set $u_{1j} = a_{1j}$ for $j = 1, \ldots, n$

For $k = 1, 2, 3 \ldots, n-1$

$$\ell_{i,k} = \frac{a_{i,k} - \sum_{m=1}^{k-1} \ell_{im} u_{m,k}}{u_{k,k}} \quad \text{for } i = k+1, \ldots, n$$

$$u_{k+1,j} = a_{k+1,j} - \sum_{m=1}^{k} \ell_{k+1,m} u_{m,j} \quad \text{for } j = k, \ldots, n$$

Note that this algorithm clearly demonstrates that you can NOT find all of L and then all of U or vice versa. One must determine a row of U, then a column

of L, then a row of U, etc.

How would you implement this algorithm? We always need to be aware of storage, especially when dealing with matrices; remember that it takes n^2 locations to store an $n \times n$ matrix so if n is very large we may not have storage for several matrices.

If you look at the equations carefully you will see that once an element a_{ij} of A appears in an equation for either ℓ_{ij} or u_{ij}, it never appears again. That means that we can overwrite A with L and U. We do this by overwriting the diagonal and upper portion of A with U and the lower portion with L except for its diagonal which we don't need because we know that it is 1. So our algorithm description could look like the following.

Input: an $n \times n$ matrix A and n

Output: the $n \times n$ matrix A overwritten with L and U

Set $a_{1j} = a_{1j}$ for $j = 1, \ldots, n$

For $k = 2, 3 \ldots, n$

$$\text{for } j = k, \ldots, n \text{ set } a_{k,j} = \frac{a_{k,j} - \sum_{m=1}^{k-1} a_{km} a_{m,j}}{a_{kk}}$$

$$\text{for } i = k, \ldots, n \text{ set } a_{i,k} = \frac{a_{i,k} - \sum_{m=1}^{k-1} a_{im} a_{m,k}}{a_{k,k}}$$

When we use this decomposition to solve the linear system $A\vec{x} = \vec{b}$ by solving $L\vec{y} = \vec{b}$ and $U\vec{x} = \vec{y}$ we really don't have to set up new vectors $\vec{y}$ and $\vec{x}$. When we solve for $\vec{y}$ we overwrite it on $\vec{b}$ and likewise when we solve for $\vec{x}$ we overwrite the solution onto $\vec{b}$ (which is now $\vec{y}$).

When we implement this algorithm, we need to incorporate a strategy that checks to see if any rows need to be interchanged. Is it enough to check to make sure we don't have a zero in the denominator?

Because we want to implement this algorithm on a computer with finite precision

arithmetic, checking to see if an entry is zero is not enough. Consider the following example.

Example Consider the linear system

$$\begin{aligned} 0.001x_1 + 1.0x_2 &= 1.00 \\ 1.00x_1 + 1.00x_2 &= 2.00 \end{aligned}$$

whose exact solution is $x_1 = 1000/999$, $x_2 = 998/999$. Assume that we carry out GE on a machine which uses two digit arithmetic. This means that we store each number as $\pm 0.d_1d_2 \times 10^{\beta}$. We first perform GE without row interchanges. To eliminate x_1 from the second equation we multiply the first by -1000 and add to get the coefficient of x_2 as -1000+1=-999 which is just -1000$= -0.10 \cdot 10^4$ in our finite precision machine. Similarly the right hand side is just 2-1000 = -998 which is also -1000 on our machine. Thus $x_2 = 1.0$. Substituting back into the first equation gives

$$0.001x_1 + 1.0 = 1.00 \implies x_1 = 0$$

which of course is not near the exact solution. If we interchange equations we

have

$$\begin{aligned} 1.00x_1 + 1.00x_2 &= 2.00 \\ 0.001x_1 + 1.0x_2 &= 1.00 \end{aligned}$$

and the second equation becomes $1 \cdot x_2 = 1$ which implies $x_2 = 1$ but now substituting into the first equation we get $x_1 = 1$ also which is an accurate solution with only two digits of accuracy.

