HIMANSHU CHOUDHARY (200020059)

CL249 : ASSIGNMENT 5

Problem

We are given a data set of X's and Y's and we have to fit an eight degree polynomial ~~on~~ and find difference between sqrt(x) and calculated polynomial on given values, and plot the values and the function.

Description of Method

We are given discrete x and y eg. x_0, y_0, x_1, y_1, etc. and we know that by the polynomial.

$$y_0 = a_0 + a_1 x_0 + a_2 x_0^2 + \cdots + a_n x_0^n \qquad \text{here } n=8$$

$$y_1 = a_0 + a_1 x_1 + a_2 x_1^2 + \cdots + a_n x_1^n$$

$$\vdots$$

$$y_n = a_0 + a_1 x_n^1 + \cdots + a_n x_n^n$$

So. it can be represented as a matrix eq.

$$\begin{bmatrix} 1 & x_0 & x_0^2 & x_0^3 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & x_1^3 & & \vdots \\ 1 & x_2 & \vdots & & \ddots & \vdots \\ \vdots & \vdots & \vdots & \cdots & \ddots & \\ 1 & x_n & x_n^2 & x_n^3 & & x_n^n \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}$$

$$A \cdot X = B$$

and we can solve this by Gauss Elimination.

and by finding the coeff., we get the polynomial.

