A Space Efficient Flexible Pivot Selection Approach to Evaluate Determinant and Inverse of a Matrix

Hafsa Athar Jafree[1]*, Muhammad Imtiaz[1], Syed Inayatullah[1], Fozia Hanif Khan[2], Tajuddin Nizami[3]

1 Department of Mathematical Sciences, University of Karachi, Karachi, Pakistan, **2** Department of Mathematics, Sir Syed University of Engineering and Technology, Karachi, Pakistan, **3** Department of Mathematics, Iqra University, Karachi, Pakistan

Abstract

This paper presents new simple approaches for evaluating determinant and inverse of a matrix. The choice of pivot selection has been kept arbitrary thus they reduce the error while solving an ill conditioned system. Computation of determinant of a matrix has been made more efficient by saving unnecessary data storage and also by reducing the order of the matrix at each iteration, while dictionary notation [1] has been incorporated for computing the matrix inverse thereby saving unnecessary calculations. These algorithms are highly class room oriented, easy to use and implemented by students. By taking the advantage of flexibility in pivot selection, one may easily avoid development of the fractions by most. Unlike the matrix inversion method [2] and [3], the presented algorithms obviate the use of permutations and inverse permutations.

Citation: Jafree HA, Imtiaz M, Inayatullah S, Khan FH, Nizami T (2014) A Space Efficient Flexible Pivot Selection Approach to Evaluate Determinant and Inverse of a Matrix. PLoS ONE 9(2): e87219. doi:10.1371/journal.pone.0087219

Editor: Gerardo Adesso, University of Nottingham, United Kingdom

Received July 26, 2013; **Accepted** December 25, 2013; **Published** February 3, 2014

Funding: The authors have no support or funding to report.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: hajafree@uok.edu.pk

Introduction

The term determinant is originally associated with the system of linear equations. It provides us with a foresight about the nature of solution of a given system of linear equations.

Maclaurin [4] published the first result on determinant of 2×2 and 3×3 system, which was generalized for $n\times n$ systems by Cramer [5]. Later, the well known Laplace expansion for evaluating determinant was proposed by Laplace, but he had then used the term 'resultant' instead of determinant. The first use of the term determinant in the modern context was done by Cauchy [6]. In 1866, Dodgson presented another method for finding the determinant of $n\times n$ systems which he named as the "Method of condensation" [7]. For larger systems the most preferred method for evaluating the determinant up till now is the Gaussian method [8]. It deals with the problem by converting the coefficient matrix into its equivalent upper/lower triangular form. The product of the pivot elements gives the determinant.

The existence of matrix inverse also depends on its determinant. The method of finding the inverse by Gaussian method is discussed later in this paper. Ahmed & Khan [2], and Khan, Shah, & Ahmad [3] proposed algorithms for the calculation of inverse of a matrix which are streamlined forms of the Gaussian method and also require permutation and inverse permutations. In this paper, we have presented two new algorithms related to the evaluation of determinant and inverse of a matrix. The first presented algorithm evaluates the determinant and is more efficient than Gaussian method as it reduces the order of the matrix at each iteration thereby saving unnecessary computations. In the second algorithm, we have presented another easy to mange way to calculate the inverse of the matrix by constructing the dictionary of the given system and thus excluding the need of permutations and inverse permutations.

Determinant of A Matrix: A Brief Review

Determinant of a square matrix $\mathbf{A}$, denoted by $det(\mathbf{A})$, is basically a real valued function. Because of its useful relationship with the matrix $\mathbf{A}$ and the solution to the system of equations of the form $\mathbf{A}x=b$, it becomes essential to have a knowledge about determinants while studying matrices. Evaluation of a determinant through its cofactor expansion (also known as Laplace expansion) is famous for lower order $n\times n$ matrices.

$$\text{Consider } \mathbf{A}=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}$$

Let M_{ij} be the minor of entry $a_{ij},(i\in\{1,2,\cdots,n\},j\in\{1,2,\cdots,n\})$ which is the determinant of the sub-matrix obtained after deleting the i^{th} row and j^{th} column of $\mathbf{A}$.

