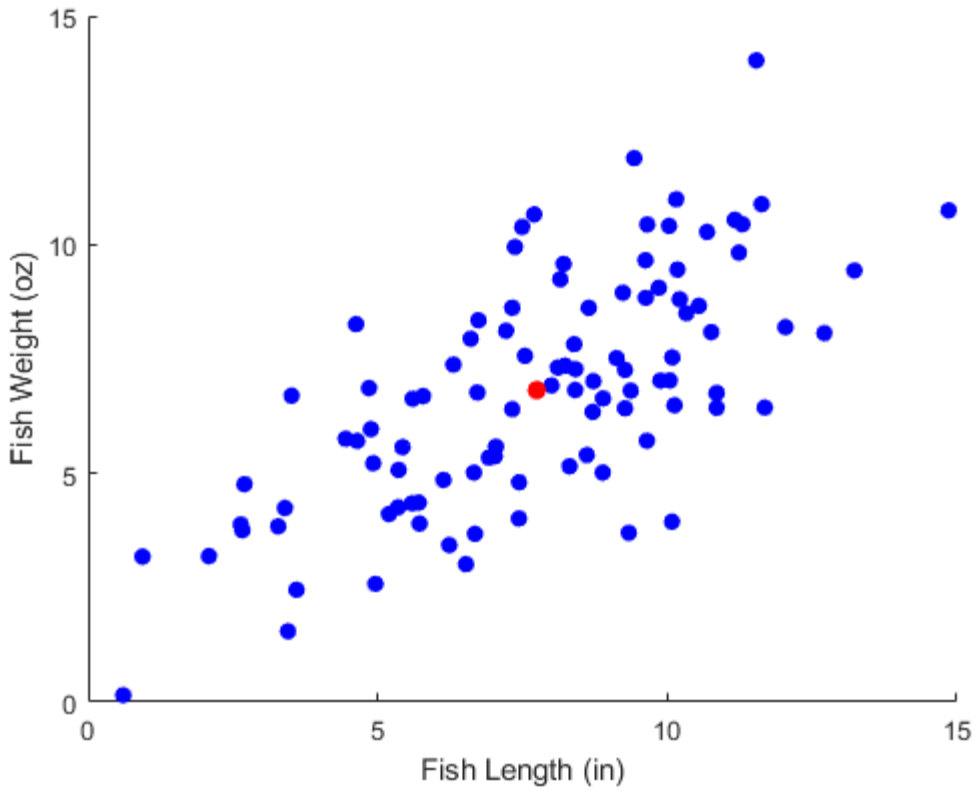
Contents

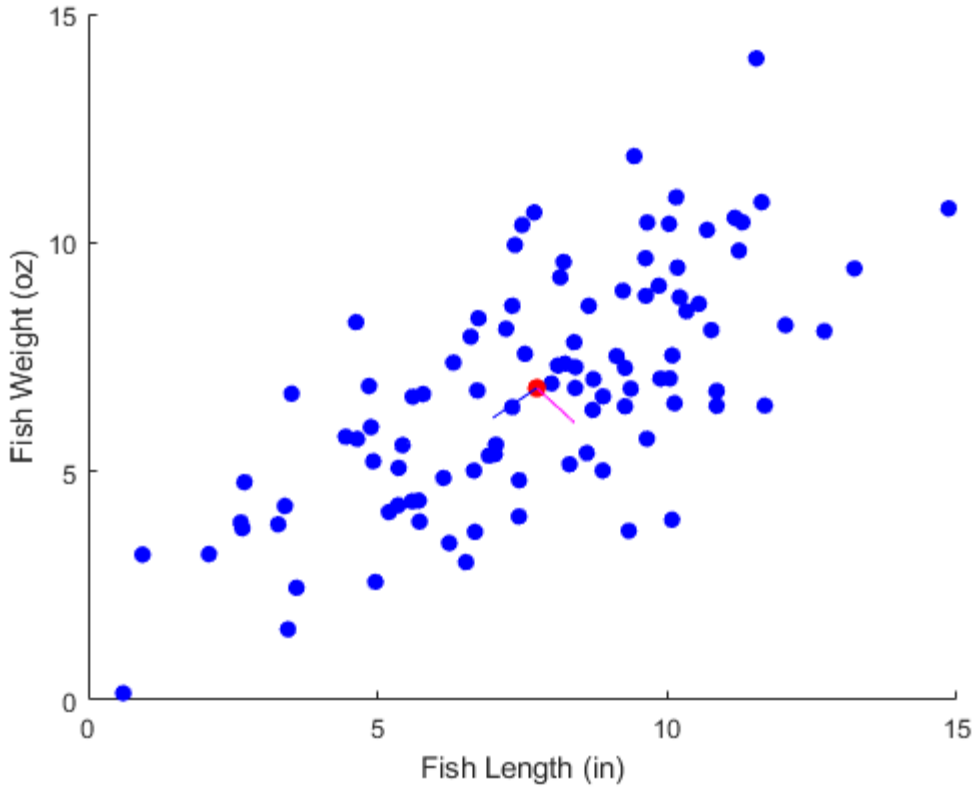

```
close all
clear all
clc

[f_id, f_length, f_weight] = textread('Rock_bass_data(1).txt', '%f%f%f', 'headerlines', 1);

x1m = mean(f_length);
x2m = mean(f_weight);

f0 = figure('Name', 'Preliminary Stats');
mksz = 40; %increased MarkerSize to 40, 20 seemed a little too small

h0 = scatter(f_length, f_weight, mksz, 'blue', 'filled');

hold on

h1 = scatter(x1m, x2m, mksz + 10, 'red', 'filled'); %increased MarkerSize to 80, 20 was smaller than
surrounding dots

% legend('Data', 'Centroid')
xlabel('Fish Length (in)')
ylabel('Fish Weight (oz)')

hold on

xm_tot = [x1m x2m];
xm_m = repmat(xm_tot, length(f_id), 1);

a1 = [f_length f_weight];


%deviation matrix
D = a1 - xm_m;

%sample covariance
%s_n = (1/(n-1))D'D
%cov(f_length, f_weight)

S = 1/(100-1) * D' * D;

%Pearson correlation coefficient
%R = G (D'D) G
%G = [ 1/||d1|| 0; 0 1/||d2||
%R = corrcoef(f_length, f_weight)
D1s = D(:,1).*D(:,1);
D1p = sqrt(sum(D1s));
D2s = D(:,2).*D(:,2);
D2p = sqrt(sum(D2s));

G = [1/D1p 0; 0 1/D2p];
```

```
R = G * (D' * D) * G;

%Echo x1m, x2m, D, S, R
```



Part B

```
%eig
[V, LAMBDA] = eig(S);

ev1 = line([x1m V(1,1)+x1m   ], [x2m V(2,1)+x2m  ]);
ev1.Color = 'm';
ev2 = line([x1m V(1,2)+x1m   ], [x2m V(2,2)+x2m  ]);
