CSC336 Tutorial 3 – Matrices, operation counts, GE/LU

QUESTION 1 *Show that the product of lower triangular (l.t.) matrices is a lower triangular matrix.*

PROOF: First consider a l.t. matrix L of size $n \times n$ and a $n \times 1$ vector x, whose first k components are 0. Which components of Lx are 0?

$$\begin{pmatrix} l_{1,1} & 0 & . & . & . & 0 \\ l_{2,1} & l_{2,2} & 0 & . & . & 0 \\ l_{3,1} & l_{3,2} & l_{3,3} & 0 & . & 0 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ l_{n,1} & l_{n,2} & l_{n,3} & . & . & l_{nn} \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ x_{k+1} \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} \sum_{j=1}^{n} l_{1,j}x_j \\ \sum_{j=1}^{n} l_{2,j}x_j \\ \vdots \\ \sum_{j=1}^{n} l_{i,j}x_j \\ \vdots \\ \sum_{j=1}^{n} l_{n,j}x_j \end{pmatrix} =^{(\dagger)} \begin{pmatrix} \sum_{j=k+1}^{n} l_{1,j}x_j \\ \sum_{j=k+1}^{n} l_{2,j}x_j \\ \vdots \\ \sum_{j=k+1}^{n} l_{i,j}x_j \\ \vdots \\ \sum_{j=k+1}^{n} l_{n,j}x_j \end{pmatrix}$$

$$\begin{matrix}\text{[because } (\dagger)\ x_j = 0, \forall j \le k \\ \text{and } (\dagger\dagger)\ l_{i,j} = 0, \forall i < j]\end{matrix} =^{(\dagger\dagger)} \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \sum_{j=k+1}^{n} l_{k+1,j}x_j \\ \vdots \\ \sum_{j=k+1}^{n} l_{nj}x_j \end{pmatrix}$$

Thus, components $1, \dots, k$ of Lx are 0.

Now consider two l.t. matrices L_1, L_2. Column j of the product L_1L_2 is the product of L_1 times the jth column of L_2, the latter being a vector, whose first $j-1$ components are 0. Therefore the first $j-1$ elements in column j of L_1L_2 are 0. Since this holds for any $j = 1, \dots, n$, the product of L_1, L_2 is l.t.

General note: Assume A and B have appropriate dimensions, so that the matrix-matrix product AB is defined. The jth column of the matrix-matrix product AB is the matrix-vector product of A with the jth column of B.

QUESTION 2 *Show that the product of unit lower triangular (u.l.t.) matrices is a unit lower triangular matrix.*

PROOF: First consider a u.l.t. matrix L of size $n \times n$ and a $n \times 1$ vector x, whose first k components are 0, and whose $k+1$st component is 1, i.e. $x_{k+1,k+1} = 1$.

As in Question 1, components $1, \dots, k$ of Lx are 0. Moreover,

$(Lx)_{k+1,k+1} = \sum_{j=k+1}^{n} l_{k+1,j}x_j =^{(\dagger)} l_{k+1,k+1}x_{k+1} =^{(\dagger\dagger)} 1$, because

$(\dagger) l_{k+1,j} = 0, \forall j > k+1$ and

$(\dagger\dagger) l_{k+1,k+1} = 1, x_{k+1} = 1$.

Now consider two u.l.t. matrices L_1, L_2. As in Question 1, the first $j-1$ elements in column j of L_1L_2 are 0. Moreover, from $(Lx)_{k+1,k+1} = 1$ shown above, the jth element in column j of L_1L_2 is 1. Since this holds for any $j = 1, \dots, n$, the product of L_1, L_2 is u.l.t.

