1.2. **Choleski decomposition.** We next look at a simplification of the LU decomposition algorithm in the case when A is symmetric, i.e., we seek a factorization of the form $A = LL^T$, known as Choleski decomposition.

We first observe that not even every symmetric, nonsingular matrix has an LU factorization. Suppose

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = LU = \begin{pmatrix} l_{11} & 0 \\ l_{21} & l_{22} \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} \\ 0 & u_{22} \end{pmatrix}.$$

Then we would need to have

$$l_{11}u_{11} = 0, \qquad l_{11}u_{12} = 1, \qquad l_{21}u_{11} = 1, \qquad l_{21}u_{12} + l_{22}u_{22} = 0.$$

But $l_{11}u_{11} = 0$ implies that either $l_{11} = 0$ or $u_{11} = 0$. Then either $l_{11}u_{12}$ or $l_{21}u_{11}$ cannot be equal to one.

However, if A is symmetric and positive definite (i.e., $x^T Ax > 0$ if $x^T x > 0$), then such a factorization is possible with $U = L^T$, i.e., $A = LL^T$. As done for the LU decomposition, the elements of L may be determined row by row by equating corresponding elements in the equation $A = LL^T$, i.e.,

$$a_{ij} = \sum_{k=1}^{j} l_{ik}l_{jk}, \qquad j = 1, \cdots, i.$$

This gives

$$l_{ij} = [a_{ij} - \sum_{k=1}^{j-1} l_{ik}l_{jk}]/l_{jj}, \quad j = 1, \cdots, i-1, \qquad l_{ii} = \left[a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2\right]^{1/2}.$$

There is also an alternative decomposition that avoids the calculation of square roots, i.e., $A = \tilde{L}D\tilde{L}^T$, where D is a positive diagonal matrix and $\tilde{L}$ is a unit lower triangular matrix. The relation between the two decompositions is that $D_{ii} = l_{ii}^2$ and $L = \tilde{L}D^{1/2}$. Then

$$LL^T = \tilde{L}D^{1/2}[\tilde{L}D^{1/2}]^T = \tilde{L}D^{1/2}[D^{1/2}]^T\tilde{L}^T = \tilde{L}D\tilde{L}^T,$$

since $D^{1/2}$ is a diagonal matrix and hence symmetric.

1.3. **Advantages of partial pivoting.** We consider the following example, taken from Forsythe and Moler: Computer Solutions of Linear Algebraic Systems (Prentice-Hall, 1967).

$$1.00 \times 10^{-4}x_1 + 1.00x_2 = 1.00, \qquad 1.00x_1 + 1.00x_2 = 2.00,$$

whose exact solution is given by $x_1 = 1.00010001\cdots$, $x_2 = 0.999099\cdots$. We now compute the solution by Gaussian elimination using 3-digit arithmetic. Subtract 1.00×10^4 times the first equation from the second. Then the second equation becomes $\alpha x_2 = \beta$, where

$$\alpha = 1.00 - 1.00 \times 10^4 \times 1.00 = 0.0001 \times 10^4 - 1.00 \times 10^4 = -1.00 \times 10^4,$$
$$\beta = 2.00 - 1.00 \times 10^4 \times 1.00 = 0.0002 \times 10^4 - 1.00 \times 10^4 = -1.00 \times 10^4,$$

using 3-digit arithmetic. Hence the computed solution is $x_2 = 1$ and $x_1 = 0$.

Now consider the same problem with pivoting. Since the largest entry in the first column occurs in the second equation, we would use the second equation to eliminate the x_1 entry

in the first equation, i.e., we multiply the second equation by 10^{-4} and subtract it from the first equation. Then the first equation becomes:

$$(1 - 10^{-4})x_2 = 1 - 2 \times 10^{-4}.$$

In 3-digit arithmetic, this is $x_2 = 1$. Hence, the computed solution is $x_2 = 1$, $x_1 = 1$.

When considering a pivoting strategy, a related issue is the concept of scaling of a matrix. The idea is that the solution x of $Ax = b$ is also the solution of $\tilde{A}x = \tilde{b}$, where $\tilde{A}$ and $\tilde{b}$ are obtained from A and b by multiplying any row of A and the corresponding element of b by a non-zero constant. Hence, choosing the largest element of a column as a pivot without normalizing the rows in some way is a problem, since almost any pivot selection could be achieved by some scaling. One technique to avoid this confusion is to choose scaling factors $k(i)$ such that each row of A has its largest element (in absolute value) equal to one. In fact, we don't actually multiply the matrix by the scaling factors, since this could introduce roundoff errors. Instead, we leave A unscaled, but choose the pivots as if the entries had been scaled, i.e., choose int_r such that