ev2.Color = 'b';

hold on

%inverse covariance matrix
S_inv = inv(S);

%Echo V, LAMBDA, S_inv
```

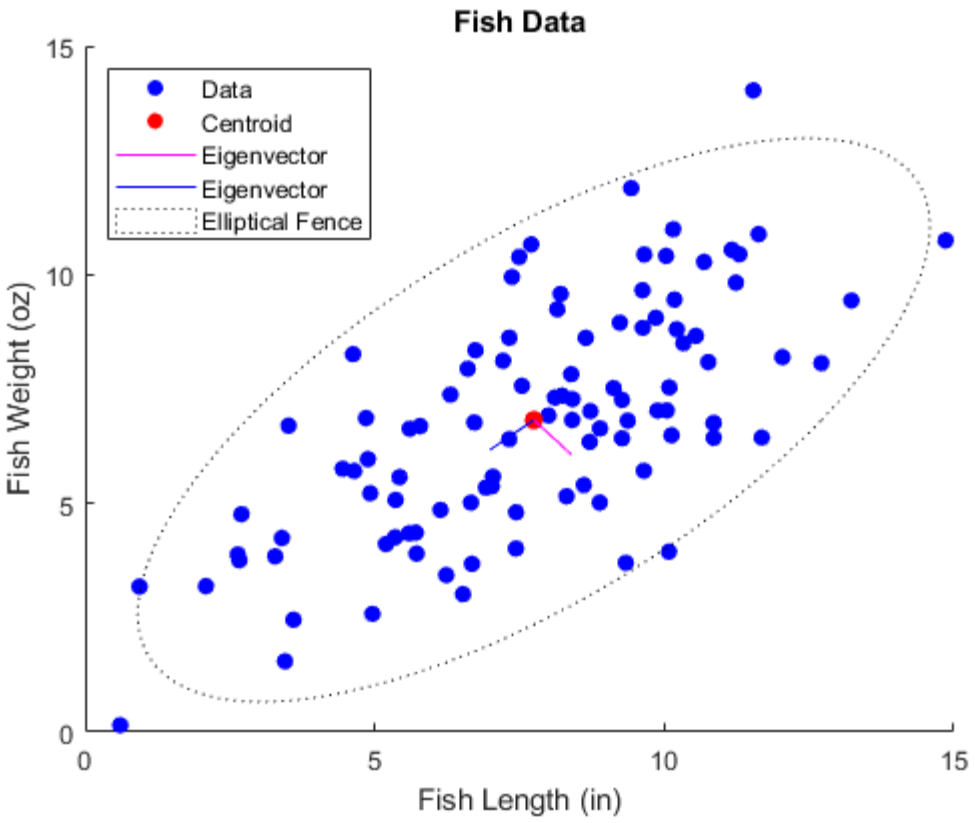


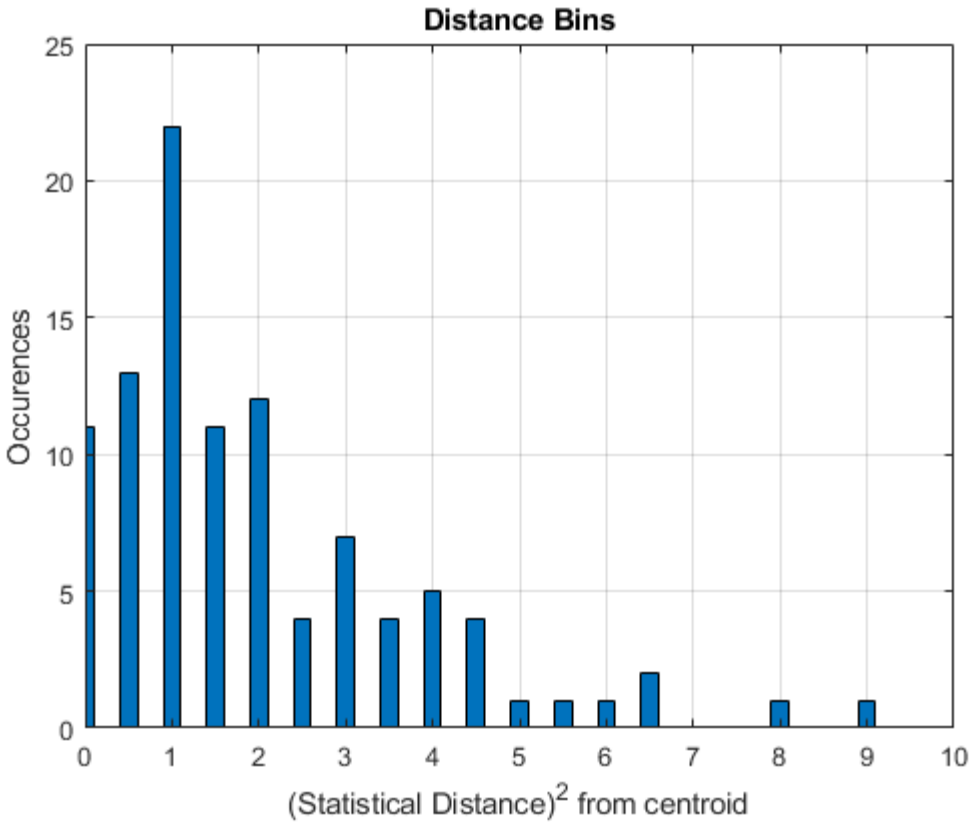
Part C

```
%statistical distance squared

my_distances = 0.2429 .* D(:,1).^2 - 0.3708 .* D(:,1) .* D(:,2) + 0.2993 .* (D(:,2).^2);


theta = 0:2*pi/360:2*pi;
eqn = 0.2429.*(cos(theta)).^2 + 0.2993.*(sin(theta)).^2 - 0.3708.*cos(theta).*sin(theta);
r = sqrt(5.99./eqn);
xc3 = r.*cos(theta) + x1m;
yc3 = r.*sin(theta) + x2m;

el2 = patch([xc3], [yc3], length(theta), 'LineStyle', ':');
el2.FaceAlpha = 0;

lgd = legend('Data', 'Centroid', 'Eigenvector', 'Eigenvector', 'Elliptical Fence' );
lgd.Location = 'northwest';
title('Fish Data')
% hold on
% e12.LineStyle = ':';
% e12.EdgeColor = 'none';

%0.95
disp('I have picked 0.05 for the alpha value');
disp('From the chi-sqare table, the critical c_squared is = 5.99');

f1 = figure('Name', 'Historgram of Statistical Distance');

[occurrences, bin_centers] = hist(my_distances, [0:0.5:10]);
h2 = bar(bin_centers, occurrences,0.4);
axis([0 10 0 25]);
grid on;
```

```
ylabel('Occurences');
xlabel('(Statistical Distance)^2 from centroid');
title('Distance Bins');
```

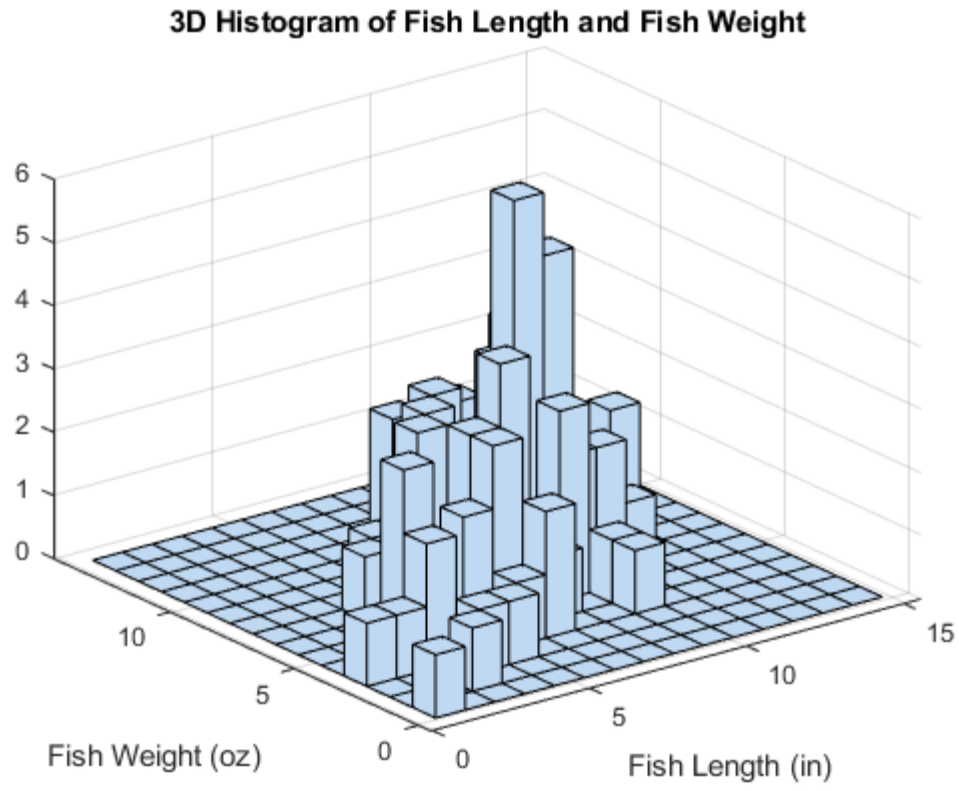
```
I have picked 0.05 for the alpha value