If j^{th} column of $\mathbf{A}$ is opted for cofactor expansion then,

$$det\,(\mathbf{A})=a_{1j}C_{1j}+a_{2j}C_{2j}+\cdots+a_{nj}C_{nj}$$

where, C_{ij} is the cofactor of entry a_{ij} such that $C_{ij}=(-1)^{i+j}M_{ij}$.

Similarly the cofactor expansion along the i^{th} row would be

$$det(\mathbf{A}) = a_{i1}C_{i1} + a_{i2}C_{i2} + \cdots + a_{in}C_{in}$$

For a matrix of order n, the evaluation of determinant by the above cofactor expansion requires computing n determinants of matrices of order $(n-1)$. Therefore it can be implemented for finding determinant of a matrix of order 2 or 3 with ease but for higher orders it becomes a tedious job. To reduce the computational effort usually the following three basic row operations of matrices are incorporated to evaluate the determinant [9], the method is known as evaluation of determinant by row reduction (also known as Gaussian method).

Elementary row operations

Let $\mathbf{A}$ be an $n \times n$ matrix, the following elementary row operations can be applied

a) Multiply a row by a non-zero constant.
b) Interchange two rows.
c) Add a multiple of one row to another row.

Effect of row operations on the value of determinant: [9]

Let $\mathbf{A}$ be an $n \times n$ matrix then

a) If $\mathbf{B}$ is the matrix that results when a single row or single column of $\mathbf{A}$ is multiplied by a scalar k, then $det(\mathbf{B}) = k\, det(\mathbf{A})$.
b) If $\mathbf{B}$ is the matrix that results when a two rows or two columns of $\mathbf{A}$ are interchanged then $det(\mathbf{B}) = -det(\mathbf{A})$.
c) If $\mathbf{B}$ is the matrix that results when a multiple of one row of $\mathbf{A}$ is added to another row or when a multiple of one column is added to another column, then $det(\mathbf{B}) = det(\mathbf{A})$.

Lemma: [9]

If $\mathbf{A}$ is an $n \times n$ triangular matrix (upper triangular, lower triangular or diagonal), then $det(\mathbf{A})$ is the product of the entries on the main diagonal of the matrix; that is, $det(\mathbf{A}) = a_{11}a_{22}\ldots a_{nn}$.

Evaluation of determinant by row reduction

The essence of the method is to transform the given matrix into its upper/lower triangular form by applying elementary row operations. The determinant can then be computed by incorporating the properties defined above in 'Effect of row operations' and the lemma.

Example.

$$\text{Consider the matrix } \mathbf{A} = \begin{bmatrix} 0 & 1 & 5 \\ 3 & -6 & 9 \\ 2 & 6 & 1 \end{bmatrix}$$

$$det(\mathbf{A}) = det\begin{bmatrix} 0 & 1 & 5 \\ 3 & -6 & 9 \\ 2 & 6 & 1 \end{bmatrix}$$

$$= -\,det\begin{bmatrix} 2 & 6 & 1 \\ 3 & -6 & 9 \\ 0 & 1 & 5 \end{bmatrix} \quad \text{property (b)}$$

$$= -2\,det\begin{bmatrix} 1 & 3 & \frac{1}{2} \\ 3 & -6 & 9 \\ 0 & 1 & 5 \end{bmatrix} \quad \text{property (a)}$$

$$= -2\,det\begin{bmatrix} 1 & 3 & \frac{1}{2} \\ 0 & -15 & \frac{15}{2} \\ 0 & 1 & 5 \end{bmatrix} \quad \text{property (c)}$$

$$= (-2)(-15)\,det\begin{bmatrix} 1 & 3 & \frac{1}{2} \\ 0 & 1 & -\frac{1}{2} \\ 0 & 1 & 5 \end{bmatrix} \quad \text{property (a)}$$

