EGR 103L – Fall 2017

Laboratory 8 - Linear Algebra

Ian Hanus (ih52)
Lab Section 1B, Tuesday 8:30-11:20
5 November 2017

I understand and have adhered to all the tenets of the Duke Community Standard in completing every part of this assignment. I understand that a violation of any part of the Standard on any part of this assignment can result in failure of this assignment, failure of this course, and/or suspension from Duke University.

Contents

List of Figures

1 Palm Problem 8.1

Part	x	y	z
a	2.4762e+00	4.7619e-02	N/A
b	-1.1818e+00	1.0909e+00	N/A
c	3.4329e+00	-6.0390e+00	4.4935e+00
d	2.0035e+00	-2.6848e+00	5.2312e+00

2 Based on Chapra Problem 8.3

Equations in matrix form:

$$\begin{bmatrix} 0 & -7 & 5 \\ 0 & 4 & 7 \\ -4 & 3 & -7 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} 50 \\ -30 \\ 40 \end{Bmatrix}$$

Solutions for this system:

$$\begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} -1.5181e+01 \\ -7.2464e+00 \\ -1.4493e-01 \end{Bmatrix}$$

Transpose of A:

$$\begin{bmatrix} 0 & 0 & -4 \\ -7 & 4 & 3 \\ 5 & 7 & -7 \end{bmatrix}$$

Inverse of A:

$$\begin{bmatrix} -1.7754e-01 & -1.2319e-01 & -2.5000e-01 \\ -1.0145e-01 & 7.2464e-02 & 0 \\ 5.7971e-02 & 1.0145e-01 & 0 \end{bmatrix}$$

Condition numbers:

$$\text{1-norm condition} = 6.4022e+00$$
$$\text{2-norm condition} = 4.0569e+00$$
$$\text{Frobenius condition} = 5.4293e+00$$
$$\infty\text{-condition} = 7.7101e+00$$

The condition numbers signify how inaccurate the solution can be. Because the condition for each fo the norms is greater than one, the system is ill-conditioned. This suggests that the solutions of the system could exhibit rounding errors. Because the magnitude of each of the condition numbers is below ten, the last digit of the solutions could exhibit roundoff errors as the logarithm of the condition numbers is between 0 and 1.

3 Based on Chapra Problem 8.10

$$
\begin{bmatrix}
-8.6603e-01 & 0 & 5.0000e-01 & 0 & 0 & 0 \\
-5.0000e-01 & 0 & -8.6603e-01 & 0 & 0 & 0 \\
8.6603e-01 & 1.0000e+00 & 0 & 1.0000e+00 & 0 & 0 \\
5.0000e-01 & 0 & 0 & 0 & 1.0000e+00 & 0 \\
0 & -1.0000e+00 & -5.0000e-01 & 0 & 0 & 0 \\
0 & 0 & 8.6603e-01 & 0 & 0 & 1.0000e+00
\end{bmatrix}
\begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ H_2 \\ V_2 \\ V_3 \end{Bmatrix}
=
\begin{Bmatrix} 0 \\ 1000 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix}
$$

```
F1: -5.000e+02
F2: +4.330e+02
F3: -8.660e+02
H2: +0.000e+00
V2: +2.500e+02
V3: +7.500e+02
```

4 Palm 8.5(b)

There is a c value with easily predicted solutions. For a c value of 0, x,y, and z are expected to be zero. This seems appears to be true on the graph. There are 201 points instead of 200 because using 200 creates an even spacing of 0.1 between all points. This would include the c value zero. Using 201 points, the spacing is different and does not include a calculated c value of zero.

5 Based on Palm 8.9

$$
\begin{bmatrix}
-1 & 1/3 & 1/3 & 0 \\
1/2 & -1 & 0 & 1/2 \\
1/3 & 0 & -1 & 1/3 \\
0 & 1/2 & 1/2 & -1
\end{bmatrix}
\begin{Bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{Bmatrix}
=
\begin{Bmatrix} -50 \\ 0 \\ -20/3 \\ 0 \end{Bmatrix}
$$

The following are the temperatures at each square.

```
T1: +1.027e+02 C
T2: +9.091e+01 C
T3: +6.727e+01 C
T4: +7.909e+01 C
```

6 Based on Palm 8.16(a)

$$
\begin{bmatrix}
x_1^2 & x_1 & 1 \\
x_2^2 & x_2 & 1 \\
x_3^2 & x_3 & 1
\end{bmatrix}
\begin{Bmatrix} a \\ b \\ c \end{Bmatrix}
=
\begin{Bmatrix} y_1 \\ y_2 \\ y_3 \end{Bmatrix}
$$

The following are the coefficients of the quadratics containing the given points.

