Maddox Gonzalez

Contents

Due 2/23/2025

Project 4 - Jacobi and Gauss-Seidel

```
% Project 4
clc, clear, close all;
format long;

A = dlmread('fin_data_A_matrix_SP_2025.txt');
b = dlmread('fin_data_b_vector_SP_2025.txt');

% A = 4*4 matrix;
% b = 4*1 vector;

% Declare initial guess vector
guess = zeros(length(b), 1);
guess(1:3:end) = 100;
guess(2:3:end) = 200;
guess(3:3:end) = 300;

% Residual and iterative tolerance
tol = 1e-4;

% SOR parameter
omega = 1;

% Run Gauss-Seidel with SOR
[x, res, conv, num_iters] = GS_SOR(A, b, guess, tol, omega);

% Print only the required outputs
fprintf('Number of iterations to converge = %i\n', num_iters);
indices = [26, 88, 151, 200];
for i = 1:length(indices)
    idx = indices(i);
    fprintf('Element %d: %.3f\n', idx, x(idx));
end
fprintf('res = %.8e\n', res(end));
fprintf('conv = %.8e\n', conv(end));
fprintf("==================================\n");

% Run GS SOR for different omegas and save number of iterations
% to convergence for each omega
omegas = 1.0:0.1:1.9; % Start:Step:Stop
iters_to_conv = zeros(size(omegas));

for i = 1:length(omegas)
    [x, res, conv, num_iters] = GS_SOR(A, b, guess, tol, omegas(i));
```

```
    iters_to_conv(i) = num_iters;
    if i == 1
        fprintf('Omega = %.2f took %i iterations\n', omegas(i), num_iters);
    end
end

for i = 2:length(omegas)
    fprintf('Omega = %.2f took %i iterations\n', omegas(i), iters_to_conv(i));
end
```

Plots

```
function [x, res, conv, n] = GS_SOR(A, b, guess, tol, omega)
    max_iter = 1e5;
    x = guess;
    n = 1;

    while n < max_iter
        x_old = x;
        for i = 1:length(b)
            x(i) = (b(i) - A(i, :) * x(:) + A(i, i) * x(i)) / A(i, i);
            x(i) = omega * (x(i)) + (1 - omega) * x_old(i);
        end

        res(n) = norm(A * x - b, inf);
        conv(n) = norm(x - x_old, inf);

        if res(n) < tol && conv(n) < tol
            break;
        end

        if res(n) > 1e15
            fprintf('Diverged');
            break;
        end

        n = n + 1;
    end
end
```

```
Number of iterations to converge = 24698
Element 26: 135.224
Element 88: 84.249
Element 151: 41.819
Element 200: 27.482
res = 9.99771488e-05
conv = 2.03046537e-06
==================================
Omega = 1.00 took 24698 iterations
Omega = 1.10 took 20211 iterations
Omega = 1.20 took 16473 iterations
Omega = 1.30 took 13309 iterations
Omega = 1.40 took 10597 iterations
Omega = 1.50 took 8246 iterations
Omega = 1.60 took 6188 iterations
```

```
Omega = 1.70 took 4372 iterations
Omega = 1.80 took 2755 iterations
Omega = 1.90 took 1298 iterations
```

Published with MATLAB® R2024b