This example demonstrates that it is not enough to look for a nonzero pivot. A partial pivoting strategy looks down a column from the (i, i) entry and below to find the entry with the largest magnitude and then exchanges those two rows. A full pivoting strategy seeks the largest entry in magnitude in the remaining block of the matrix; for example at the second step we have a^1_{22} as a potential pivot and we search among all entries a^1_{ij} from $i, j = 2, n$ for the largest entry in magnitude and then interchange rows or columns. Typically a partial pivoting strategy is adequate.

We now want to know if we use a partial pivoting strategy does that solve all of our problems? Can we get in trouble in any other way? The next example demonstrates another problem.

Example Consider the linear system

$$\begin{aligned} 10x_1 + 10000x_2 &= 10000 \\ x_1 + x_2 &= 2 \end{aligned}$$

whose exact solution is $x_1 = 1000/999$, $x_2 = 998/999$ because the system was obtained by multiplying the first equation in the previous example by 10,000. Now our partial pivoting strategy says that we don't have to interchange rows. If we solve this using two digit arithmetic as before, we get $x_2 = 1$ and then $10x_1 + 10,000 = 10,000$ or $x_1 = 0$. We are in trouble again. This time the difficulty is that the matrix is not properly scaled, i.e., the entries vary wildly.

To write a multi-purpose LU factorization routine one should include a partial pivoting strategy along with a row scaling strategy. We first define scale factors s_i for each row $i = 1, \ldots, n$ of A by

$$s_i = \sum_{j=1}^{n} |a_{ij}| .$$

So for our example, $s_1 = 10,010$ (so in our precision 10,000) and $s_2 = 2$. Then instead of finding the largest entry in the column we find the largest entry scaled

by the corresponding s_i. So in our example we have $10/10,000$ and $1/2$ so we should interchange rows and thus our problem is solved accurately.

We can do an operation count for this algorithm. We summarize in the table below.

	mult/div	add/sub
$u_{1j}, j = 1, \ldots, n$	0	0
$\ell_{i1}, i = 2, \ldots, n$	(n-1)1	0
$u_{2j}, j = 2, \ldots, n$	(n-1)1	(n-1)1
$\ell_{i2}, i = 3, \ldots, n$	(n-2)(2)	(n-1)1
$u_{3j}, j = 3, \ldots, n$	(n-2)(2)	(n-2)(2)
$\ell_{i3}, i = 4, \ldots, n$	(n-3)3	(n-2)(2)
$\vdots$		
$u_{(n-1)j}, j = n-1, n$	2(n-2)	2(n-2)
$\ell_{n(n-1)}$	(1)(n-2)	1(n-2)
u_{nn}	1(n-1)	1(n-1)

Summing these we see that it takes a total of

$$\sum_{i=1}^{n-1}(i)(n-i) = n\sum_{i=1}^{n-1} i - \sum_{i=1}^{n-1} i^2$$

multiplications/divisions for U and a similar number of additions/subtractions. Combining these we see that we have

$$n\sum_{i=1}^{n-1} i - \sum_{i=1}^{n-1} i^2 = n\frac{(n)(n-1)}{2} - \frac{(n-1)(2n-1)(n)}{6} = \frac{n^3}{6} + l.o.t.$$

where we have used our formulas from calculus for summing integers and their squares. Because L requires the same number of operations in n^3 we see that the work required is $\mathcal{O}(\frac{n^3}{3})$ multiplications/divisions and a like number of additions/subtractions.

There are many variants of LU factorization; for example

$A = LU$ where L is lower triangular and U is unit upper triangular

$A = LDU$ where L is unit lower triangular, U is unit upper triangular and D is diagonal

$$A = LL^T \quad \text{where } L \text{ is lower triangular}$$
$$A = LDL^T \quad \text{where } L \text{ is unit lower triangular, } D \text{ diagonal}$$

This last two decompositions hold if A is symmetric and positive definite; the first is called the Cholesky decomposition.

The Cholesky decomposition is so important that we state the following theorem because many matrices from discretizing PDEs result in symmetric, positive definite matrices. Note that it is "if and only if" so that the algorithm will give us a test for whether a symmetric matrix is positive definite.

Cholesky decomposition. Let A be an $n \times n$ symmetric matrix. Then $A = LL^T$ if and only if A is positive definite. Moreover, the decomposition is unique if the diagonal entries of A are chosen to be positive.