$$|k(int_r)l_{int_r,r}| = \max_{i \geq r} |k(i)l_{ir}|.$$

2. Matrix and vector norms

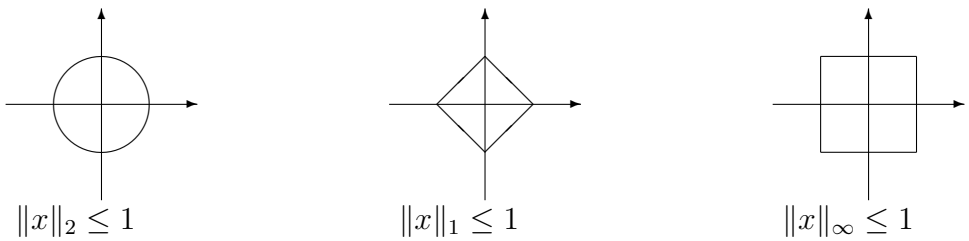
To analyze numerical methods for problems in linear algebra, it is helpful to define various ways of comparing the sizes of vectors and matrices. We do this by defining the concept of a norm and then give concrete examples of norms of vectors and matrices that are convenient for various applications.

Definition: A norm in $\mathbb{R}^n$ is a function that assigns to each x in $\mathbb{R}^n$, a non-negative number $\|x\|$ (called the norm of x) satisfying: (i) $\|x\| = 0$ if and only if $x = 0$, (ii) $\|\alpha x\| = |\alpha|\|x\|$ for each $x \in \mathbb{R}^n$ and every constant α, and (iii) $\|x + y\| \leq \|x\| + \|y\|$ for each x and y in $\mathbb{R}^n$.

The examples we will use are the p-norms defined by $\|x\|_p = (\sum_{i=1}^n |x_i|^p)^{1/p}$, where $x = (x_1, \cdots, x_n)$. In particular, we shall use

$$\|x\|_1 = \sum_{i=1}^n |x_i|, \qquad \|x\|_2 = (\sum_{i=1}^n |x_i|^2)^{1/2}, \qquad \|x\|_\infty = \max_{1 \leq i \leq n} |x_i| \qquad \text{max norm.}$$

For the $1, 2, \infty$ norms, the sets $\{x = (x_1, x_2) : \|x\| \leq 1\}$ are shown in the figure below.



$\|x\|_2 \leq 1$ $\|x\|_1 \leq 1$ $\|x\|_\infty \leq 1$

We will also need the concept of a matrix norm. In addition to satisfying properties (i), (ii), and (iii), we would also like to have the property: (iv) $\|AB\| \leq \|A\|\|B\|$. There is a very natural way this can be done. Set

$$\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|}, \tag{2.1}$$

where $\|x\|$ denotes some vector norm of x. It is clear from the definition that for any $y \neq 0 \in R^n$,

$$\frac{\|Ay\|}{\|y\|} \leq \max_{x \neq 0} \frac{\|Ax\|}{\|x\|} = \|A\|.$$

Hence,

$$\|Ay\| \leq \|A\|\|y\|,$$

and this result holds for $y = 0$ as well. We can then show that property (iv) follows from this result.

$$\|AB\| = \max_{x \neq 0} \frac{\|ABx\|}{\|x\|} \leq \max_{x \neq 0} \frac{\|A\|\|Bx\|}{\|x\|} \leq \|A\|\|B\|.$$

For each of the examples of vector norms given above, we can also derive concrete formulas for the corresponding matrix norms. We omit the derivation and just give the results.

$$\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^{n} |a_{ij}| \qquad \text{maximum column sum}$$

$$\|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^{n} |a_{ij}| \qquad \text{maximum row sum}$$

$$\|A\|_2 = [\rho(A^*A)]^{1/2} \qquad \text{spectral norm}$$

where $\rho(A) = \max_s |\lambda_s(A)|$ and $\lambda_s(A)$ denotes the sth eigenvalue of A. Here $A^* = \bar{A}^T$, where $\bar{A}$ denotes the complex conjugate and the superscript T denotes the transpose.

There is sometimes confusion between the spectral radius $\rho(A)$ and the spectral norm, defined above. Some relationships are given by the following.

a) If A is Hermitian, i.e., $A^* = A$, then $\|A\|_2 = \rho(A)$.

b) For any matrix norm defined by (2.1), $\rho(A) \leq \|A\|_2$.

It is easy to see (a) in the case when A is real and symmetric. Then $A^* = A$, and since the eigenvalues of A^2 are the square of the eigenvalues of A, we find that $\|A\|_2 = \rho(A)$.

Remark: Although $\|x\|_2$ is called the Euclidean norm of the vector x, $\|A\|_2$ is the spectral norm, rather than the Euclidean norm of the matrix A. There is also a norm, defined by $\|A\|_F = (\sum_{i,j=1}^{n} (a_{ij})^2)^{1/2}$, that is called the Frobenius norm.