From the chi-sqare table, the critical c_squared is = 5.99
```





Part D

```
out_ind = find(my_distances > 5.99);

out_fid = f_id(out_ind);
f2 = figure('Name', 'Hist3D');
nbinx = 15;
nbiny = 15;
hist3([f_length f_weight], 'Nbins', [nbinx,nbiny])
xlabel('Fish Length (in)')
ylabel('Fish Weight (oz)')
title('3D Histogram of Fish Length and Fish Weight')
```



Echo

```
%Echo x1m, x2m, D, S, R

diary vj_prob1.txt
echo on

%x1 mean
x1m

%x2 mean
x2m

%Deviation Matrix D
D

%Sample Covariance
S

%Pearson Correlation Coefficient Matrix R
R

%V
V

%LAMBDA
LAMBDA

%Covariance inverse S^-1
S_inv

%Distance Values
my distances
```

```
disp('I have picked 0.05 for the alpha value');
disp('From the chi-sqare table, the critical c_squared is = 5.99');

%outlier f_id
out_fid

echo off
```

```
%x1 mean
x1m

x1m =

    7.7474


%x2 mean
x2m

x2m =

    6.8205


%Deviation Matrix D
D

D =

   -1.2246   -3.8057
   -0.7062   -1.2368
    3.9361   -0.3826
   -6.8071   -3.6400
   -1.6121   -1.9653
    7.1107    3.9273
   -2.0260   -2.9182
   -2.0389   -2.4605
   -2.8657   -0.8555
    3.4181    3.7221
   -7.1405   -6.6723
   -0.0461    3.8395
    0.8625   -1.4211
    0.8965    1.7973
    2.9400    3.4585
    1.9076    3.6200
    1.5225   -0.3990
   -3.1026   -1.1115
    0.6627    0.0026