To obtain the equations for the Cholesky decomposition we once again equate

matrices. We have

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix} = \begin{pmatrix} \ell_{11} & 0 & 0 & \cdots & 0 \\ \ell_{21} & \ell_{22} & 0 & \cdots & 0 \\ \ell_{31} & \ell_{32} & \ell_{33} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \ell_{n1} & \ell_{n2} & \ell_{n3} & \cdots & \ell_{nn} \end{pmatrix} \begin{pmatrix} \ell_{11} & \ell_{21} & \ell_{31} & \cdots & \ell_{n1} \\ 0 & \ell_{22} & \ell_{32} & \cdots & \ell_{n2} \\ 0 & 0 & \ell_{33} & \cdots & \ell_{n3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \ell_{nn} \end{pmatrix}$$

If we equate the (1,1) entries of the matrices on each side we have

$$a_{11} = \ell_{11}^2 \implies \ell_{11} = \pm\sqrt{a_{11}}$$

Note that the first step fails if $a_{11} < 0$. If we choose our sign when we take the square root the decomposition is unique; here we take $+\sqrt{a_{11}}$. The remaining first column of L can be found by

$$\ell_{i1} = \frac{a_{i1}}{\ell_{11}}, \quad i = 2, 3, \ldots, n$$

For the (2,2) entry we again find that we need to take a square root

$$\ell_{22} = \sqrt{a_{22} - \ell_{21}^2}$$

It's not obvious from the fact that A is positive definite that the quantity under the square root is non-negative but this can be demonstrated. Continuing in this manner we get equations for our LL^T decomposition.

For $k = 1, 2, \ldots, n$

$$l_{kk} = \left[a_{kk} - \sum_{j=1}^{k-1} \ell_{kj}^2 \right]^{1/2}$$

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

$$\ell_{ik} = \frac{a_{ik} - \sum_{j=1}^{k-1} \ell_{ij}\ell_{kj}}{\ell_{kk}}$$

As before we don't actually form a new matrix L, we simply overwrite A. We expect that it would take roughly half the storage (since the matrix is symmetric) and approximately half the operations. However this still means the algorithm is $\mathcal{O}(n^3)$ to perform the decomposition.

We will look at variants of LU in the homework.

Exercise What is the advantage of using the LDL^T decomposition over the LL^T decomposition?

So it seems that we now have algorithms for solving $A\vec{x} = \vec{b}$ so why should we look at any other methods? There are two reasons; the first is storage and the second is that these methods may not work for all matrices. How can this be the case because we have exact equations? The problem is round off. If we have a system $A\vec{x} = \vec{b}$ where small changes in the data (such as in A or $\vec{b}$) produce large changes in the solution $\vec{x}$ then we say the system is ill-conditioned. We need to determine how to quantify ill-conditioning so that we can recognize it when it happens to us. To this end, we need to first review norms.

Vector and Matrix Norms

Vector Norms

The Euclidean length of a vector which you learned in algebra (or before) is actually a norm. Recall that the Euclidean length of $\vec{x}$ is found by

$$\Big[\sum_{i=1}^{n} x_i^2\Big]^{1/2}$$

We want to generalize this concept to include other measures of a norm. We can view the Euclidean length as a map (or function) whose domain is $I\!R^n$ and whose range is all scalars i.e., $f : I\!R^n \rightarrow I\!R^1$. What properties does this Euclidean length have? We know that the length is always ≥ 0 and only $=$0 if the vector is identically zero. We know what multiplication of a vector by a scalar k does to the length; i.e., it changes by the length by $|k|$. Also, from the triangle inequality we know that the length of the sum of two vectors is always $\leq$ the sum of the two lengths. We combine these properties into a formal definition of a vector.

A vector norm, denoted $\|\vec{x}\|$, is a map from $\mathbb{R}^n$ to $\mathbb{R}^1$ which has the properties

1. $\|\vec{x}\| \geq 0$ and $= 0$ only if $\vec{x} = \vec{0}$
2. $\|k\vec{x}\| = |k|\|\vec{x}\|$
3. $\|\vec{x}\| + \|\vec{y}\| \leq \|\vec{x}\| + \|\vec{y}\|$ (triangle inequality)

There are other ways to measure the length of vectors. All we have to do is find a map which satisfies the above three conditions; however, practically it should be useful. Three of the most useful vector norms are defined below.