QUESTION 3 *Show that the inverse of a l.t. matrix is a l.t. matrix.*

PROOF: Consider a l.t. matrix L and let $X = L^{-1}$. We have $LX = \mathbf{I}$

$$\Rightarrow L \cdot \begin{pmatrix} x_{1,1} & x_{1,2} & . & . & x_{1,n} \\ x_{2,1} & x_{2,2} & . & . & x_{2,n} \\ x_{3,1} & x_{3,2} & . & . & x_{3,n} \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ x_{n,1} & x_{n,2} & . & . & x_{nn} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & . & 0 \\ 0 & 1 & \dots & . & 0 \\ . & . & \dots & . & . \\ . & . & \dots & . & . \\ . & . & \dots & . & . \\ 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix} \quad \text{Let } e_j = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1_{(j\text{th row})} \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Note that e_j is the jth column of $\mathbf{I}$ (identity).

Let also $X_{*,j}$ be the jth column of X.

Then $LX_{*,j} = e_j, \forall j = 1, \dots, n$.

Thus, by solving $LX_{*,j} = e_j, j = 1, \dots, n$, we can compute X.

Since L is l.t., the systems $LX_{*,j} = e_j, j = 1, \dots, n$, are solved by forward substitution.

Consider applying forward substitution to $LX_{*,j} = e_j$

$$\begin{pmatrix} l_{1,1} & 0 & . & . & . & 0 \\ l_{2,1} & l_{2,2} & 0 & . & . & 0 \\ l_{3,1} & l_{3,2} & l_{3,3} & 0 & . & 0 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ l_{n,1} & l_{n,2} & l_{n,3} & . & . & l_{n,n} \end{pmatrix} \begin{pmatrix} x_{1,j} \\ x_{2,j} \\ . \\ . \\ . \\ . \\ x_{nj} \end{pmatrix} = \begin{pmatrix} 0 \\ . \\ 0 \\ 1_{(j\text{th row})} \\ 0 \\ . \\ 0 \end{pmatrix} \Rightarrow$$

$$\begin{aligned} x_{1,j} &= 0/l_{1,1} = 0 \\ x_{2,j} &= (0 - l_{2,1}x_{1,j})/l_{2,2} = 0 \\ &\vdots \\ x_{j-1,j} &= (0 - l_{j-1,1} \cdot 0 - l_{j-1,2} \cdot 0 - \ldots - l_{j-1,j-2} \cdot 0)/l_{j-1,j-1} = 0 \\ x_{j,j} &= (1 - 0 - \ldots)/l_{j,j} = 1/l_{j,j} \\ x_{j+1,j} & \quad \text{possibly non zero} \\ &\vdots \end{aligned}$$

Thus, $x_{i,j} = 0, i = 1, \ldots, j-1$, and this holds $\forall j$. And thus, $x_{i,j} = 0, \forall i < j$, i.e. X is l.t.

QUESTION 4 *Show that the inverse of a u.l.t. matrix is a u.l.t. matrix.*

PROOF: Consider a u.l.t. matrix L and let $X = L^{-1}$. We have $LX = \mathbf{I}$, as in Question 3. Consider applying forward substitution to $LX_{*,j} = e_j$. The results of Question 3 still hold, i.e. X is l.t., and, moreover, $x_{j,j} = 1/l_{j,j} = 1/1 = 1$. Since this holds for $j = 1, \ldots, n$, X is u.l.t.

QUESTION 5 *Obtain the operation counts for computing the product of matrices $C = A \cdot B$, where $A \in \mathcal{R}^{l \times m}$, $B \in \mathcal{R}^{m \times n}$, by the standard matrix-matrix multiplication algorithm,*

```
for i = 1 to l do
    for j = 1 to n do
        c(i,j) = 0
        for k = 1 to m do
            c(i,j) = c(i,j) + a(i,k)*b(k,j)
        endfor
    endfor
endfor
```

ANSWER: Note: $c_{i,j} = \sum_{k=1}^{m} a_{i,k} b_{k,j}$, $C \in \mathcal{R}^{l \times n}$.