   -4.4653   -2.9795
    2.5804    1.6812
   -0.2040    0.7500
    1.6205   -0.0132
   -1.0259   -0.0539
    3.7889    7.2085
    3.0120    1.2674
   -4.2338   -0.1234
   -5.1118   -2.9376
   -5.0852   -3.0642
```

```
 2.1344   0.2061
 1.8808   2.0163
-2.3929  -2.5619
 2.3346  -2.8789
 0.6675   0.4572
-4.3492  -2.5777
-2.1426  -0.1906
-3.2990  -1.0635
 3.4883   3.0049
-0.4246   1.8002
-4.1493  -4.3651
-2.7869  -4.2379
-1.4406   0.5573
 2.7980   1.8411
 0.5664  -1.6672
 2.2836   3.5904
 1.3770   0.6986
 0.4890   0.5335
-1.9642  -0.1319
 2.4285   2.6336
-0.3784   3.1286
-2.8231  -1.6009
-4.2976  -5.2750
 3.5486   3.6262
 1.9001  -1.1097
-3.1206   1.4419
-0.7256  -1.4497
 2.4054   4.1727
-2.1556  -2.4763
 1.1376  -1.8030
 1.8774   2.8390
-1.5116  -3.3886
-0.5300   1.2957
 2.4676   1.9849
 2.3387   0.7131
 1.1430  -0.1816
 1.6810   5.0723
 2.2957   0.2102
-2.3182  -1.2482
-0.3070  -2.0157
 3.1047  -0.3879
-0.8290  -1.4787
-0.2529   3.5673
 4.9637   1.2441
-1.0856  -1.8049
 0.9636  -0.4774
-1.1425   1.1216
-0.3082  -2.8078
 2.1056   2.2377
-1.0667  -3.1439
 0.9777   0.1948
 1.5877  -3.1229
-2.8973   0.0450
 1.4851   2.1275
-5.6618  -3.6301
 0.6447   1.0033
 5.4830   2.6136
 3.8822   4.0655
 0.3628   0.4932
 0.4042   2.4230
 1.5225   0.4423
```

```
    2.3787   -0.3315
    0.2533    0.1000