$$= (-2)(-15)\,det\begin{bmatrix} 1 & 3 & \frac{1}{2} \\ 0 & 1 & -\frac{1}{2} \\ 0 & 1 & \frac{11}{2} \end{bmatrix} \quad \text{property (c)}$$

$$= (-2)(-15)(11/2) \quad \text{lemma 3}$$

$$= 165$$

The New Method

Approach of row reduction may involve all the elementary row operations, as illustrated in above example. Here we are defining an operation (say pivot operation for evaluation of determinant) which consists of $(n-1)$ applications of row operation (c) only, and can be employed to evaluate the determinants avoiding involvement of row operations (a) and (b).

For example consider the following matrix,

$$\begin{bmatrix} a_{11} & \cdots & a_{1j} & \cdots & a_{1n} \\ \vdots & & \vdots & & \vdots \\ a_{i1} & \cdots & a_{ij} & & a_{in} \\ \vdots & & \vdots & & \vdots \\ a_{n1} & \cdots & a_{nj} & \cdots & a_{nn} \end{bmatrix}$$

Selecting a non-zero pivot element located at any arbitrary location (i,j), say a_{ij}, and performing row operation (c) to make the remaining elements of the j^{th} column zero we get

$$\begin{bmatrix} a_{11}-\frac{a_{i1}}{a_{ij}}\times a_{1j} & \cdots & a_{1j}-\frac{a_{ij}}{a_{ij}}\times a_{1j} & \cdots & a_{1n}-\frac{a_{in}}{a_{ij}}\times a_{1j} \\ \vdots & & \vdots & & \vdots \\ a_{i1} & \cdots & a_{ij} & \cdots & a_{in} \\ \vdots & & \vdots & & \vdots \\ a_{n1}-\frac{a_{i1}}{a_{ij}}\times a_{nj} & \cdots & a_{nj}-\frac{a_{ij}}{a_{ij}}\times a_{nj} & \cdots & a_{nn}-\frac{a_{in}}{a_{ij}}\times a_{nj} \end{bmatrix}$$

$$\begin{bmatrix} a_{11}-\frac{a_{i1}}{a_{ij}}\times a_{1j} & \cdots & 0 & \cdots & a_{1n}-\frac{a_{in}}{a_{ij}}\times a_{1j} \\ \vdots & & \vdots & & \vdots \\ a_{i1} & \cdots & a_{ij} & \cdots & a_{in} \\ \vdots & & \vdots & & \vdots \\ a_{n1}-\frac{a_{i1}}{a_{ij}}\times a_{nj} & \cdots & 0 & \cdots & a_{nn}-\frac{a_{in}}{a_{ij}}\times a_{nj} \end{bmatrix}$$

Here we can see that the pivot row remains unchanged while the $(p,k)^{th}$ element of the matrix, $a'_{pk}=a_{pk}-\frac{a_{pj}}{a_{ij}}\times a_{ik}$, where a_{pj} and a_{ik} are the elements corresponding to the pivot element in the pivot column and pivot row respectively.

Cofactor expansion along the j^{th} column will give

$$det\,(\mathbf{A})=a_{ij}C_{ij}=(-1)^{i+j}a_{ij}M_{ij}$$

$$det\,(\mathbf{A})=(-1)^{i+j}a_{ij}\;det\begin{bmatrix} a_{11}-\frac{a_{i1}}{a_{ij}}\times a_{1j} & \cdots & a_{1n}-\frac{a_{in}}{a_{ij}}\times a_{1j} \\ \vdots & \ddots & \vdots \\ a_{n1}-\frac{a_{i1}}{a_{ij}}\times a_{nj} & \cdots & a_{nn}-\frac{a_{in}}{a_{ij}}\times a_{nj} \end{bmatrix}$$

The above procedure results in a computation of determinant of a matrix with reduced order of $(n-1)$. A repetition of the above procedure will eventually give a 2×2 determinant. For a $n\times n$ matrix there must be n non-zero pivot elements. The product of respective pivot elements gives the value of the determinant. However, if at any step there is no non-zero pivot exists, we can deduce that the determinant of the given matrix is 0.

