Solution of linear Systems by iteration methods

Mr. Edem. K. Bankas

Iteration Methods

- In certain cases, such as when a system of equations is large, iterative methods of solving equations are more advantageous.
- Elimination methods, such as Gaussian elimination, are prone to large round-off errors for a large set of equations.

Iteration Methods

- For Iteration methods, we start with an approximation to the true solution and implement it on a set of computational cycle, if successful, the computational cycle is repeated again and again to provide better and better approximation to the true value.

Gauss–Seidel Method

- Solve the linear system below by Gauss-Seidel method using three iterations.

$$8x_1 + 2x_2 + 3x_3 = 30$$

$$x_1 - 9x_2 + 2x_3 = 1$$

$$2x_1 + 3x_2 + 6x_3 = 31$$

Solution

- First Iteration

Putting $x_2 = 0$ and $x_3 = 0$ into equ (1):

$$8x_1 = 30$$

$$x_1 = \frac{30}{8}$$

Putting $x_1 = \frac{30}{8}$ and $x_3 = 0$ into equ (2):

$$\frac{30}{8} - 9x_2 = 1$$

$$x_2 = \frac{11}{36}$$

Putting $x_1 = \frac{30}{8}$ and $x_2 = \frac{11}{36}$ into equ (3)

$$x_3 = \frac{271}{72} = 3.76$$

Solution

- Second Iteration

Please continue …. up to the 4^{th} iteration

n	0	1	2	3	4
x_1	0.0000	3.7500	2.2600	2.0400	
x_2	0.0000	0.3056	0.9800	0.9900	
x_3	0.0000	3.7600	3.9200	3.9900	

Example 2

- Solve the linear system using Gauss seidel method with three iterations

$$4x + 3y - z = 2$$

$$9x + 13y - 2z = 20$$

$$11x + y - 3z = 41$$

Jacobi Method

This method uses two assumptions:

1. The system

$$
\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\
&\vdots \\
a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n &= b_n
\end{aligned}
$$

Has a unique solution

Jacobi Method

- The coefficient Matrix A has no zero on its main diagonal. If any diagonal entries are zero, then row or column must be interchanged to obtain a coefficient matrix that has no zero on the main diagonal

Jacobi Method

- To begin the Jacobi method, solve the first equation for x_1, the second equation for x_2 and so on.

$$x_1^1 = \frac{1}{a_{11}}(b_1 - a_{12}x_2^0 - \cdots - a_{1n}x_n^0)$$

$$x_2^1 = \frac{1}{a_{22}}(b_2 - a_{21}x_1^0 - a_{23}x_3^0 - \cdots - a_{2n}x_n^0)$$

$$x_n^1 = \frac{1}{a_{nn}}(b_n - a_{n1}x_1^0 - a_{n2}x_2^0 - \cdots - a_{nn-1}x_{n-1}^0)$$

Example

- Use the Jacobi method to approximate the solution of the linear system

$$5x_1 - 2x_2 + 3x_3 = -1$$
$$-3x_1 + 9x_2 + x_3 = 2$$
$$2x_1 - x_2 - 7x_3 = 3$$

Continue with the iteration until two successive approximations are identical when rounded to 3 sig. fig.

```
% A program to compute the loan payment from a Bank
% The LOAN AMOUNT to take
LA = input('Enter the Loan Amount ');
% The number of years the loan is payerble
NY = input('Length of year ');
% Interest rate
APR = input('Interest rate ');
% Compute the interest rate per month
IPM = APR/(12*100)
% Compute the number of months
NM = NY*12
%Compute and display the Amount payerble per month
PMT = (LA*IPM)/(1-(1+IPM)^(-NM))
fprintf('The Amount Payerble every month is %f', PMT)
```

-