    4.2920    1.3767
   -1.0073    1.5270
   -5.0477   -2.0585
   -2.5526   -2.7092
   -0.4254   -0.4169
    0.4627    2.7596
   -2.3867   -1.7475
    3.1042   -0.0629


%Sample Covariance
S

S =

    7.8117    4.8406
    4.8406    6.3410


%Pearson Correlation Coefficient Matrix R
R

R =

    1.0000    0.6878
    0.6878    1.0000


%V
V

V =

    0.6518   -0.7584
   -0.7584   -0.6518


%LAMBDA
LAMBDA

LAMBDA =

    2.1802         0
         0   11.9725


%Covariance inverse S^-1
S_inv

S_inv =

    0.2429   -0.1854
   -0.1854    0.2993


%Distance Values
my_distances
```

```
my_distances =

    2.9709
    0.2551
    4.3653
    6.0332
    0.6125
    6.5428
    1.3535
    0.9615
    1.3048
    2.2669
    8.0432
    4.4785
    1.2396
    0.5646
    1.9093
    2.2455
    0.8359
    1.4293
    0.1060
    2.5669
    0.8547
    0.2352
    0.6458
    0.2360
    8.9121
    1.2688
    4.1650
    3.3619
    3.3137
    0.9561
    0.6699
    1.0821
    6.2965
    0.0576
    2.4263
    0.9746
    1.6812
    1.7714
    1.2972
    3.1688
    2.8825
    0.8948
    1.0060
    1.2599
    2.0848
    0.2499
    0.0465
    0.8463
    1.1369
    3.4035
    1.0272
    4.4084
    2.2229
    2.0273
    4.6563
    0.3668
    2.8950
    0.9847
    2.0478
```

```
    1.2922
    2.0924
    0.8254