Operation counts: $l \cdot n \cdot m$ flops. (1 flop = 1 addition + 1 multiplication)

QUESTION 6 *Obtain the operation counts for computing $A \cdot B \cdot C$ as $(A \cdot B) \cdot C$ or as $A \cdot (B \cdot C)$, where $A \in \mathcal{R}^{1 \times n}$, $B \in \mathcal{R}^{n \times 1}$, $C \in \mathcal{R}^{1 \times n}$.*

ANSWER:

1. $(A \cdot B) \cdot C$:
   $A \cdot B \rightarrow 1 \cdot n \cdot 1 = n$ flops,
   $(AB) \cdot C \rightarrow 1 \cdot 1 \cdot n = n$ flops, $2n$ flops in total.
2. $A \cdot (B \cdot C)$:
   $B \cdot C \rightarrow n \cdot 1 \cdot n = n^2$ flops,
   $A \cdot (BC) \rightarrow 1 \cdot n \cdot n = n^2$ flops, $2n^2$ flops in total.

For $n > 1$, $(A \cdot B) \cdot C$ is much faster than $A \cdot (B \cdot C)$.

QUESTION 7 *Obtain the operation counts for computing $A \cdot B \cdot C$ as $(A \cdot B) \cdot C$ or as $A \cdot (B \cdot C)$, where $A \in \mathcal{R}^{k \times l}$, $B \in \mathcal{R}^{l \times m}$, $C \in \mathcal{R}^{m \times n}$.*

ANSWER:

1. $(A \cdot B) \cdot C$:
   $A \cdot B \to k \cdot l \cdot m$ flops, $\quad AB \in \mathcal{R}^{k \times m}$
   $(AB) \cdot C \to k \cdot m \cdot n$ flops $k \cdot m \cdot (l + n)$ flops in total.
2. $A \cdot (B \cdot C)$:
   $B \cdot C \to l \cdot m \cdot n$ flops, $\quad BC \in \mathcal{R}^{l \times n}$
   $A \cdot (BC) \to k \cdot l \cdot n$ flops, $l \cdot n \cdot (k + m)$ flops in total.

In general, we cannot tell which one is faster. For particular cases, one or the other is preferred. (A dynamic programming algorithm that finds the most efficient *Matrix Chain Multiplication* may be covered in CSC373 "Algorithm Design and Analysis".)

Examples:

- $k = 3, l = 5, m = 3, n = 2, \quad 3 \cdot 3 \cdot (5 + 2) = 63 > 60 = 5 \cdot 2 \cdot (3 + 3)$.
- $k = 3, l = 3, m = 5, n = 2, \quad 3 \cdot 5 \cdot (3 + 2) = 75 > 48 = 3 \cdot 2 \cdot (3 + 5)$.
- $k = 2, l = 3, m = 5, n = 3, \quad 2 \cdot 5 \cdot (3 + 3) = 60 < 63 = 3 \cdot 3 \cdot (2 + 5)$.

QUESTION 8 *Let $A \in \mathcal{R}^{n \times n}$ be invertible. Using GE/LU, give an algorithm for computing the inverse of A, that requires n^3 flops.*

ANSWER: Let $X = A^{-1}$ be the inverse of A (invertible). Then $AX = \mathbf{I}$. Express $AX = \mathbf{I}$ as

$$A \cdot [X_{*,1} X_{*,2} \cdots X_{*,n}] = [e_1 e_2 \cdots e_n]$$

where $X_{*,j}$ is the j-th column of X, and e_j the unit vector with "1" at row j.

This means that

$$A \cdot X_{*,j} = e_j$$

So to compute X, it suffices to compute the column vectors $X_{*,j}$; i.e. to solve n linear systems with the same matrix A and different right sides, namely $e_j, j = 1, \ldots, n$. So we perform LU decomposition to A and apply forward-and-backward substitutions to $LY_{*,j} = e_j, UX_{*,j} = Y_{*,j}, j = 1, \ldots, n$.

