Algorithm for Gaussian elimination with partial pivoting

Gaussian elimination

Do for $i = 1, \ldots, n-1$ (Use row 1 to $n-1$ as pivoting row)

G1: Find pivot element

Find the smallest integer p ($i \leq p \leq n$) so that $a_{pi} = \max_{i \leq j \leq n} |a_{ji}|$.

If $a_{pi} = 0$ a unique solution does not exist.

Else if $p \neq i$ interchange row i and p: $E_p \leftrightarrow E_i$.

G2: Perform the Gaussian elimination with the non-zero pivot.

Do for $j = i+1, \ldots, n$ (On all rows below the pivot element)

$m_{ji} = \dfrac{a_{ji}}{a_{ii}}$

Do for $k = i+1, \ldots, n$ (For all columns on the right of the pivot element)

$a_{jk} = a_{jk} - m_{ji} a_{ik}$

End Do (k-loop)

$b_j = b_j - m_{ji} b_i$

End Do (j-loop)

End Do (i-loop)

Backward substitution

B1: Check whether backward substitution is possible.

If $a_{nn} = 0$ no unique solution exists

B2: Perform backward substitution.

Do for $i = n, \ldots, 1$ (From the last row back to the first)

Do for $j = i+1, \ldots, n$ (Subtract contributions from known values x_j)

$b_i = b_i - a_{ij} x_j$

End Do (j-loop)

$x_i = \dfrac{b_i}{a_{ii}}$

End Do (i-loop)

Remarks:

- Gaussian elimination with partial pivoting is more robust than the basic Gaussian elimination algorithm.
- The only step that changes is step G1. This step now requires a little bit more time since you always have to search the whole column below the pivot element. Compared to the number of multiplacitions/divisions and additions/subtractions the extra work is negligible.