Problem 1

Find determinant of the matrix $\mathbf{A}=[a_{ij}]$ of order n.

Algorithm 1

```
Step 1: Set $d := 1$,
Step 2: set $P := \{1,2,3,\ldots\ldots,n\}$
Step 3: Select any $p \in P$ such that $L := \{k{:}\ a_{pk} \neq 0,\ k \in P\}$.
Step 4: If $L = \varphi$ then $d = 0$ and go to step 7
    Otherwise $m := a_{pk}$ $p \in P$, $k \in L$, and
    $a_{ij} := a_{ij} - \frac{a_{ik} \times a_{pj}}{m} \quad \forall i \in P-\{p\}$ and $j \in P-\{k\}$
Step 5: Reduce the order of $\mathbf{A}$ by removing $p^{th}$ row and $k^{th}$ column of $\mathbf{A}$. Also set $n := n-1$
Step 6: $d := (-1)^{p+k} \times m \times d$. If $n \neq 0$ go to step 2.
Step 7: $det(\mathbf{A}) = d$. Exit.
```

Example

$$\text{Consider the matrix}\begin{bmatrix} 2 & 5 & 3 & 2 \\ 4 & 10 & 1 & 7 \\ 1 & 5 & 2 & 2 \\ 2 & 1 & 2 & 1 \end{bmatrix}$$

Iteration 1:

$P := \{1,2,3,4\}$, here we take $p=1$ so, $L := \{1,2,3,4\}$. Taking $k=1$

$$\mathbf{A} := \begin{bmatrix} 0 & -5 & 3 \\ \frac{5}{2} & \frac{1}{2} & 1 \\ -4 & -1 & -1 \end{bmatrix}$$

$$d := (-1)^{1+1}\times 2 = 2$$

Iteration 2:

$P := \{1,2,3\}$, here we take $p=1$ so, $L := \{2,3\}$. Taking $k=2$

$$\mathbf{A} := \begin{bmatrix} \frac{5}{2} & \frac{13}{10} \\ -4 & \frac{-8}{5} \end{bmatrix}$$

$$d := (-1)^{1+2}\times(-5)\times 2 = 10$$

Iteration 3:

$P := \{1,2\}$, here we take $p=1$ so, $L := \{1,2\}$. Taking $k=1$

$$\mathbf{A} := [\frac{12}{25}]$$

$$d := (-1)^{1+1}\times(\frac{5}{2})\times 10 = 25$$

Iteration 4:

$P := \{1\}$, here we take $p=1$ so, $L := \{1\}$. Taking $k=1$

$$d := (-1)^{1+1}\times(\frac{12}{25})\times 25 = 12$$

Hence determinant of given matrix is 12.

Comparison with row reduction method

The above example illustrates the efficiency of the algorithm in terms of memory storage and the number of elements computed. If a matrix of order 4 is solved by row reduction method the number of elements computed will be 12, 6 and 2 in first, second and third iterations respectively. However we have computed 9, 4 and 1 elements in the respective iterations. So the total number of element computations needed for row reduction method is 20, but our method needs only 14 element computations. Also, at each iteration the size of the matrix has been reduced. Row reduction method needs to store 16 elements for each iteration, hence on the whole method requires 48 elements to be stored in the memory, while in contrast our algorithm stores $16+9+4+1=30$ elements, which is a noteworthy reduction in terms of storage requirement.

A comparison of the number of elements computed and stored to evaluate the determinant of different orders by row reduction and our algorithm are shown in table 1.

