INDIAN INSTITUTE OF TECHNOLOGY GANDHINAGAR

MA 202: MATHEMATICS - IV
Semester–II, Academic Year 2022-23

Tutorial Set -3
Question - 2

By

Kush Patel
[20110131]

- In question 2, we are asked to prove that the number of steps required to complete Gauss elimination (forward sub- stitution and backward elimination) is $O(n^3)$, for an $n \times n$ system of equations. We also need to show that the number of steps for Gauss-Seidel iterations is $O(n^2)$ for that same system of equations.

- For the Gaussian elimination, in the first pass, the number of iterations in the middle loop will be (n-1). It can be observed from the code that for every one of these iterations, there would be total (n+1) multiplications.

- Therefore, the total number of multiplications after traversing these two loops would be $(n^2 - 1)$. Similarly, the number of addition/subtraction operations would be (n-1)(n). Now, while traversing the outer loop, we would need a summation of (n-j)(n-j+1), where k traverses from 1 to n-1. This is the algorithm that goes on in the case of forward substitution.

```
function Xr =T9_20110131(A,B)
% Matrix A(n*n)
% Matrix B(n*1)                  % System Ax=B
P = [A B];                       % Constructing the new augmented matrix P
p = size(P);                     % Calculating the size of augmented matrix, P
% Check whether all diagonal elements of Matrix P or Matrix A are non zero
for m = 1:p(1)
    if P(m,m) == 0
       disp('Gauss elimination method can not applicable. Rearrange the equations!!!');
             %Diagonal element zero. Hence pivote can't be calculated.
       return
    end
end
% Run a loop to perform all steps of Gauss Elimination
% Finding zeros of lower triangular matrix.
for m = 1:p(1)-1
    a=P(m,m);
     P(m,:)= P(m,:)/a;                % Devide all elements by its diagonal element
     % run a loop to perform a row opertaion
    for k=m+1:p(1)
      P(k,:)= P(k,:)- P(k,m)* P(m,:);
    end
end
% Perform a operation on last row
 a=P(p(1),p(1));
 P(p(1),:)= P(p(1),:)/a;
 s=0;
for m=p(1):-1:2        %Finding the final solutions
    for k=m+1:p(2)
        s=s+P(m-1,k-1)* P(k-1,p(2));
```

Gauss Elimination Method

➔ Next, when we perform back-substitution, in the first pass of the first loop, the number of iterations of the interior loop would be (n-j). Next, the outer loop runs from n-1 to 1, which means that in each of these (n-1) iterations of the outer loop, the inner loop performs (n-j) additions and multiplications. Therefore, the total number of operations would be (n-j)(n-1), which is of order n^2.

➔ Summing these results, we find the order of Gaussian elimination as $2n^3/3 + O(n^2)$.

```
% Gauss Seidel Iteration method
% Call the function by giving two matrix as a input
function Xr = GaussSeidel(A,B)
% Matrix A(n*n)
% Matrix B(n*1)              % System Ax=B
n = size(B);                 % Calculating the size of matrix B
p = n(1);                    % Define a number of rows in matrix B
tol = 0.0001;                % define a value of tolerance
Err = ones(p,1);             % Define a error column matrix
X  = zeros(p,1);             % Define a initial guess of solution
C  = zeros(p,1);             % Define a dummy column matrix for calculations
% Run a loop to find  final result by applying the condition of Gauss
% Seidel ietration method
while max(Err) > tol
    for i = 1:p
        C(i,1) = X(i,1);
        X(i,1) = (1/A(i,i)) * (B(i,1) - sum(A(i,:) * X(:,1)) + A(i,i)*X(i,1));
        Err(i,1) = abs(C(i,1) - X(i,1));
        C(i,1) = X(i,1);
    end
end
disp('Solution by Gauss Seidel Iteration Method is:')
Xr = X;
end
```

Gauss-Seidel Method

➔ In the first pass of the first loop, the inner loop traverses from 1 to i-1, that is, it performs these operations for (i-1) times. While the second inner loop similarly performs these operations for (n-i) times. Therefore, the total number of times these operations are performed in the inner loop is (n-1). Now the outer loop runs from 1 to n.

➔ This means the total number of operations performed after traversing these two loops is (n-1)(n). Now, the while loop runs until the error becomes less than the tolerance. This might be a constant value.

➔ Summing these results, we find the order of the Gaussian-Seidel Method as $O(n^2)$.