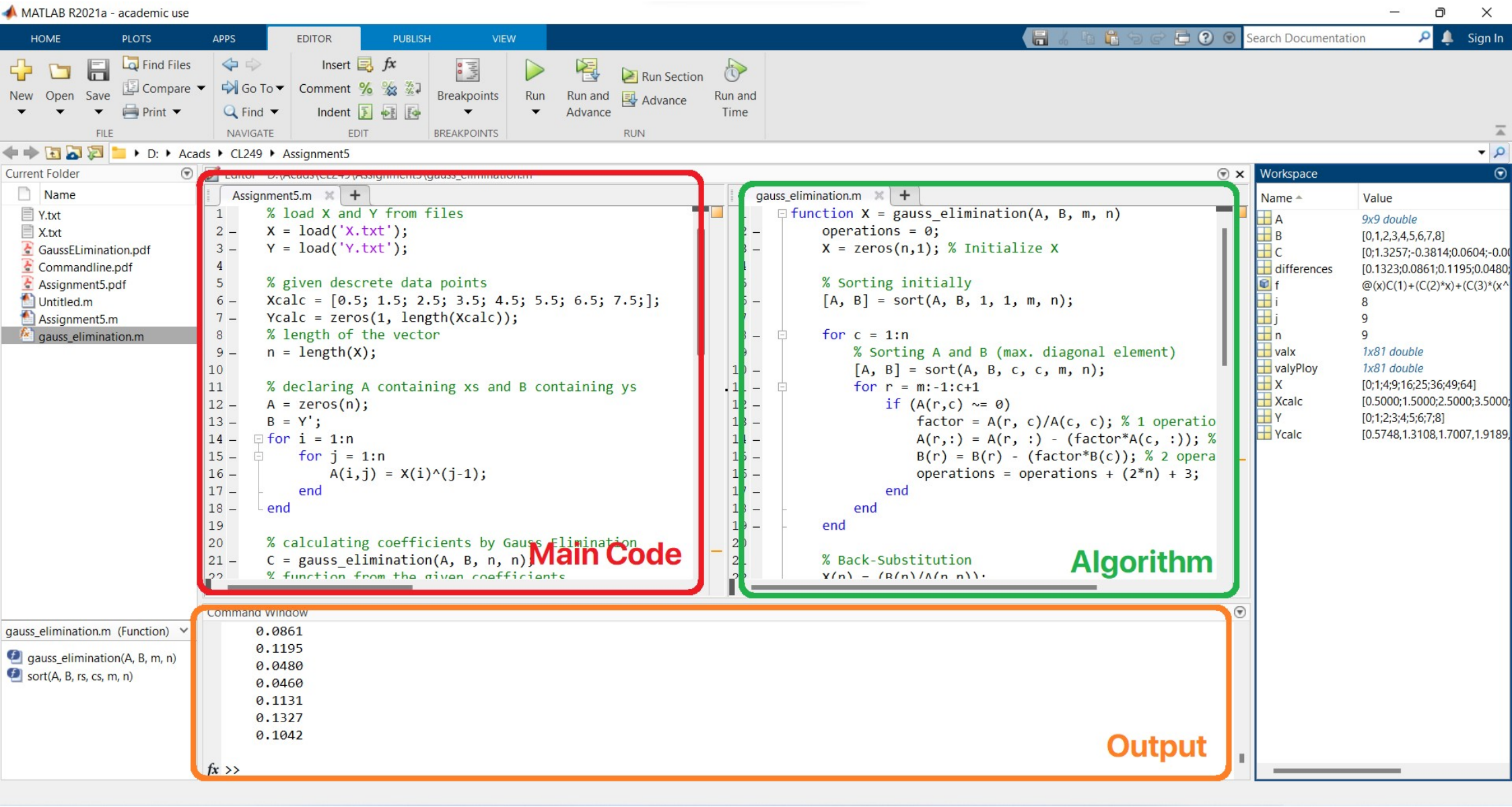
Pseudo Code

main

load X and Y
and descrete values
calculate matrices A and B.
~~Solve AX = B. as~~
get X = $A^{-1}B$
X : gauss_elim(A, B)
define function f(x) from coefficients
decalre difference matrix
loop from 1 : length(diff)
→ difference = |f(x) - sqrt(x)|
get descrete values X
y = f(x)
plot x and y

```
% load X and Y from files
X = load('X.txt');
Y = load('Y.txt');

% given descrete data points
Xcalc = [0.5; 1.5; 2.5; 3.5; 4.5; 5.5; 6.5; 7.5;];
Ycalc = zeros(1, length(Xcalc));
% length of the vector
n = length(X);

% declaring A containing xs and B containing ys
A = zeros(n);
B = Y';
for i = 1:n
    for j = 1:n
        A(i,j) = X(i)^(j-1);
    end
end

% calculating coefficients by Gauss Elimination
C = gauss_elimination(A, B, n, n);
% function from the given coefficients
f = @(x) C(1) + (C(2)*x) + (C(3)*(x^2)) + (C(4)*(x^3)) + (C(5)*(x^4)) + (C(6)*(x^5)) ↙
+ (C(7)*(x^6)) + (C(8)*(x^7)) + (C(9)*(x^8));
differences = zeros(length(Xcalc), 1);

% calculating differences in the given data points
for i = 1:length(Xcalc)
    differences(i) = abs(sqrt(Xcalc(i)) - f(Xcalc(i)));
end

% values for plotting
valx = 0:0.1:8;
valyPloy = zeros(1, length(valx));

for i = 1:length(valx)
    valyPloy(i) = f(valx(i));
end

for i = 1:length(Xcalc')
    Ycalc(i) = f(Xcalc(i));
end

% Plotting values
plot(Xcalc', Ycalc, 'kd', 'LineWidth', 1.25, 'DisplayName', 'Given Data Points');
xlabel('X');
ylabel('f(x)');
hold on
fplot(@(x) sqrt(x), 'b:', 'LineWidth', 1.25)
plot(valx, valyPloy, 'r', 'DisplayName', 'f(x) (Polynomial)', 'LineWidth', 1.25)
legend

disp('Coefficients (a0, a1, .... , a8)')
disp(C)
disp('Differences (0.5, 1.5, .... , 7.5)')
differences
```

```
function X = gauss_elimination(A, B, m, n)
    operations = 0;
    X = zeros(n,1); % Initialize X

    % Sorting initially
    [A, B] = sort(A, B, 1, 1, m, n);

    for c = 1:n
        % Sorting A and B (max. diagonal element)
        [A, B] = sort(A, B, c, c, m, n);
        for r = m:-1:c+1
            if (A(r,c) ~= 0)
                factor = A(r, c)/A(c, c);  % 1 operation
                A(r,:) = A(r, :) - (factor*A(c, :));  % 2*n operations
                B(r) = B(r) - (factor*B(c));  % 2 operations
                operations = operations + (2*n) + 3;
            end
        end
    end

    % Back-Substitution
    X(n) = (B(n)/A(n,n));
    operations = operations + 1;
    for i = m-1:-1:1
        sum = 0;
        for j = n:-1:i+1
            sum = sum + (X(j)*A(i, j));
            operations = operations + 2;
        end
        X(i) = (B(i) - sum)/A(i, i);
        operations = operations + 2;
    end

end

% Sorting Function
function [mat1, mat2] = sort(A, B, rs, cs, m, n)
    for s = rs:m-1
        for r = rs:m-1
            if abs(A(r, cs)) < abs(A(r+1, cs))
                temp1 = A(r, :);
                A(r, :) = A(r+1, :);
                A(r+1, :) = temp1;
                temp2 = B(r);
                B(r) = B(r+1);
                B(r+1) = temp2;
            end
        end
    end
    mat1 = A;
    mat2 = B;
end
```