Inverse of A Matrix: A Brief Review

Consider a matrix $\mathbf{A}=[a_{ij}]$ of order n. To evaluate the inverse of the matrix, say $\mathbf{B}$, one must solve the following n system of equations, for $x_1, x_2, \ldots, x_n$

$$\begin{array}{cccccc} a_{11}x_1 & a_{12}x_2 & \cdots & a_{1n}x_n & = & y_1 \\ a_{21}x_1 & a_{22}x_2 & \cdots & a_{2n}x_n & = & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ a_{n1}x_1 & a_{n2}x_2 & \cdots & a_{nn}x_n & = & 0 \end{array} \quad \text{system } (1)$$

$$\begin{array}{cccccc} a_{11}x_1 & a_{12}x_2 & \cdots & a_{1n}x_n & = & 0 \\ a_{21}x_1 & a_{22}x_2 & \cdots & a_{2n}x_n & = & y_2 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ a_{n1}x_1 & a_{n2}x_2 & \cdots & a_{nn}x_n & = & 0 \end{array} \quad \text{system } (2)$$

$$\vdots$$

$$\vdots$$

$$\begin{array}{cccccc} a_{11}x_1 & a_{12}x_2 & \cdots & a_{1n}x_n & = & 0 \\ a_{21}x_1 & a_{22}x_2 & \cdots & a_{2n}x_n & = & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ a_{n1}x_1 & a_{n2}x_2 & \cdots & a_{nn}x_n & = & y_n \end{array} \quad \text{system } (n)$$

The solutions of the above system are

From system (1) : $x_1=b_{11}y_1, x_2=b_{21}y_2, \cdots, x_n=b_{n1}y_n$
From system (2) : $x_1=b_{12}y_1, x_2=b_{22}y_2, \cdots, x_n=b_{n2}y_n$

$$\vdots$$

From system (n) : $x_1=b_{1n}y_1, x_2=b_{2n}y_2, \cdots, x_n=b_{nn}y_n$

Then the matrix $\mathbf{B}=\begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} \end{bmatrix}$ is called the inverse of matrix $\mathbf{A}$.

This procedure could be compactly preformed by expressing the above system in the augmented form,

$$\mathbf{A}x=\mathbf{I}y \tag{1}$$

where $x=[x_1 \quad x_2 \quad \cdots \quad x_n]^T$ $y=[y_1 \quad y_2 \quad \cdots \quad y_n]^T$ and $\mathbf{I}$ is identity matrix of order n.

If $\mathbf{A}$ is invertible, applying successive elementary row operations of Gaussian method yields,

$$\mathbf{I}x=\mathbf{A}^{-1}y.$$

The New Approach Based on Dictionary Notation

Now consider again the Equation (1) in expanded form,

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

Usually the following augmented matrix is used to solve above system,

$$\begin{array}{cccccccc} x_1 & x_2 & \cdots & x_n & y_1 & y_2 & \cdots & y_n \end{array}$$
$$\left[\begin{array}{cccccccc} a_{11} & a_{12} & \cdots & a_{1n} & 1 & 0 & \cdots & 0 \\ a_{21} & a_{22} & \cdots & a_{2n} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & 0 & 0 & \cdots & 1 \end{array}\right]$$

Table 1. Comparison of row reduction method with algorithm 1 for evaluating determinant of matrix.

Order	Row reduction method		Algorithm 1	
	No. of Element Computations	Storage Requirements (no. of elements)	No. of Element Computations	Storage Requirements (no. of elements)
2	2	8	1	5
4	20	48	14	30
5	40	125	30	55
7	112	343	94	140
10	330	1000	285	385
12	572	1728	506	650

doi:10.1371/journal.pone.0087219.t001

Here, we are using the concept of dictionary notation developed by [1]. Now basic variables are the variables whose coefficients are in the form of any column of identity matrix, and a basis is collection of all basic variables. We can see that basis for the above matrix $B=\{y_1,y_2,\cdots,y_n\}$ and we may consider $N=\{x_1,x_2,\cdots,x_n\}$ as the non-basis of the current matrix. The objective is to convert the basic variables into non-basic variables and vice versa by using pivot operations. Using the dictionary concept defined by [1] we can remove the basic columns from the matrix and construct the following dictionary form with basis B and non-basis $\mathcal{N}$: (Note basic variables are shown in left most column and non basic variables in the top row of the dictionary).