Operation counts:

$$\frac{n^3}{3} + n \cdot 2 \cdot \frac{n^2}{2} = \frac{4n^3}{3}$$

Actually, we can do better.

Assume we compute the LU factorization of A, $A = LU$.

Apply forward substitution to $LY_{*,n} = e_n$: Unknowns $y_{1,n}$ through $y_{n-1,n}$ will be 0, thus no cost to compute them, and $y_{n,n} = 1/1 = 1$ (no cost).

Apply forward substitution to $Ly_j = e_j$: Unknowns $y_{1,j}$ through $y_{j-1,j}$ will be 0, thus no cost to compute them, and $y_{j,j} = 1/l_{j,j} = 1/1 = 1$ (no cost). Unknowns $y_{j+1,j}$ through $y_{n,j}$ require computation.

$$\begin{aligned}
y_{j+1,j} &= (0 - l_{j+1,j} y_{j,j}) / l_{j+1,j+1} = -l_{j+1,j} \\
y_{j+2,j} &= (0 - l_{j+2,j} y_{j,j} - l_{j+2,j+1} y_{j+1,j}) / l_{j+2,j+2} \\
&\vdots \\
y_{n,j} &= (0 - l_{n,j} y_{j,j} - \cdots - l_{n,n-1} y_{n-1,j}) / l_{n,n}
\end{aligned}$$

We need approximately $1 + 2 + \cdots + (n - j) = \sum_{i=1}^{n-j} i = (n - j)(n - j + 1)/2$ adds and mults (flops).

Total flops for forward substitutions:

$$\begin{aligned}
\sum_{j=1}^{n} \frac{(n-j)(n-j+1)}{2} &= \sum_{j=1}^{n-1} \frac{j(j+1)}{2} = \sum_{j=1}^{n-1} \frac{j^2}{2} + \sum_{j=1}^{n-1} \frac{j}{2} \\
&= \frac{1}{2} \left[\frac{(n-1)n(2n-1)}{6} + \frac{n(n-1)}{2} \right]
\end{aligned}$$

$$\approx \frac{n^3}{6}$$

Apply backward substitution to $UX_{*,j} = Y_{*,j}$: $n^2/2$ flops (adds and mults).

(Note: $Y_{*,j}$ does not have special structure.)

Total flops for b/s: $nn^2/2 = n^3/2$ flops.

Total flops for inverse:

$$\frac{n^3}{3} \text{ (LU) } + \frac{n^3}{6} \text{ (F/S) } + n \cdot \frac{n^2}{2} \text{ (B/S) } = n^3$$

QUESTION 9 *Given a nonsingular matrix $A \in \mathcal{R}^{n\times n}$ and a matrix $B \in \mathcal{R}^{n\times k}$, find an efficient way to compute $A^{-1}B$.*
ANSWER: Two ways of computing $A^{-1}B$:

1. Computing A^{-1} costs n^3 flops. Computing the matrix-matrix product $A^{-1}B$ costs n^2k. Total $n^3 + n^2k$ flops.

2. Alternatively, we can compute the solution to $AX = B$. Note that the solution is a matrix $X \in \mathcal{R}^{n\times k}$.

   Let $X_{*,j}$, $j = 1, \ldots, k$, be the jth column of X, and $B_{*,j}$, $j = 1, \ldots, k$, the jth column of B. Apply LU factorization to A ($n^3/3$ flops), and obtain the L and U factors of A. Then, for each $j = 1, \ldots, k$, do forward and backward substitutions $LY_{*,j} = B_{*,j}$, $UX_{*,j} = Y_{*,j}$ (n^2k flops). Total $n^3/3 + n^2k$ flops, clearly less than the flops in 1.

If k is of lesser order than n, then approach 2 is about 3 times faster than approach 1.
If k is much larger than n, then approach 2 is still faster than approach 1.
Never compute the inverse A^{-1} unless the inverse itself is explicitly required.