A short note on solving linear algebraic equations : Gauss-Jordon

Suppose we are asked to solve a linear system of equations, say with 3 variables x_1, x_2, x_3 but can be easily extended to n variables,

$$\mathbf{A}\,\mathbf{X} = \mathbf{B} \quad \Rightarrow \quad \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \tag{1}$$

The first step of solving the above equation (1) is to represent it in *augmented matrix form*,

$$[\mathbf{A}|\mathbf{B}] \quad \Rightarrow \quad \left[\begin{array}{ccc|c} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{array}\right] \tag{2}$$

Next follows reducing the above matrix **A** or the augmented matrix of equation (2) in *reduced row echelon form* (RREF) by using **Gauss-Jordan elimination** through any one or a combination of three elementary row operations

- swapping two rows
- multiplying a row by a nonzero number
- adding or subtracting a multiple of one row to another row

In the reduced row echelon form,

1. all rows with only zero entries are at the bottom of the matrix
2. the first nonzero entry in a row (called **pivot**) of each nonzero row is to the right of the leading entry of the row above it
3. leading entry *i.e.* pivot in any nonzero row is 1
4. all other entries in the row or column containing a leading 1 are zeros.

For example, the above matrix **A** or the augmented matrix $[\mathbf{A}|\mathbf{B}]$ in reduced row echelon form is either one of the two shown below,

$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & \tilde{b}_1 \\ 0 & 1 & \tilde{a}_{23} & \tilde{b}_2 \\ 0 & 0 & 0 & \tilde{b}_3 \end{array}\right] \quad \text{or} \quad \left[\begin{array}{ccc|c} 1 & 0 & 0 & \tilde{b}_1 \\ 0 & 1 & 0 & \tilde{b}_2 \\ 0 & 0 & 1 & \tilde{b}_3 \end{array}\right] \tag{3}$$

To solve the system of linear equation (1) if we encounter the left RREF of (3) then we will have arbitrary values for x_3, in other words, we will not get a unique solution. But for the right RREF we will obtain a unique solution. To perform Gauss-Jordan elimination with *partial pivoting*

1. Swap the rows so that all rows with all zero entries are at the bottom

$$\left[\begin{array}{ccc|c} a_{11} & a_{12} & a_{13} & b_1 \\ 0 & 0 & 0 & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{array}\right] \xrightarrow{R_2 \leftrightarrow R_3} \left[\begin{array}{ccc|c} a_{11} & a_{12} & a_{13} & b_1 \\ a_{31} & a_{32} & a_{33} & b_3 \\ 0 & 0 & 0 & b_2 \end{array}\right]$$

2. Swap the rows so that the row with leftmost nonzero entry is at the top

$$\left[\begin{array}{ccc|c} 0 & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{array}\right] \xrightarrow{R_1 \leftrightarrow R_2} \left[\begin{array}{ccc|c} a_{21} & a_{22} & a_{23} & b_2 \\ 0 & a_{12} & a_{13} & b_1 \\ a_{31} & a_{32} & a_{33} & b_3 \end{array}\right] \quad \text{where,} \quad a_{21} \neq 0$$

The first row is now the *pivot row* and a_{21} is the *pivot element*. In partial pivoting, the algorithm selects the entry with largest absolute value from the pivoting column. Therefore, if the pivot element turns out to be 0 (as above) or order(s) of magnitude smaller than a_{21} or a_{31}, then swap the rows such that the row with largest leftmost entry is at the top.

3. Multiply the *pivot row* $a_{11}^0 \cdots a_{13}^0$, b_1^0 with the reciprocal of *pivot element* a_{11}^0, thus making the leading entry 1. The *pivot column* is $a_{11}^0 \cdots a_{31}^0$.

$$\left[\begin{array}{ccc|c} a_{11}^0 & a_{12}^0 & a_{13}^0 & b_1^0 \\ a_{21}^0 & a_{22}^0 & a_{23}^0 & b_2^0 \\ a_{31}^0 & a_{32}^0 & a_{33}^0 & b_3^0 \end{array}\right] \xrightarrow{R_1^0/a_{11}^0 \rightarrow R_1^1} \left[\begin{array}{ccc|c} 1 & a_{12}^1 & a_{13}^1 & b_1^1 \\ a_{21}^0 & a_{23}^0 & a_{23}^0 & b_2^0 \\ a_{31}^0 & a_{32}^0 & a_{33}^0 & b_3^0 \end{array}\right]$$

where $a_{1(1,2,3)}^1 = a_{1(1,2,3)}^0/a_{11}^0$ and $b_1^1 = b_1^0/a_{11}^0$.