$$\begin{array}{c|cccc|} & x_1 & x_2 & \cdots & x_n \\ \hline y_1 & a_{11} & a_{12} & \cdots & a_{1n} \\ y_2 & a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & & \vdots \\ y_n & a_{n1} & a_{n2} & \cdots & a_{nn} \end{array}$$

Pivot operation for evaluating inverse

The following pivot operations [1] may be applied to enter x_i into basis B and y_j into non-basis $\mathcal{N}$,

i) Divide the pivot row by pivot element and the pivot column by negative of the pivot element (except the pivot element).

ii) The remaining $(n-1)^2$ elements are determined by the formula as defined in the new method of determinant.

iii) Reciprocate the pivot element.

If the number of pivot elements is equal to the order of the matrix the resulting matrix gives the inverse otherwise we may conclude that the inverse does not exist.

Problem 2

Find the inverse of matrix $\mathbf{A}=[a_{ij}]$ of order $n\times n$.

Algorithm 2

Step 1: Set $H:=\{1,2,3,\ldots\ldots,n\}$, $B:=\{y_h\}$ and $\mathcal{N}:=\{x_h\}$, $\forall h\in H$. Construct dictionary of the matrix $\mathbf{A}$, i.e. $D(\mathbf{A})$.

Step 2: Set $P:=\{p{:}y_p\in B\}$

Step 3: If $P=\varphi$, go to step 6.

Otherwise, $L:=\{k{:}\ a_{pk}\neq 0,\ x_k\in\mathcal{N}\}$

Step 4: If $L=\varphi$ then inverse does not exist. Exit

Otherwise, for any $p\in P,k\in L$

$$m_i := -\frac{a_{ik}}{a_{pk}}\ \forall i\in H-\{p\}$$

$$a_{ij} := a_{ij}+a_{pj}\times m_i\ \forall i\in H-\{p\},\forall j\in H-\{k\}$$

$$a_{pj} := \frac{a_{pj}}{a_{pk}}\ \forall j\in H-\{k\}$$

$$a_{ik} := m_i\ \forall i\in H-\{p\}$$

$$a_{pk} := \frac{1}{a_{pk}}$$

Step 5: $B:=B+\{x_k\}-\{y_p\}$ and $N:=N-\{x_k\}+\{y_p\}$. Update $D(\mathbf{A})$ and go to step 2.

Step 6: $Inv(\mathbf{A})=[a_{ij},\forall i,j\in H : x_i\in B,y_j\in N]$ Exit.

Example

$$\text{Consider } \mathbf{A}=\begin{bmatrix} 2 & 5 & 3 & 2 \\ 5 & 10 & 1 & 7 \\ 3 & 5 & 2 & 2 \\ 2 & 1 & 2 & 1 \end{bmatrix}$$

$$D(\mathrm{A})=\begin{array}{c|cccc|} & x_1 & x_2 & x_3 & x_4 \\ \hline y_1 & 2 & 5 & 3 & 2 \\ y_2 & 5 & 10 & 1 & 7 \\ y_3 & 3 & 5 & 2 & 2 \\ y_4 & 2 & 1 & 2 & 1 \end{array}$$

Here $\mathcal{N}:=\{x_1,x_2,x_3,x_4\}$, $B:=\{y_1,y_2,y_3,y_4\}$

Iteration 1:

$H:=\{1,2,3,4\}$, $P:=\{1,2,3,4\}$, taking $p=1$ we get $L:=\{1,2,3,4\}$. Taking $k=1$

$$D(A)=\begin{array}{c|cccc|} & y_1 & x_2 & x_3 & x_4 \\ \hline x_1 & \frac{1}{2} & \frac{5}{2} & \frac{3}{2} & 1 \\ y_2 & -2 & 0 & -5 & 3 \\ y_3 & -\frac{1}{2} & \frac{5}{2} & \frac{1}{2} & 1 \\ y_4 & -1 & -4 & -1 & -1 \end{array}$$

