Brian KYANJO
Home work 5
Math 566

1. Gaussian Elimination for a Structured matrix.

Determine the number of operations.

Addition	Subtraction	Multiplication	Division
0	$n-2$	$n-2$	$n-2$
0	$n-1$	$n-1$	$n-1$
0	$n-1$	$n-1$	$n-1$
0	$n-2$	$n-2$	$n-2$
+ n	n	n	n
n	$5n-6$	$5n-6$	$5n-6$

Total operations $= n + 5n-6 + 5n-6 + 5n-6$
$= \underline{\underline{(16n-18)}}$ operations

No.2

a) Derive a fast algorithm for solving these system using a similar approach done in class.

Consider

$$\begin{bmatrix} a_1 & c_1 & e_1 & & & \\ b_1 & a_2 & c_2 & e_2 & & \\ d_1 & b_2 & a_3 & c_3 & e_3 & \\ & d_2 & b_3 & a_4 & c_4 & e_4 \\ & & & & & \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ \end{bmatrix}$$

Step 1: Upper triangular form,

for $i=1$ to $n-2$ do

$$b_i' = b_i / a_i$$

$$d_i' = \frac{d_i}{a_i}$$

$$a_{i+1} = a_{i+1} - b_i' c_i$$

$$c_{i+1} = c_{i+1} - b_i' e_i$$

$$b_{i+1} = b_{i+1} - d_i' c_i$$

$$a_{i+2} = a_{i+2} - d_i' e_i$$

$$f_{i+1} = f_{i+1} - b_i' f_i$$

$$f_{i+2} = f_{i+2} - d_i' f_i$$

end for

$$a_n = a_n - \frac{b_{n-1}}{a_{n-1}} c_{n-1}$$

$$f_n = f_n - \frac{b_{n-1}}{a_{n-1}} f_{n-1}$$

$$x_n = \frac{f_n}{a_n}$$

$$x_{n-1} = \frac{(f_{n-1} - c_{n-1} x_n)}{a_{n-1}}$$

Step 2: Back Substitution

for $i = n-2$ to -1 to 1 do

$$x_i = \frac{(f_i - c_i x_{i+1} - e_i x_{i+2})}{a_i}$$

end for

b) Determine the exact number of operations your algorithm requires for solving a general n-by-n penta diagonal system.

Additions	Subtractions	Multiplications	Divisions
$6(n-2)$	$6(n-2)$	$6(n-2)$	$2(n-2)$
0	3	3	4
0	$2(n-2)$	$2(n-2)$	$(n-2)$
+			
$6(n-2)$	$8n-13$	$8n-13$	$3n-2$

Total operations $= 6(n-2) + 8(n-13) + 8(n-13) + 3n-2)$

$= 25n - 202$ operations.

No. 4

Derive a procedure for computing the entries in $\tilde{R}$ and D for this new $A = \tilde{R}^T D \tilde{R}$ decomposition

Consider $A \in \mathbb{R}^{4\times 4}$

$$\underset{A}{\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{12} & a_{22} & a_{23} & a_{24} \\ a_{13} & a_{23} & a_{33} & a_{34} \\ a_{14} & a_{24} & a_{34} & a_{44} \end{bmatrix}} = \underset{\tilde{R}^T}{\begin{bmatrix} 1 & 0 & 0 & 0 \\ r_{12} & 1 & 0 & 0 \\ r_{13} & r_{23} & 1 & 0 \\ r_{14} & r_{24} & r_{34} & 1 \end{bmatrix}} \underset{D}{\begin{bmatrix} d_1 & 0 & 0 & 0 \\ 0 & d_2 & 0 & 0 \\ 0 & 0 & d_3 & 0 \\ 0 & 0 & 0 & d_4 \end{bmatrix}} \underset{\tilde{R}}{\begin{bmatrix} 1 & r_{12} & r_{13} & r_{14} \\ 0 & 1 & r_{23} & r_{24} \\ 0 & 0 & 1 & r_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}}$$