The most common vector norms in $\mathbb{R}^n$ are:

1. Euclidean norm, denoted $\|\vec{x}\|_2$ and defined by

$$\|\vec{x}\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2} = \left(\vec{x}^T \vec{x}\right)^{1/2}$$

2. Max or infinity norm, denoted $\|\vec{x}\|_\infty$ and defined by

$$\|\vec{x}\|_\infty = \max_{1 \le i \le n} |x_i|$$

3. one-norm, denoted $\|\vec{x}\|_1$ and defined by $\|\vec{x}\|_1 = \sum_{i=1}^{n} |x_i|$

The Euclidean and one norms are special cases of the general family of p-norms

$$\|\vec{x}\|_p = \left(\sum_{i=1}^{n} x_i^p\right)^{1/p}$$

Exercise Determine $\|\vec{x}\|_1$, $\|\vec{x}\|_2$ and $\|\vec{x}\|_\infty$ for the given vector

$$\vec{x} = (-3, 2, 4, -7)^T$$

Exercise Sketch the unit ball for each norm, i.e., sketch all points in $\mathbb{R}^2$ such that

$$\{(x_1, x_2) \quad \text{such that} \quad \|\vec{x}\|_p = 1\} \quad \text{for } p = 1, 2, \infty$$

Exercise If A is an orthogonal matrix, show that $\|A\vec{x}\|_2 = \|\vec{x}\|_2$.

Many times we use a norm to measure the length of an error vector, i.e., we will associate a number with a vector. In the previous example we saw that different norms give us different numbers for the same vector. How different can these numbers be? Each norm actually measures a different attribute of a norm. However, should we be worried that if we can show a particular norm of the error goes to zero, then the other norms might not go to zero too? The following definition helps us to quantify this concept.

Let $\|\cdot\|_\alpha$ and $\|\cdot\|_\beta$ denote any two vector norms. Then these norms are norm-equivalent if there exists constants C_1, C_2 greater than zero such that

$$C_1\|\vec{x}\|_\beta \leq \|\vec{x}\|_\alpha \leq C_2\|\vec{x}\|_\beta \quad \text{for all } \vec{x}$$

Note that if this inequality holds, we also have the equivalent statement

$$\frac{1}{C_2}\|\vec{x}\|_\alpha \leq \|\vec{x}\|_\beta \leq \frac{1}{C_1}\|\vec{x}\|_\alpha \quad \text{for all } \vec{x}$$

If two norms are norm-equivalent and we have that $\|\vec{x}\|_\beta \to 0$ then clearly $\|\vec{x}\|_\alpha \to 0$.

We claim that any pair of our three vector norms are norm-equivalent. We demonstrate that this is true for one set here and you should convince yourselves of the other pairs.

Example The norms $\|\vec{x}\|_\infty$ and $\|\vec{x}\|_2$ are equivalent. We have

$$\|\vec{x}\|_2^2 = \sum_{i=1}^{n} x_i^2 \geq \max |x_i|^2 = \|\vec{x}\|_\infty^2$$

so $C_1 = 1$. Also

$$\|\vec{x}\|_2^2 = \sum_{i=1}^{n} x_i^2 \leq n \max |x_i|^2 = n\|\vec{x}\|_\infty^2$$

so $C_2 = \sqrt{n}$.

$$\|\vec{x}\|_\infty \leq \|\vec{x}\|_2 \leq \sqrt{n}\|\vec{x}\|_\infty \quad \text{for all } \vec{x}$$

Exercise Demonstrate that the norms $\|\vec{x}\|_\infty$ and $\|\vec{x}\|_1$ are equivalent.

Our next goal is to associate a matrix with a number; i.e., we want to define a matrix norm.

Matrix Norms

Now we turn to associating a number to each matrix. We could choose our norms analogous to the way we did for vector norms; e.g., we could associate to our matrix the number $\max_{ij} |a_{ij}|$. However, this is actually not very useful because remember our goal is to study linear systems $A\vec{x} = \vec{b}$.