    0.8421
    0.8623
    0.4041
    5.2253
    1.1144
    0.6988
    1.0094
    2.8328
    0.3668
    4.1590
    4.1580
    0.5347
    0.4643
    1.1688
    2.0617
    0.8285
    1.9911
    0.1729
    5.3696
    2.0880
    0.7189
    4.1095
    0.1624
    4.0331
    2.7554
    0.0384
    1.4338
    0.3719
    1.6996
    0.0092
    2.8507
    1.5148
    3.6044
    1.2152
    0.0302
    1.8579
    0.7511
    2.4141


disp('I have picked 0.05 for the alpha value');
I have picked 0.05 for the alpha value
disp('From the chi-sqare table, the critical c_squared is = 5.99');
From the chi-sqare table, the critical c_squared is = 5.99

%outlier f_id
out_fid

out_fid =

     4
     6
    11
    25
    33
```

```
echo off
```

Published with MATLAB® R2018b

Problem 2a

```
clear all
clc
u = [98 0 2 0]';
A = [3/5 1/5 1/10 0; 1/5 2/5 0 0; 0 1/5 8/10 1/10; 1/5 1/5 1/10 9/10];

for i = 1:42
    u = A * u;
end

diary vj_problem2a.txt
echo on

disp('Steady State u:');
u

disp('It took k = 42 iterations to reach 4-decimal convergence.');

[V, LAMBDA] = eig(A);


echo off
```

```
disp('Steady State u:');
Steady State u:
u

u =

    9.3750
    3.1250
   31.2500
   56.2500


disp('It took k = 42 iterations to reach 4-decimal convergence.');
It took k = 42 iterations to reach 4-decimal convergence.

