Naive Gaussian Elimination

The following algorithms implement naive Gaussian elimination followed by back substitution to compute the solution of \( Ax = b \), where \( A \) is an \( n \times n \) matrix with \( ij \)th entry \( a_{ij} \) and \( b \) is an \( n \)-vector with \( i \)th component \( b_i \).

These are structured as most modern software library routines are.\(^1\) The first algorithm performs naive Gaussian elimination on \( A \), overwriting each \( a_{ik} \) in the lower triangular part of \( A \) with the “multiplier” \(-a_{ik}/a_{kk}\). In library software, this algorithm is usually coded in a routine separate from the other two; in an application, it usually requires most of the computation. The second and third algorithms are usually combined in one routine that uses the output of the first routine. The second algorithm performs the same row operations on \( b \) that were performed on \( A \). The third algorithm performs back substitution, overwriting \( b \) with the solution. Note that these algorithms require no more storage than is required for \( A \) and \( b \).

Naive Gaussian Elimination:

\[
\begin{align*}
\text{For } k = 1, \ldots, n - 1 & \\
\quad \text{For } i = k + 1, \ldots, n & \\
\quad \quad a_{ik} & \leftarrow -a_{ik}/a_{kk} \\
\quad \quad \text{For } j = k + 1, \ldots, n & \\
\quad \quad \quad a_{ij} & \leftarrow a_{ij} + a_{ik}a_{kj}
\end{align*}
\]

Row Operations on \( b \):

\[
\begin{align*}
\text{For } k = 1, \ldots, n - 1 & \\
\quad \text{For } i = k + 1, \ldots, n & \\
\quad b_i & \leftarrow b_i + a_{ik}b_k
\end{align*}
\]

Back Substitution:

\[
\begin{align*}
\quad b_n & \leftarrow b_n/a_{nn} \\
\quad \text{For } i = n - 1, \ldots, 1 & \\
\quad b_i & \leftarrow \left( b_i - \sum_{j=i+1}^{n} a_{ij}b_j \right)/a_{ii}
\end{align*}
\]

\(^1\)The MATLAB “backslash” command, which produces the solution of \( Ax = b \) with the single command \( A \backslash b \), appears to be an exception. In fact, it separates the operations on \( A \) from those on \( b \) as above “under the covers.” Because naive Gaussian elimination is unstable, MATLAB’s backslash command actually implements Gaussian elimination with partial pivoting.