The general definition of a matrix norm is a map from all $m \times n$ matrices to $\mathbb{R}^1$ which satisfies the properties

(i) $\|A\| \geq 0$ and $= 0$ only if $a_{ij} = 0$ for all i, j.

(ii) $\|kA\| = |k|\|A\|$ for scalars k

(iii) $\|AB\| \leq \|A\|\|B\|$

(iv)$\|A + B\| \leq \|A\| + \|B\|$

You should recognize most of these as being analogous to the properties of a vector norm.

However, the most useful matrix norms are those that are generated by a vector norm; again the reason for this is that we want to solve $A\vec{x} = \vec{b}$ so if we take the norm of both sides of the equation it is a vector norm and on the left hand side we have the norm of a matrix times a vector.

We will define an induced matrix norm as the largest amount any vector is magnified when multiplied by that matrix, i.e.,

$$\|A\| = \max_{\vec{x} \in I\!R^n \ \ \vec{x} \neq 0} \frac{\|A\vec{x}\|}{\|\vec{x}\|}$$

Note that all norms on the right hand side are vector norms. We will denote a vector and matrix norm using the same notation; the difference should be clear from the argument. We say that the vector norm on the right hand side induces the matrix norm on the left. Note that sometimes the definition is written in an equivalent way as

$$\|A\| = \sup_{\vec{x} \in I\!R^n \ \ \vec{x} \neq 0} \frac{\|A\vec{x}\|}{\|\vec{x}\|}$$

A very useful inequality is

$$\|A\vec{x}\| \leq \|A\|\|\vec{x}\| \quad \text{for any induced norm}$$

Why is this true?

$$\|A\| = \max_{\vec{x}\in\mathbb{R}^n \ \vec{x}\neq 0} \frac{\|A\vec{x}\|}{\|\vec{x}\|} \geq \frac{\|A\vec{y}\|}{\|\vec{y}\|} \implies \|A\vec{y}\| \leq \|A\|\|\vec{y}\|$$

for any $\vec{y} \neq 0$.

The problem with this definition is that it doesn't tell us how to compute a matrix norm for a general matrix A. The following result gives us a way to calculate matrix norms induced by the ℓ_∞ and ℓ_1 norms; the matrix norm induced by ℓ_2 norm will be addressed later after we have introduced eigenvalues.

Let A be an $m \times n$ matrix. Then

$$\|A\|_\infty = \max_{1 \le i \le m} \left[\sum_{j=1}^{n} |a_{ij}| \right] \quad \text{(max absolute row sum)}$$

$$\|A\|_1 = \max_{1 \le j \le n} \left[\sum_{i=1}^{m} |a_{ij}| \right] \quad \text{(max absolute column sum)}$$

Exercise Determine $\|A\|_\infty$ and $\|A\|_1$ where

$$A = \begin{pmatrix} 1 & 2 & -4 \\ 3 & 0 & 12 \\ -20 & -1 & 2 \end{pmatrix}$$

Proof (part i) We will prove that $\|A\|_\infty$ is the maximum row sum (in absolute

value). We will do this by proving that

$$\|A\|_\infty \leq \max_{1\leq i\leq m}\Big[\sum_{j=1}^{n}|a_{ij}|\Big] \quad \text{and then showing} \quad \|A\|_\infty \geq \max_{1\leq i\leq m}\Big[\sum_{j=1}^{n}|a_{ij}|\Big]$$

First recall that if $A\vec{x}=\vec{b}$ then

$$b_i=\sum_{j=1}^{n}a_{ij}x_j \implies \|\vec{b}\|_\infty=\max_i|b_i|=\max_i|\sum_{j=1}^{n}a_{ij}x_j|$$

For the first inequality we know that by definition

$$\|A\|_\infty=\max_{\vec{x}\in I\!R^n}\frac{\|A\vec{x}\|_\infty}{\|\vec{x}\|_\infty}$$