Points	a	b	c
(1,4), (4,73), (5,120)	6.00e+00	-7.00e+00	5.00e+00
(1,4), (4,-73), (5,120)	5.47e+01	-2.99e+02	2.48e+02
(1,4), (4,73), (4,120)	N/A	N/A	N/A
(1,4), (4,73), (5,-120)	-5.40e+01	2.93e+02	-2.35e+02

A Codes and Output

A.1 Chapra83.m

```
 % I have adhered to all the tenets of the
 % Duke Community Standard in creating this code.
 % Signed: [ih52]
%% Initialize workspace
clear; format short e

%% Setup matrices
A = [0 -7 5 ;0 4 7; -4 3 -7]
b = [50; -30; 40]

%% Solutions for the system
Solutions = A\b

%% Transpose and inverse of A
ATrans = A'
AInv = inv(A)

%% Calculate the condition numbers
Norm1Cond = cond(A,1)
Norm2Cond = cond(A,2)
NormFroCond = cond(A,'fro')
NormInfCond = cond(A,inf)
```

A.2 Chapra810.m

```
 % I have adhered to all the tenets of the
 % Duke Community Standard in creating this code.
 % Signed: [ih52]
%% Initialize workspace
clear; format short e

%% Rewrite equations in matrix form
A = [-cosd(30) 0 cosd(60) 0 0 0
    -sind(30) 0 -sind(60) 0 0 0
    cosd(30) 1 0 1 0 0
    sind(30) 0 0 0 1 0
    0 -1 -cosd(60) 0 0 0
    0 0 sind(60) 0 0 1]
b = [0; 1000; 0; 0; 0; 0]

%% Solve for unknowns
Solutions = A\b;

%% Save in text file
FID = fopen('TrussData.txt', 'w');

fprintf(FID, 'F1: %+1.3e \n', Solutions(1))
fprintf(FID, 'F2: %+1.3e \n', Solutions(2))
fprintf(FID, 'F3: %+1.3e \n', Solutions(3))
fprintf(FID, 'H2: %+1.3e \n', Solutions(4))
fprintf(FID, 'V2: %+1.3e \n', Solutions(5))
fprintf(FID, 'V3: %+1.3e \n', Solutions(6))

fclose(FID)
```

A.3 Palm85

```
 % I have adhered to all the tenets of the
 % Duke Community Standard in creating this code.
 % Signed: [ih52]
%% Initialize workspace
clear; format short e

%% Create coefficient matrix A
A = [1 -5 -2; 6 3 1; 7 3 -5];

%% Create c values for plotting
c = linspace(-10,10,201);

%% Loop through different c values
for k = 1:length(c)
    b = [11*c(k); 13*c(k); 10*c(k)];
    MyVals = A\b
    x(k) = MyVals(1);
    y(k) = MyVals(2);
    z(k) = MyVals(3);
end

%% Make and save plot
plot(x,c,'k-',y,c,'c-.',z,c,'m--')
legend('x-values','y-values','z-values')
title('Plot of Palm 8.5(b)')
xlabel('c value')
ylabel('Solutions value')
print -depsc Palm85bPlot
```

A.4 Palm89

```
 % I have adhered to all the tenets of the
 % Duke Community Standard in creating this code.
 % Signed: [ih52]
%% Initialize workspace
clear; format short e

%% Create matrices
A = [-1 1/3 1/3 0; 1/2 -1 0 1/2;1/3 0 -1 1/3;0 1/2 1/2 -1]
b = [-1/3*150;0;-1/3*20;0]

%% Find temperatures
Temps = A\b

%% Print and save results
FID = fopen('TempData.txt','w')

fprintf(FID,'T1: %+1.3e C\n', Temps(1))
fprintf(FID,'T2: %+1.3e C\n', Temps(2))
fprintf(FID,'T3: %+1.3e C\n', Temps(3))
fprintf(FID,'T4: %+1.3e C\n', Temps(4))

fclose(FID)
```

A.5 Palm816

```
 % I have adhered to all the tenets of the
 % Duke Community Standard in creating this code.
 % Signed: [ih52]
%% Create function
function [a, b, c] = findquad(x,y,flag)

%% Check user inputs
if nargin < 2
   error('Not enough arguments.')
end
if nargin == 2
    flag = 0;
end

%% Create matrices
Amat = [x(1)^2, x(1), 1;x(2)^2, x(2), 1;x(3)^2, x(3), 1];
bmat = [y(1); y(2); y(3)];

%% Check condition number
if cond(Amat,2) > 10^5
    error('Matrix is ill-conditioned')
end

%% Solve for a, b, & c
Solutions = Amat\bmat;
a = Solutions(1)
b = Solutions(2)
c = Solutions(3)

%% Plot of polynomial
if flag == 1
    MyPoly = @(x) a*x.^2+b*x+c;
    Xmin = min(x) - 0.1*(max(x)-min(x));
    Xmax = max(x) + 0.1*(max(x)-min(x));
    Xvals = linspace(Xmin,Xmax,100);
    plot(Xvals,MyPoly(Xvals),'k-')
    hold on
    for k = 1:3
        plot(x(k),y(k),'bs','LineWidth',3,'MarkerSize',12,'MarkerFaceColor','g')
    end
    hold off
    grid on
    xlabel('x')
    ylabel('y')
    title(sprintf('Graph of y =%0.2ex^2%+0.2ex%+0.2e',a,b,c))
end
print -depsc Palm816fig3
```

