Algorithm for Solving Systems of Linear Equations

Aakash Sarmandal

Research Scholar, Jodhpur National University, Rajasthan, INDIA

Abstract: We review here some of the basic definitions and elementary algebraic operations on matrices. There are many applications as well as much interesting theory revolving around these concepts, which we encourage you to explore after reviewing this article. Gaussian elimination is an algorithm for solving systems of linear equations. It can also be used to find the rank of a matrix, to calculate the determinant of a matrix, and to calculate the inverse of an invertible square matrix. Row pivoting, column pivoting, complexity theory etc. explained in this article.

Keywords: Algebra, Big O notation, Gaussian Elimination, Optimization, Sparse Matrices, Time complexity

1. INTRODUCTION

We review here some of the basic definitions and elementary algebraic operations on matrices.

There are many applications as well as much interesting theory revolving around these concepts, which we encourage you to explore after reviewing this article.

A matrix is simply a rectangular array of numbers. For example,

A =	a ₁₁	a_{12}	• • •	a_{1n}
	a_{21}	a_{22}	• • •	a_{2n}
	:	:	${}^{*}\cdot,$:
	a_{m1}	a_{m2}		a_{mn}

Developments in linear algebra have contributed a great deal to progress in solving optimization problems on computers. The relationship between optimization algorithms and matrix methods in linear algebra has been called symbiotic (O'Leary 2000, 1). Many literature citations exist and there are many textbooks and references on the subject including Stewart (1973), Strang (1976), Golub and Van Loan (1983), Kahan, Molar and Nash (1989), and Bjorck(1996). Gauss elimination, Gauss-Jordan elimination, LU Decomposition, Cholesky Factorization, QR Factorization, and Singular Value Decomposition that are some of the main techniques used to efficiently solve matrix equations of the form Ax = b and, thus, to solve optimization problems. We will give brief descriptions of Gauss elimination, Gauss-Jordan elimination. However, complete coverage of the subject is beyond the scope of this work. There are also specialized techniques available that involve choices related to the nature of the problem and the sparsity of the **A** matrix.

Gauss and Gauss-Jordan Elimination

In linear algebra, Gaussian elimination is an algorithm for solving systems of linear equations. It can also be used to find the rank of a matrix, to calculate the determinant of a matrix, and to calculate the inverse of an invertible square matrix. The method is named after Carl Friedrich Gauss but it was not invented by him.

Elementary row operations are used to reduce a matrix to what is called triangular form (in numerical analysis) or row echelon form (in abstract algebra). Gauss–Jordan elimination, an extension of this algorithm, reduces the matrix further to diagonal form, which is also known as reduced row echelon form. Gaussian elimination alone is sufficient for many applications, and requires fewer calculations than the Gauss–Jordan version.

Solution of Ax = b by Gaussian elimination is conversion of **A** to a triangular matrix by elementary operations followed by determination of **x** by back substitution (Harris and Stocker 1998, 441). These terms are be defined for later use.

□ Triangular matrix. A matrix whose elements are zero on one side of its diagonal.

> Upper triangular matrix. Non-zero elements of the triangular matrix are on or above the diagonal.

> Lower triangular matrix. Non-zero elements of the triangular matrix are on or below the diagonal.

Elementary operations. Operations on the matrix equation Ax=b that do not change the answer. There are three types of elementary operations on the matrix equation Ax = b.

> Multiply a row of \mathbf{A} and corresponding row of \mathbf{b} by a scalar factor.

Add or subtract a multiple of a row of **A** and the corresponding row of **b** to another row of A and corresponding row of b. Mathematically, this multiplication is represented by equation (1):

$$(1 - s_{rc}E^{rc})Ax = (1 - s_{rc}E^{rc})b \equiv E_{s}^{rc}Ax = E_{s}^{rc}b$$

where,

 s_{rc} is a scalar multiplier for the operation E_s^{rc} ,

$$E_{ij}^{rc} = \begin{cases} 1 \text{ if } i = r \land j = c \\ 0 \text{ if } i \neq r \lor j \neq c \end{cases}$$

$$E_{s}^{rc} \equiv (1 - s_{rc}E^{rc})$$
(1)

Row Pivoting. Interchanging two rows of **A** and the corresponding rows of **b**. Row pivoting amounts to multiplying both sides of Ax=b by a matrix R^{rs} to give $R^{rs}Ax = R^{rs}b$ that interchanges row r and row s. The row pivot matrix, R^{rs} , is given by equation (2):

$$R_{ij}^{rs} = \begin{cases} 1 \text{ if } i = r \land j = s \\ 1 \text{ if } i = s \land j = r \\ 0 \text{ otherwise} \end{cases}$$
(2)

Column Pivoting. Interchanging two columns of **A** and the corresponding rows of **x**. Column pivoting may be represented by multiplication by a matrix C^{cs} to obtain: A $C^{cs}[C^{cs}]$ -1x = b which represents the interchange of columns c and s in **A** and rows c and s of **x**. Note that since the operation is a row pivot on **x**, then $[C^{cs}]$ -1 = R^{cs}.