Now lets simplify the numerator to get

$$\|A\vec{x}\|_\infty=\max_i\Big|\sum_{j=1}^{n}a_{ij}x_j\Big|\leq\max_i\sum_{j=1}^{n}|a_{ij}|\;|x_j|\leq\|\vec{x}\|_\infty\max_i\sum_{j=1}^{n}|a_{ij}|$$

Thus the ratio reduces to

$$\frac{\|A\vec{x}\|_\infty}{\|\vec{x}\|_\infty}\leq\frac{\|\vec{x}\|_\infty\max_i\sum_{j=1}^{n}|a_{ij}|}{\|\vec{x}\|_\infty}=\max_i\sum_{j=1}^{n}|a_{ij}|$$

and hence

$$\|A\|_\infty = \max_{\vec{x}\in I\!R\,\vec{x}\neq 0} \frac{\|A\vec{x}\|_\infty}{\|\vec{x}\|_\infty} \leq \max_i \sum_{j=1}^{n} |a_{ij}| \,.$$

Now for the second inequality we know that

$$\|A\|_\infty \geq \frac{\|A\vec{y}\|_\infty}{\|\vec{y}\|_\infty}$$

for any $\vec{y} \in I\!R^n$ because equality in the definition holds here for the maximum of this ratio. So now we will choose a particular $\vec{y}$ and we will construct it so that it has $\|\vec{y}\|_\infty = 1$. First let p be the row where A has its maximum row sum (if there are two rows, take the first), i.e.,

$$\max_i \sum_{j=1}^{n} |a_{ij}| = \sum_{j=1}^{n} |a_{pj}|$$

Now we will take the entries of $\vec{y}$ to be ± 1 so its infinity norm is one. Specifically we choose

$$y_i = \begin{cases} 1 & \text{if } a_{pj} \geq 0 \\ -1 & \text{if } a_{pj} < 0 \end{cases}$$

Defining $\vec{y}$ in this way means that $a_{ij}y_j = |a_{ij}|$. Using this and the fact that $\|\vec{y}\|_\infty = 1$ we have

$$\|A\|_\infty \geq \frac{\|A\vec{y}\|_\infty}{\|\vec{y}\|_\infty} = \max_i \left| \sum_{j=1}^{n} a_{ij}y_j \right| \geq \left| \sum_{j=1}^{n} a_{pj}y_j \right| = \left| \sum_{j=1}^{n} |a_{pj}| \right| = \sum_{j=1}^{n} |a_{pj}|$$

but the last quantity on the right is the maximum row sum and the proof is complete. ■

There is one matrix norm that occurs frequently which is NOT an induced norm. It is called the Frobenius norm and is defined as

$$\|A\|_{\mathrm{F}} = \left[\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2 \right]^{1/2}$$

for a general $m \times n$ matrix A. Note that the definition of the norm provides us with a way to calculate it.

Condition Number of a Matrix

We said that one of our goals was to determine if small changes in the data of a linear system produces small changes in the solution. Now lets assume we want to solve $A\vec{x} = \vec{b}$, $\vec{b} \neq \vec{0}$ but instead we solve

$$A\vec{y} = \vec{b} + \vec{\Delta b}$$

that is, we have perturbed the right hand side by a small about $\vec{\Delta b}$. We assume that A is invertible, i.e., A^{-1} exists. For simplicity, we have not perturbed the coefficient matrix A at this point. What we want to see is how much the solution $\vec{y}$ to the perturbed system differs from the solution $\vec{x}$ of the unperturbed system. Lets write $\vec{y}$ as $\vec{x} + \vec{\Delta x}$ and so our change in the solution will be Δx. The two systems are

$$A\vec{x} = \vec{b} \qquad A(\vec{x} + \vec{\Delta x}) = \vec{b} + \vec{\Delta b}$$

What we would like to get is an estimate for the relative change in the solution, i.e.,

$$\frac{\|\Delta x\|}{\|\vec{x}\|}$$

in terms of the relative change in $\vec{b}$ where $\|\cdot\|$ denotes any induced vector norm. Subtracting these two equations gives

$$A\vec{\Delta}x = \vec{\Delta}b \quad \text{which implies} \quad \vec{\Delta}x = A^{-1}\vec{\Delta}b$$