4. Convert the remaining elements of pivot column by adding / subtracting multiples of pivot row from the following rows,

$$\begin{aligned} a_{2(1,2,3)}^1 &= a_{2(1,2,3)}^0 - a_{21}^0 * R_1^1, \qquad & b_2^1 &= b_2^0 - a_{21}^0 * b_1^1 \\ a_{3(1,2,3)}^1 &= a_{3(1,2,3)}^0 - a_{31}^0 * R_1^1, \qquad & b_3^1 &= b_3^0 - a_{31}^0 * b_1^1 \end{aligned}$$

which result in

$$\left[\begin{array}{ccc|c} 1 & a_{12}^1 & a_{13}^1 & b_1^1 \\ 0 & a_{22}^1 & a_{23}^1 & b_2^1 \\ 0 & a_{32}^1 & a_{33}^1 & b_3^1 \end{array}\right]$$

5. Repeat the above steps 3 and 4 with the new *pivot element* being a_{22}^1 and the *pivot row* and *pivot column* respectively are $a_{22}^1 \cdots b_2^1$ and $a_{12}^1 \cdots a_{32}^1$. Subsequently for all the rows. This results in an augmented matrix,

$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & b_1^3 \\ 0 & 1 & 0 & b_2^3 \\ 0 & 0 & 1 & b_3^3 \end{array}\right] \tag{4}$$

Now, given the above augmented matrix, the solution to the system of linear equation (1) obviously is $x_1 = b_1^3$, $x_2 = b_2^3$, $x_3 = b_3^3$.

The above is the process that is called Gauss-Jordan elimination using *partial pivoting.* It is important to note that no matter what steps and in which order they are applied, the final augmented matrix in (4) is unique.

Let us consider a concrete example by trying to solve the following system

$$\begin{aligned} 2y + 5z &= 1 && (5)\\ 3x - y + 2z &= -2 && (6)\\ x - y + 3z &= 3 && (7) \end{aligned}$$

The corresponding augmented matrix is

$$\left[\begin{array}{rrr|r} 0 & 2 & 5 & 1\\ 3 & -1 & 2 & -2\\ 1 & -1 & 3 & 3 \end{array}\right] \tag{8}$$

Since we have $a_{11} = 0$, swap $R_1 \leftrightarrow R_2$ (or R_3), it changes nothing of the system of the linear equation in our example above. Thus the new augmented matrix is

$$\left[\begin{array}{rrr|r} 3 & -1 & 2 & -2\\ 0 & 2 & 5 & 1\\ 1 & -1 & 3 & 3 \end{array}\right] \tag{9}$$

The steps 3 is applied by dividing the first row (pivot row) of the new augmented matrix (9) by 3. Since $a_{21}^0 = 0$ already $R_2^1 = R_2^0$, and step 4 implies $R_3^1 = R_3^0 - R_1^1$, yielding

$$\xrightarrow{R_1^0/3 \to R_1^1} \left[\begin{array}{rrr|r} 1 & -1/3 & 2/3 & -2/3\\ 0 & 2 & 5 & 1\\ 1 & -1 & 3 & 3 \end{array}\right] \xrightarrow{R_3^0 - R_1^1 \to R_3^1} \left[\begin{array}{rrr|r} 1 & -1/3 & 2/3 & -2/3\\ 0 & 2 & 5 & 1\\ 0 & -2/3 & 7/3 & 11/3 \end{array}\right] \tag{10}$$

Perform step 3 on the pivot row R_2^1

$$\begin{aligned} &\xrightarrow{R_2^1/2 \to R_2^2} \left[\begin{array}{rrr|r} 1 & -1/3 & 2/3 & -2/3\\ 0 & 1 & 5/2 & 1/2\\ 0 & -2/3 & 7/3 & 11/3 \end{array}\right] \xrightarrow{R_1^1 + R_2^2/3 \to R_1^2} \left[\begin{array}{rrr|r} 1 & 0 & 9/6 & -3/6\\ 0 & 1 & 5/2 & 1/2\\ 0 & -2/3 & 7/3 & 11/3 \end{array}\right]\\ &\xrightarrow{R_3^1 + 2R_2^2/3 \to R_3^2} \left[\begin{array}{rrr|r} 1 & 0 & 9/6 & -3/6\\ 0 & 1 & 5/2 & 1/2\\ 0 & 0 & 12/3 & 24/6 \end{array}\right] \end{aligned} \tag{11}$$