Back substitution. The process of solving **Ax=b** when **A** is in triangular form. For example, if **A** is in upper triangular

Available online at www.ignited.in E-Mail: ignitedmoffice@gmail.com form, then $xn = b_n/Ann$. But, since **A** is upper triangular, we have

 $A_{n-1,n-1}x_{n-1} + A_{n-1,n}x_n = b_{n-1} => x_{n-1} = (b_{n-1} - A_{n-1,nxn}x_n)/A_{n-1,n-1}$. This pattern is repeated until we have solved for x1.

With the above definitions, it is now possible to illustrate the Gaussian elimination method. If, in E^{rc} , we require that $s_{rc} = A_{rc}/A_{cc}$, then the formation of an upper triangular matrix is represented in equation (3):

$$\left[\prod_{j=1}^{n} \left\{\prod_{i=j+1}^{m} \mathbf{E}_{s}^{ij}\right\}\right] \mathbf{A}\mathbf{X} = \left[\prod_{j=1}^{n} \left\{\prod_{i=j+1}^{m} \mathbf{E}_{s}^{ij}\right\}\right] \mathbf{b}$$
(3)

It is important to note that the scalar multiplier is computed based on the current modification of the **A** matrix. The above procedure can fail through attempted division by a zero valued diagonal element even if the **A** matrix is nonsingular. A zero valued diagonal element may result by chance due to the combination of previous operations. Thus, row pivoting is absolutely required for stability of the method as in equation (4.

$$\left[\prod_{j=1}^{n}\left\{\prod_{i=j+1}^{m}\mathbf{E}_{s}^{ij}\right\}\mathbf{R}^{jMaxj}\right]\mathbf{A}\mathbf{x}=\left[\prod_{j=1}^{n}\left\{\prod_{i=j+1}^{m}\mathbf{E}_{s}^{ij}\right\}\mathbf{R}^{jMaxj}\right]\mathbf{b}$$
(4)

where Maxjis the row in column j below the diagonal with the largest value.

It is further helpful to pivot the largest element in a column to the diagonal in order to help reduce round-off error. Other types of pivoting, including column pivoting, are possible. There is great deal of debate about pivoting methods in the literature (Press *et al.* 1988, 38-39).



In practice, E and R are sparse matrices and it is more efficient to perform the elimination directly instead of with matrix operations. Also, it is not necessary to keep the lower triangle of **A** up to date. The algorithm is usually performed in place and the final A contains the upper triangular values and garbage in the lower part. Several algorithms for Gaussian elimination are available in the literature (Nash 1990, 76; Harris and Stocker 1998, 441-445; Press *et al.* 1988, 41-43).

n linear algebra, Gauss–Jordan elimination is an algorithm for getting matrices in reduced row echelon form using elementary row operations. It is a variation of Gaussian elimination. Gaussian elimination places zeros below each pivot in the matrix, starting with the top row and working downwards. Matrices containing zeros below each pivot are said to be in row echelon form. Gauss–Jordan elimination goes a step further by placing zeros above and below each pivot; such matrices are said to be in reduced row echelon form. Every matrix has a reduced row echelon form, and Gauss–Jordan elimination is guaranteed to find it.

It is named after Carl Friedrich Gauss and Wilhelm Jordan because it is a variation of Gaussian elimination as Jordan described in 1887. However, the method also appears in an article by Clasen published in the same year. Jordan and Clasen probably discovered Gauss–Jordan elimination independently.

Computer science's complexity theory shows Gauss– Jordan elimination to have a time complexity of O(n3) for an n by n matrix (using Big O Notation). This result means it is efficiently solvable for most practical purposes. As a result, it is often used in computer software for a diverse set of applications. However, it is often an unnecessary step past Gaussian elimination. Gaussian elimination shares Gauss-Jordan's time complexity of O(n3) but is generally faster. Therefore, in cases in which achieving reduced row echelon form over row echelon form is unnecessary, Gaussian elimination is typically preferred. The Gaussian elimination technique solves the equation Ax=b by converting **A** to a triangular matrix. Now, suppose we have a new **b** vector. Then, **A** will have to be reconstructed and the process repeated. With Gauss-Jordan elimination, **A** is converted to the unit matrix, the inverse of **A** is calculated and the new **b** vectors may be converted to new **x** vectors by multiplication by A^{-1} . The process is an extension of Gaussian elimination in which the upper triangular matrix formed is further processed by elementary operations to the unit matrix. Solving the augmented matrix equation (5) with the Gauss-Jordan algorithm results in solutions for $x_1, x_2, ..., x_n$, and A^{-1} (Press *et al.* 1988, 37).

$$\mathbf{A}[\mathbf{x}_1 \bigsqcup \mathbf{x}_2 \bigsqcup \ldots \bigsqcup \mathbf{x}_n \bigsqcup \mathbf{A}^{-1}] = [\mathbf{b}_1 \bigsqcup \mathbf{b}_2 \bigsqcup \ldots \bigsqcup \mathbf{b}_n \bigsqcup 1]$$
(5)

However, the additional storage required to perform the algorithm and the round-off error that results from multiplication of new **b** vectors by \mathbf{A}^{-1} make the Gauss-Jordan method less attractive than other algorithms.

REFERENCES:

Nash 1990, 76; Harris and Stocker 1998, 441-445; Press et al. 1988, 41-43