Now we take the (vector) norm of both sides of the equation and then use our favorite inequality above

$$\|\vec{\Delta}x\| = \|A^{-1}\vec{\Delta}b\| \leq \|A^{-1}\| \, \|\vec{\Delta}b\|$$

Remember our goal is to get an estimate for the relative change in the solution so we have a bound for the change. What we need is a bound for the relative change. Because $A\vec{x} = \vec{b}$ we have

$$\|A\vec{x}\| = \|\vec{b}\| \implies \|\vec{b}\| \leq \|A\|\|\vec{x}\| \implies \frac{1}{\|\vec{b}\|} \geq \frac{1}{\|A\|\|\vec{x}\|}$$

Now we see that if we divide our previous result for $\|\vec{\Delta}x\|$ by $\|A\|\|\vec{x}\| > 0$ we can use this result to introduce $\|\vec{b}\|$ in the denominator. We have

$$\frac{\|\vec{\Delta}x\|}{\|A\|\|\vec{x}\|} \leq \frac{\|A^{-1}\|\|\vec{\Delta}b\|}{\|A\|\|\vec{x}\|} \leq \frac{\|A^{-1}\|\|\vec{\Delta}b\|}{\|\vec{b}\|}$$

Multiplying by $\|A\|$ gives the desired result

$$\frac{\|\vec{\Delta x}\|}{\|\vec{x}\|} \leq \|A\|\|A^{-1}\|\frac{\|\vec{\Delta b}\|}{\|\vec{b}\|}$$

If the quantity $\|A\|\|A^{-1}\|$ is small, then this means small relative changes in $\vec{b}$ result in small relative changes in the solution but if it is large, we could have a large relative change in the solution.

We can also derive an estimate for the case where we perturb both A and $\vec{b}$, i.e., we solve

$$(A + \Delta A)(\vec{x} + \vec{\Delta x}) = (\vec{b} + \vec{\Delta b})$$

In this case we get the estimate

$$\frac{\|\vec{\Delta x}\|}{\|\vec{x}\|} \leq \frac{\|A\|\|A^{-1}\|}{1 - \|A^{-1}\|\|\Delta A\|\|}\left[\frac{\|\Delta A\|}{\|A\|} + \frac{\|\vec{\Delta b}\|}{\|\vec{b}\|}\right]$$

Note that if $\Delta A = 0$ then this reduces to our previous estimate. So we see that

the term $\|A\|\|A^{-1}\|$ plays an important role in determining if small changes in the data produce large changes in the solution. It is so important that we give it a special name.

The condition number of a square matrix A is defined as

$$\mathcal{K}(A) \equiv \|A\|\|A^{-1}\|$$

Note that the condition number depends on what norm you are using. We say that a matrix is *well-conditioned* if $\mathcal{K}(A)$ is "small" and *ill-conditioned* otherwise.

Example Find the condition number for each of the following matrices using the infinity norm.

$$A_1 = \begin{pmatrix} 1 & 2 \\ 4 & 3 \end{pmatrix} \qquad A_2 = \begin{pmatrix} 1 & 2 \\ -0.998 & -2 \end{pmatrix}$$

First we need to find the inverse of each matrix and then take the norms. Note

the following "trick" for taking the inverse of a 2×2 matrix

$$A_1^{-1} = \frac{-1}{5}\begin{pmatrix} 3 & -2 \\ -4 & 1 \end{pmatrix} \qquad A_2^{-1} = \begin{pmatrix} 500 & 500 \\ -249.5 & -250 \end{pmatrix}$$

Now

$$\mathcal{K}_\infty(A_1) = (7)(1) = 7 \qquad \mathcal{K}_\infty(A_2) = (3)(1000) = 3000$$

Exercise Calculate the $\mathcal{K}_\infty(A)$ where A is

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

and comment on when the condition number will be large.

A classic example of an ill conditioned matrix is the Hilbert matrix; the 4×4 Hilbert matrix is given below and others can be constructed in an analogous

manner.

$$\begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \end{pmatrix}$$

Here are some of the condition numbers (using the matrix norm induced by the ℓ_2 vector norm).

n	$\mathcal{K}_2(A) \approx$
2	19
3	524
4	15,514
5	476,607
6	1.495 10^7