Again steps 3 and 4 are applied on the last pivot row R_3^2,

$$
\xrightarrow{3R_3^2/12\to R_3^3}
\left[\begin{array}{ccc|c}
1 & 0 & 9/6 & -3/6\\
0 & 1 & 5/2 & 1/2\\
0 & 0 & 1 & 1
\end{array}\right]
\xrightarrow{R_2^2-5R_3^3/2\to R_2^3}
\left[\begin{array}{ccc|c}
1 & 0 & 9/6 & -3/6\\
0 & 1 & 0 & -2\\
0 & 0 & 1 & 1
\end{array}\right]
$$

$$
\xrightarrow{R_1^2-9R_3^3/6\to R_1^3}
\left[\begin{array}{ccc|c}
1 & 0 & 0 & -2\\
0 & 1 & 0 & -2\\
0 & 0 & 1 & 1
\end{array}\right]
\tag{12}
$$

Therefore, the solution is $x = -2$, $y = -2$, $z = 1$.

The Gauss-Jordon elimination technique can easily be extended to obtain inverse of an invertible matrix. To do that, we begin with re-writing the form of (1),

$$
\mathbf{A}\cdot\mathbf{B} = \mathbb{1} \quad\Rightarrow\quad
\begin{pmatrix}
a_{11} & a_{12} & a_{13}\\
a_{21} & a_{22} & a_{23}\\
a_{31} & a_{32} & a_{33}
\end{pmatrix}
\cdot
\begin{pmatrix}
b_{11} & b_{12} & b_{13}\\
b_{21} & b_{22} & b_{23}\\
b_{31} & b_{32} & b_{33}
\end{pmatrix}
=
\begin{pmatrix}
1 & 0 & 0\\
0 & 1 & 0\\
0 & 0 & 1
\end{pmatrix}
\tag{13}
$$

where $\mathbf{B} = \mathbf{A}^{-1}$. So the augmented matrix is

$$
\left[\begin{array}{ccc|ccc}
a_{11} & a_{12} & a_{13} & 1 & 0 & 0\\
a_{21} & a_{22} & a_{23} & 0 & 1 & 0\\
a_{31} & a_{32} & a_{33} & 0 & 0 & 1
\end{array}\right]
\equiv [\mathbf{A}|\mathbb{1}]
\tag{14}
$$

A Gauss-Jordon on (14) will yield the inverse in the place of $\mathbb{1}$ part of the augmented matrix.

Determining the determinant is even easier because one just needs a row echelon matrix, instead of fully reduced row echelon matrix. The steps involve is

$$
\det\begin{pmatrix}
a_{11} & a_{12} & a_{13}\\
a_{21} & a_{22} & a_{23}\\
a_{31} & a_{32} & a_{33}
\end{pmatrix}
= \det\begin{pmatrix}
a'_{11} & a'_{12} & a'_{13}\\
0 & a'_{22} & a'_{23}\\
0 & 0 & a'_{33}
\end{pmatrix}
= (-1)^n a'_{11}\, a'_{22}\, a'_{33}
\tag{15}
$$

where n is the number of swaps *i.e.* number of times the rows are interchanged, $a'_{ii} \neq 0$ and none of a'_{ii} has to be 1.

Flow chart for Gauss-Jordon elimination

An $A = N \times N$ matrix when augmented becomes $Ab = N \times (N+1)$ matrix for solving linear equations and $Ab = N \times (N+N)$ matrix for inverse, where $nrows = N$, $ncols = N+1$ or $N+N$. One version of partial pivoting involve swapping rows such that $Ab[0][0]$ is maximum of all the elements in 0^{th} column. If you wish you may modify the following pseudo-code accordingly. The pseudo-codes are

```
partialPivot(Ab,m,nrows,ncols)
    pivot = Ab[m][m]
    if (Ab[m][m] ≠ 0) return;
    else Loop:  r = m + 1, ... nrows
       if (Ab[r][m] ≠ 0) then pivot = Ab[r][m]; swapRows(Ab,m,r,ncols);
       else next r;
    if (pivot == 0) return (No unique solution);

swapRows(Ab,pvelm,rw,ncols)
    temp[ncols] = 0
    Loop:  c = 1, ... ncols
       temp[c] = Ab[pvelm][c]
       Ab[pvelm][c] = Ab[rw][c]
       Ab[rw][c] = temp[c]

gaussJordon(Ab,nrows,ncols)
    Loop:  r = 1, ... nrows
       partialPivot(Ab,r,nrows,ncols); exit if no solution exists
       Loop:  c = r, ... ncols do Ab[r][c]* = 1/Ab[r][r]
       Loop:  r1 = 1, ... nrows
          if (r1 == r || Ab[r1][r] == 0) next r1;
          else
             factor = Ab[r1][r]
             Loop:  c = r, ... ncols do Ab[r1][c]− = factor * Ab[r][c]
    print Ab[r][r] as solutions
```