$\mathcal{N}:=\{y_1,x_2,x_3,x_4\}$, $B:=\{x_1,y_2,y_3,y_4\}$

Iteration 2:

$P:=\{2,3,4\}$, taking $p=2$ we get $L:=\{3,4\}$. Taking $k=3$

$$D(A)=\begin{array}{c|cccc|} & y_1 & x_2 & y_2 & x_4 \\ \hline x_1 & -\frac{1}{10} & \frac{5}{2} & \frac{3}{10} & \frac{19}{10} \\ x_3 & \frac{2}{5} & 0 & -\frac{1}{5} & -\frac{3}{5} \\ y_3 & -\frac{7}{10} & \frac{5}{2} & \frac{1}{10} & \frac{13}{10} \\ y_4 & -\frac{3}{5} & -4 & -\frac{1}{5} & -\frac{8}{5} \end{array}$$

$\mathcal{N}:=\{y_1,x_2,y_2,x_4\}$, $B:=\{x_1,x_3,y_3,y_4\}$

Iteration 3:

$P:=\{3,4\}$, taking $p=3$ we get $L:=\{2,4\}$. Taking $k=2$

Table 2. Comparison of Gaussian method with the algorithm 2 for evaluating inverse of matrix.

Order	Gaussian Method		Algorithm 2	
	No. of Element Computations	Storage Requirements (no. of elements)	No. of Element Computations	Storage Requirements (no. of elements)
2	12	8	8	4
4	80	32	64	16
5	150	50	125	25
7	392	98	343	49
10	1100	200	1000	100
12	1872	288	1728	144

doi:10.1371/journal.pone.0087219.t002

$$D(A)=\begin{array}{c|cccc} & y_1 & y_3 & y_2 & x_4 \\ x_1 & \frac{3}{5} & -1 & \frac{1}{5} & \frac{3}{5} \\ x_3 & \frac{2}{5} & 0 & -\frac{1}{5} & -\frac{3}{5} \\ x_2 & -\frac{7}{25} & \frac{2}{5} & \frac{1}{25} & \frac{13}{25} \\ y_4 & -\frac{43}{25} & \frac{8}{5} & -\frac{1}{25} & \frac{12}{25} \end{array}$$

$\mathcal{N}:=\{y_1,y_3,y_2,x_4\}$, $B:=\{x_1,x_3,x_2,y_4\}$

Iteration 4:

$P:=\{4\}$, taking $p=4$ we get $L:=\{4\}$. Taking $k=4$

$$D(A)=\begin{array}{c|cccc} & y_1 & y_3 & y_2 & y_4 \\ x_1 & \frac{11}{4} & -3 & \frac{1}{4} & -\frac{5}{4} \\ x_3 & -\frac{7}{4} & 2 & -\frac{1}{4} & \frac{5}{4} \\ x_2 & \frac{19}{12} & -\frac{4}{3} & \frac{1}{12} & \frac{13}{12} \\ x_4 & -\frac{43}{12} & \frac{10}{3} & -\frac{1}{12} & \frac{25}{12} \end{array}$$

$\mathcal{N}:=\{y_1,y_3,y_2,y_4\}$, $B:=\{x_1,x_3,x_2,x_4\}$

Now place the elements with respect to indices of variables in B and $\mathcal{N}$.