B Figures

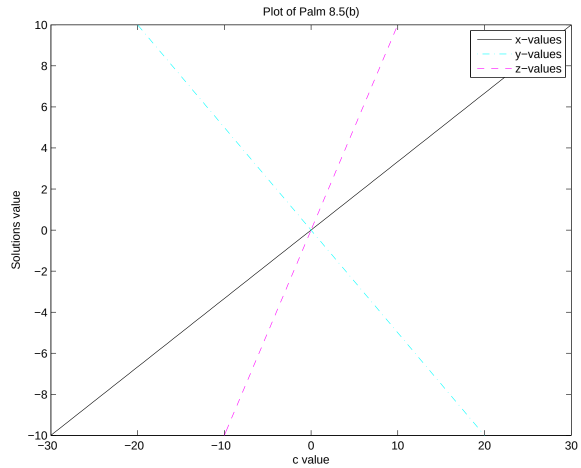

Figure 1: Plot of Palm 8.5(b)

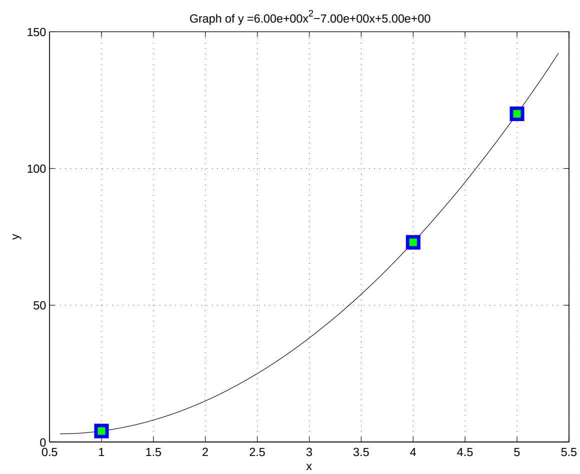

Figure 2: Plot of Polynomial for Points (1,4), (4,73), (5,120)

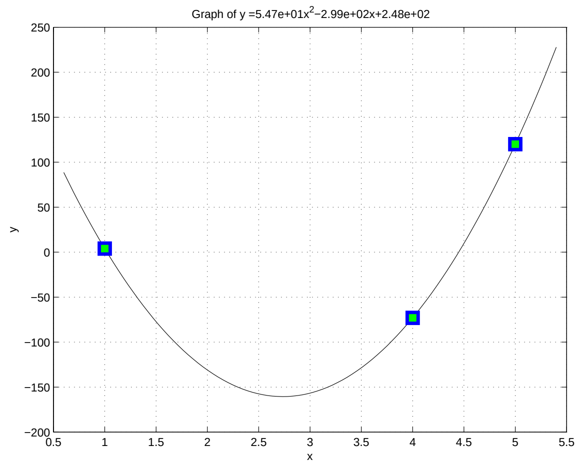

Figure 3: Plot of Polynomial for Points (1,4), (4,-73), (5,120)

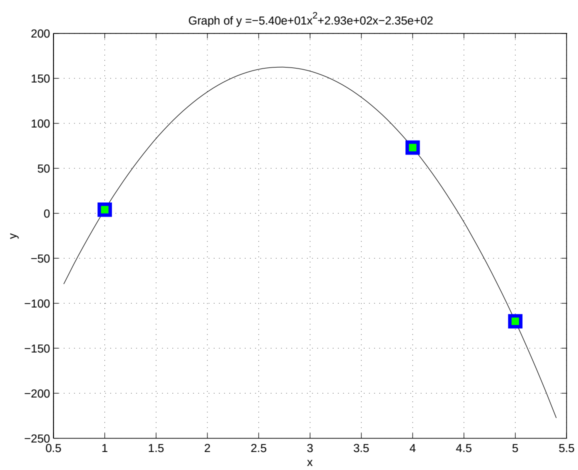

Figure 4: Plot of Polynomial for Points (1,4), (4,73), (5,-120)