Step ①

$d_{11} = a_{11}$ and $r_{ij} d_{11} = a_{ij}$

$r_{ij} = \frac{1}{d_{11}}(a_{ij})$, $j = 2, 3, \dots, n$

step ②

$a_{22} = r_{12}^2 d_{11} + d_{22}$ and $r_{12} r_{1j} d_{11} + r_{2j} d_{22} = a_{2j}$

$d_{22} = a_{22} - r_{12}^2 d_{11}$ $\quad$ $r_{2j} = \frac{1}{d_{22}}(a_{2j} - r_{12} r_{1j} d_{11})$,

$j = 3, 4, 5, \dots, n$

Step ③

$a_{33} = r_{13}^2 d_{11} + r_{23}^2 d_{22} + d_{33}$

d_{33}

$$d_{33} = \left(a_{33} - \left(r_{13}^2 d_{11} + r_{23}^2 d_{22}\right)\right) \quad \text{and} \quad a_{3j} = r_{13} r_{1j} d_{11} + r_{23} r_{2j} d_{22} + r_{3j} d_{33}$$

Repeats

$$r_{3j} = \frac{1}{d_{33}}\left(a_{3j} - \left(r_{13} r_{1j} d_{11} + r_{23} r_{2j} d_{22}\right)\right)$$

$$j = 4, 5, 6, \ldots, n$$

step(n)

$$d_{nn} = \left(a_{nn} - \left(r_{1n}^2 d_{11} + r_{2n}^2 d_{22} + r_{3n}^2 d_{33} + \cdots + r_{n-1,n}^2 d_{n-1,n-1}\right)\right)$$

Pseudocode

$d_{11} = a_{11}$

for $k = 1$ to n do

$$d_{kk} = \left(a_{kk} - \sum_{i=1}^{k-1} r_{ik}^2 d_{ii}\right)$$

for $j = k+1$ to n do

$$r_{kj} = \frac{1}{d_{kk}}\left(a_{kj} - \left(\sum_{i=1}^{k-1} r_{ik} r_{ij} d_{ii}\right)\right)$$

end

end

NO 5
(Sherman-Morrison formula)

(a) Suppose you have a fast algorithm for solving $Ay = c$. Explain how to use this algorithm to design a fast algorithm for solving $(A - uv^T)x = b$

- Assume the fast algorithm in solving $Ay = c$ costs $O(n^2)$.
- To solve $(A - uv^T)x = b$ in $O(n^2)$ time, we are going to do the following.
- Since $Ax - uv^T x = b$ differs from $Ay = c$ by just subtracting uv^T for some columns vectors u and v.
- The algorithm can be obtained using Sherman-Morrison formula as ~~form~~ follows:

1. Solve $Az = b \Rightarrow z = A^{-1}b$
2. Solve $Ay = u \Rightarrow y = A^{-1}u$

$\Big\}\ O(n^2)$

3. Compute $\alpha = v^T y$
4. Compute $\beta = v^T z$
5. Compute $x = z + \dfrac{\beta}{1-\alpha} y$

$\Big\}\ O(n)$

So $O(n^2) + O(n) = O(n^2)$

Hence the fast algorithm solves $(A - uv^T)x = b$ in $O(n^2)$, since A is already factored.

Therefore from Sherman-Morrison formula

$$x = z + \frac{\beta}{1-\alpha} y$$

$$x = \left(A^{-1} + \left(\frac{A^{-1}uv^TA^{-1}}{1 - v^TA^{-1}u}\right)\right)b$$

$$\underline{\underline{x = (A - uv^T)^{-1}b}}$$

b) Identify the vectos u and v in ~~from~~ this system.

$$u = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad v = \begin{bmatrix} 2 \\ 2 \\ 2 \\ \vdots \\ 2 \end{bmatrix}$$