MATLAB R2021a - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Search Documentation
Sign In
New
Open
Save
Find Files
Compare
Print
FILE
Go To
Find
NAVIGATE
Insert
Comment
Indent
EDIT
Breakpoints
BREAKPOINTS
Run
Run and Advance
Run Section
Advance
Run and Time
RUN
D: ▸ Acads ▸ CL249 ▸ Assignment5
Current Folder
Name
Y.txt
X.txt
GaussELimination.pdf
Commandline.pdf
Assignment5.pdf
Untitled.m
Assignment5.m
gauss_elimination.m
Assignment5.m
% load X and Y from files
X = load('X.txt');
Y = load('Y.txt');
% given descrete data points
Xcalc = [0.5; 1.5; 2.5; 3.5; 4.5; 5.5; 6.5; 7.5;];
Ycalc = zeros(1, length(Xcalc));
% length of the vector
n = length(X);
% declaring A containing xs and B containing ys
A = zeros(n);
B = Y';
for i = 1:n
for j = 1:n
A(i,j) = X(i)^(j-1);
end
end
% calculating coefficients by Gauss Elimination
C = gauss_elimination(A, B, n, n);
Main Code
gauss_elimination.m
function X = gauss_elimination(A, B, m, n)
operations = 0;
X = zeros(n,1); % Initialize X
% Sorting initially
[A, B] = sort(A, B, 1, 1, m, n);
for c = 1:n
% Sorting A and B (max. diagonal element)
[A, B] = sort(A, B, c, c, m, n);
for r = m:-1:c+1
if (A(r,c) ~= 0)
factor = A(r, c)/A(c, c); % 1 operatio
A(r,:) = A(r, :) - (factor*A(c, :)); %
B(r) = B(r) - (factor*B(c)); % 2 opera
operations = operations + (2*n) + 3;
end
end
end
% Back-Substitution
Algorithm
Workspace
Name
Value
A
9x9 double
B
[0,1,2,3,4,5,6,7,8]
C
[0;1.3257;-0.3814;0.0604;-0.0
differences
[0.1323;0.0861;0.1195;0.0480;
f
@(x)C(1)+(C(2)*x)+(C(3)*(x^
i
8
j
9
n
9
valx
1x81 double
valyPloy
1x81 double
X
[0;1;4;9;16;25;36;49;64]
Xcalc
[0.5000;1.5000;2.5000;3.5000;
Y
[0;1;2;3;4;5;6;7;8]
Ycalc
[0.5748,1.3108,1.7007,1.9189,
Command Window
0.0861
0.1195
0.0480
0.0460
0.1131
0.1327
0.1042
fx >>
Output
gauss_elimination.m (Function)
gauss_elimination(A, B, m, n)
sort(A, B, rs, cs, m, n)

```
>> Assignment5
Coefficients (a0, a1, .... , a8)
         0
    1.3257
   -0.3814
    0.0604
   -0.0050
    0.0002
   -0.0000
    0.0000
   -0.0000

Differences (0.5, 1.5, .... , 7.5)

differences =

    0.1323
    0.0861
    0.1195
    0.0480
    0.0460
    0.1131
    0.1327
    0.1042

>>
```

Figure 1

File Edit View Insert Tools Desktop Window Help