
```

clc; clear;

% Define matrix A and vector b, Ax=b
A = [1 1 1; 0 0 1; 1 0 1; 2 0 5; -7 8 0; 1 2 -1];
a = [3; 1; 2; 8; 0; 1];

% (a) Find the least squares solution
% Apply the formula
x_ls = (A' * A) \ (A' * a);

disp('The Least Squares solution:')
disp(x_ls)

% (b) Implement the Conjugate Gradient Method to get the same answer
% Define matrix Q and vector c
Q = A' * A;
b = A' * a;

% Set tolerance and maximum number of iterations
tol = 1e-4;
max_iter = 1000;

% Initialize x and calculate g0 and d0
x = zeros(size(Q,1),1);
g = Q * x - b;
d = -g;

for i = 1:max_iter
    alpha = -(g' * d) / (d' * Q * d);
    x = x + alpha * d;
    g = Q * x - b;

    % Check the convergence
    if norm(g) < tol
        break;
    end

    beta = (g' * Q * d) / (d' * Q * d);
    d = -g + beta * d;
end

disp('The Conjugate Gradient Method solution:')
disp(x)

The Least Squares solution:
    0.8503
    0.7438
    1.2548

The Conjugate Gradient Method solution:
    0.8503
    0.7438

```

1.2548

Published with MATLAB® R2023b