For example Here $H:=\{1,2,3,4\}$, So $x_1 \in B$ and $y_1 \in N$, implies $a_{11}=\frac{11}{4}$. Also $x_1 \in B$ and $y_3 \in N$ implies $a_{13}=-3$. Similarly placing the remaining elements we get

$$D(A)=\begin{array}{c|cccc} & y_1 & y_2 & y_3 & y_4 \\ x_1 & \frac{11}{4} & \frac{1}{4} & -3 & -\frac{5}{4} \\ x_2 & \frac{19}{12} & \frac{1}{12} & -\frac{4}{3} & -\frac{13}{12} \\ x_3 & -\frac{7}{4} & -\frac{1}{4} & 2 & \frac{5}{4} \\ x_4 & -\frac{43}{12} & -\frac{1}{12} & \frac{10}{3} & \frac{25}{12} \end{array}$$

Comparison with Gaussian method

The above example illustrates the efficiency of the algorithm in terms of memory storage and the number of elements computed. If a matrix of order 4 is solved by Gaussian method the number of elements computed will be 20 at each iteration. However our method requires computation of 16 elements at each iteration. So the total number of element computations needed for Gaussian method is 80, on the other hand for our method it is 64. Also the Gaussian method needs to store 32 elements for each iteration, hence on the whole Gaussian method requires 128 elements to be stored in the memory, while in contrast our algorithm stores 64 elements, which is a noteworthy reduction in terms of storage requirement.

A comparison of the number of elements computed and stored to evaluate the inverse by Gaussian method and our algorithm are shown in table 2.

Applications

Matrix determinant and inverse have applications in various fields like mathematics, economics, physics, biology etc. Solving different models like population growth involve the use of matrix determinant and inverse. Matrix inverse and determinant are also used in cryptography [10]. Linear transformations (rotation, reflection, translation etc.) involve the calculation of matrix inverse. Matrix inverse and determinant are also employed in operations research while solving linear programs, revised simplex method and markov chains. Determinants of order 3 are used to find area of triangles and for testing colinearity of points. Least square analysis of data requires the evaluation of matrix inverse [11]. p-dimensional volume of parallelepiped in $\Re^m$ is determined by computing determinant [12].

Conclusion

This paper presented easy algorithms for computations of determinant and inverse of a matrix. Since the order of the given matrix has been reduced at each step while calculating its determinant, the algorithm reduces the storage requirement (as exhibited in the example). The calculation of inverse has been done using the dictionary notation which obviates the use of permutations and makes it easier to cope with in class room teaching. Ill conditioned system can also be handled as the selection of pivots has been kept arbitrary, thus improving the numerical accuracy of the system.

Author Contributions

Conceived and designed the experiments: SI. Performed the experiments: FHK TN. Wrote the paper: HAJ. Model formulation: HAJ. Algorithm designing: HAJ MI SI.

References

1. Chvatal V (1983) Linear Programming. United States of America: W.H. Freeman and Company.
2. Ahmad F, Khan H (2010) An efficient and simple Algorithm for matrix inversion. International Journal of Technology Diffusion, 20–27.
3. Khan H, Shah IA, Ahmad F (2010) An efficient and generic algorithm for matrix inversion. Internantional Jounal of Technology diffusion, 36–41.
4. Maclaurin C (1748) A Treatise of Algebra. London: A. Millar and J.Nourse.
5. Cramer G (1750) Introduction to the analysis of algebraic curves.
6. Augustin CL (1815) Memoire sur les fonctions qui ne peuvent obtenir que deux valeurs égales et des signes contraires par suite des transpositions operées entre les variables qu'elles renferment. de l'École polytechnique.
7. Rice A, Torrence E (2006) Lewis Caroll's Condensation method for evaluating determinants. Math Horizons, 12–15.
8. Higham NJ (2011) Gaussian Elimination. John Wiley & Sons, Inc., 230–238.
9. Anton H, Rorres C (2005) Elementary Linear Algebra. John, Wiley & Sons, Inc.
10. Vellaikannan B, Mohan DV, Gnanaraj V (2010) A note on the application of quadratic forms in coding theory with a note on security. Int J Comp Tech Appl, p. 78–87.
11. Greenberg G, Sarhan AE (1959) Matrix inversion, its interest and application in analysis of data. Journal of the American Statistical Association, 755–766.
12. Kuttler K (2013) Linear Algebra, Theory and Applications. www.math.byu.edu/klkuttle/linearalgebra.pdf.