3. Special Linear Systems
3.1. Symmetric Positive Definite Systems

• Reading: Trefethen and Bau (1997), Lecture 23

• Simplify Gaussian elimination when $A$ has special properties
  - Symmetry
  - positive definiteness
  - Banded structure
  - Sparsity
  - Block structure

• **Definition 1:** $A \in \mathbb{R}^{n \times n}$ is symmetric if $A^T = A$

• **Definition 2:** $A \in \mathbb{R}^{n \times n}$ is *positive definite* if $x^T Ax > 0$
  for all nonzero $x \in \mathbb{R}^n$

• If $A$ is positive definite:
  - and $X \in \mathbb{R}^{n \times k}$ has rank $k$ then $X^T AX$ is positive definite
  - then all of its principal submatrices are positive definite
  - then all of its diagonal entries are positive
  - then it may be factored as
    $$A = LU = LDM^T$$
    where $L$ and $M$ are unit lower triangular and $D$ is diagonal with positive entries
  
  * The entries of $D$ are the diagonal entries of $U$
  
  – These properties are proved in Golub and Van Loan (1993), Section 4.2 or Demmel (1997), Section 2.7
Symmetric Positive Definite Systems

- Symmetric positive definite systems:
  - Arise in the finite difference or finite element solution of elliptic PDEs
  - $A$ may be factorized as
    $$A = LDL^T$$ (1a)

  with
  $$L = \begin{bmatrix}
  1 & & & \\
  l_{21} & 1 & & \\
  l_{31} & l_{32} & & \\
  & \vdots & \ddots & \\
  l_{n1} & l_{n2} & & 1
  \end{bmatrix}, \quad D = \begin{bmatrix}
  d_1 & & & \\
  & d_2 & & \\
  & & \ddots & \\
  & & & d_n
  \end{bmatrix}$$ (1b)

  * Zero entries are not shown

  - By direct factorization without pivoting
    $$d_j = a_{jj} - \sum_{k=1}^{j-1} d_k l_{jk}^2,$$ (2a)
  
    $$l_{ij} = \frac{1}{d_j} (a_{ij} - \sum_{k=1}^{j-1} d_k l_{jk} l_{ik}), \quad i = j + 1 : n,$$
    $$j = 1 : n$$ (2b)

  * Summations are zero when the lower limit exceeds the upper one
Symmetric Positive Definite Factorization

function \( A = \text{ldlt}(A) \)

% \text{ldlt}: Factorization of a symmetric positive definite n-by-n
% matrix \( A \) into the product \( LDL^T \), where \( L \)
% is unit lower triangular and \( D \) is diagonal. On return,
% \( A(i,j) \) stores \( L(i,j) \) if \( i > j \) and \( D(i) \) if \( i = j \).

\[
[n \ n] = \text{size}(A);
\]
\[
\text{for } j = 1:n
\]
\[
\begin{align*}
&\% \text{ Compute } v(k), k = 1: j - 1 \\
&\text{for } k = 1: j - 1 \\
&\quad v(k) = A(j,k) \ast A(k,k);
\end{align*}
\]
\[
\text{end}
\]
\[
\begin{align*}
&\% \text{ Compute } d(j) \\
&v(j) = A(j,j) - \text{dot}(A(j,1:j-1),v(1:j-1)); \\
&A(j,j) = v(j);
\end{align*}
\]
\[
\begin{align*}
&\% \text{ Compute } l(i,j), i = j + 1: n \\
&\text{for } i = j + 1:n \\
&A(i,j) = (A(i,j) - \text{dot}(A(i,1:j-1),v(1:j-1)))/v(j);
\end{align*}
\]
\[
\text{end}
\]
\[
\text{end}
\]

- We let \( v_k = d_k l_{jk}, k = 1 : j - 1 \)
- We should use a symmetric storage mode, cf. Section 1.2
- The algorithm requires approximately \( n^3/3 \) FLOPs (half that of Gaussian elimination)
- Pivoting is not necessary for stability
Forward and Backward Substitution

- From (1a)
  \[ \mathbf{Ax} = \mathbf{LDL}^T \mathbf{x} = \mathbf{b} \]

  - Let
  \[ \mathbf{L}^T \mathbf{x} = \mathbf{y}, \quad \mathbf{Dy} = \mathbf{z}, \quad \mathbf{Lz} = \mathbf{b} \quad (3) \]

  - Forward substitution:
  \[ z_i = b_i - \sum_{k=1}^{i-1} l_{ik} z_k, \quad i = 1 : n \quad (4a) \]

  - Diagonal scaling
  \[ y_i = z_i / d_i, \quad i = 1 : n \quad (4b) \]

  - Backward substitution
  \[ x_i = y_i - \sum_{k=i+1}^{n} l_{ki} x_k, \quad i = n : -1 : 1 \quad (4c) \]

- Cholesky Factorization:
  \[ \mathbf{A} = \tilde{\mathbf{L}} \tilde{\mathbf{L}}^T, \quad \tilde{\mathbf{L}} = \mathbf{LD}^{1/2} \]

  - The diagonal entries of \( \tilde{\mathbf{L}} \) are square roots of the entries of \( \mathbf{D} \)
  - cf. Trefethen and Bau (1997), Lecture 23
  - The \( n \) square roots may be expensive