[V, LAMBDA] = eig(A);


echo off
```

Published with MATLAB® R2018b

Problem 2B

```
clear all
clc

H = [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0;
    0.11,0.17,0,0,0,0,0,0,0,0,0,0,0,0,0;
    0.11,0.17,0,0,0,0,0,0,0,0,0,0,0,0,0;
    0.11,0,0,0.33,0,0,0,0,0,0,0,0,0,0,0;
    0.11,0,0,0,0,0.2,0,0,0,0,0,0,0,0,0;
    0.11,0,0,0,0,0.2,0.2,0,0,0,0,0,0,0,0;
    0.11,0.17,0,0.33,0,0,0,0,0,0,0,0,0.25,0.17,0;
    0.11,0,0,0,0,0,0,0,0.5,0.33,0,0,0,0,0;
    0.11,0.17,0,0,0,0,0,0.5,0,0.33,0,0,0,0,0;
    0.11,0.17,0.33,0,0,0.2,0.2,0.5,0.5,0.33,0.25,0,0,0.17,0;
    0,0.17,0.33,0.33,1,0.2,0.2,0,0,0,0.25,0.25,0.25,0.17,0.33;
    0,0,0.33,0,0,0.2,0,0,0,0,0.25,0.25,0.25,0.17,0.33;
    0,0,0,0,0,0,0.2,0,0,0,0.25,0.25,0.25,0.17,0.33;
    0,0,0,0,0,0,0.2,0,0,0,0,0,0,0.17,0;
    0,0,0,0,0,0,0,0,0,0,0,0.25,0,0,0];
n = 15;
alpha = 0.8;
G = alpha * H + (1 - alpha)/n * ones(15,15);

I = zeros(15,1);
I(1) = 100;
a = [0 0 0 0 0 00 0 0 0 0 0  0 0 0 0]';
for ii = 1:4000
    I = G * I;
    a(:,ii+1) = I;

    if abs(a(:,ii+1) - a(:,ii)) < 0.00005
        break
    end
end

I_inf = a(:,3788);

diary vj_problem2b.txt
echo on

% H Matrix
H

% Google Matrix
G

%Final State I_inf
I_inf

echo off
```

```
% H Matrix
H

H =

  Columns 1 through 7

         0         0         0         0         0         0         0
    0.1100    0.1700         0         0         0         0         0
    0.1100    0.1700         0         0         0         0         0
    0.1100         0         0    0.3300         0         0         0
    0.1100         0         0         0         0    0.2000         0
    0.1100         0         0         0         0    0.2000    0.2000
    0.1100    0.1700         0    0.3300         0         0         0
    0.1100         0         0         0         0         0         0
    0.1100    0.1700         0         0         0         0         0
    0.1100    0.1700    0.3300         0         0    0.2000    0.2000
         0    0.1700    0.3300    0.3300    1.0000    0.2000    0.2000
         0         0    0.3300         0         0    0.2000         0
         0         0         0         0         0         0    0.2000
         0         0         0         0         0         0    0.2000
         0         0         0         0         0         0         0

  Columns 8 through 14

         0         0         0         0         0         0         0
         0         0         0         0         0         0         0
         0         0         0         0         0         0         0
         0         0         0         0         0         0         0
         0         0         0         0         0         0         0
         0         0         0         0         0         0         0
         0         0         0         0         0    0.2500    0.1700
         0    0.5000    0.3300         0         0         0         0
    0.5000         0    0.3300         0         0         0         0
    0.5000    0.5000    0.3300    0.2500         0         0    0.1700
         0         0         0    0.2500    0.2500    0.2500    0.1700
         0         0         0    0.2500    0.2500    0.2500    0.1700
         0         0         0    0.2500    0.2500    0.2500    0.1700
         0         0         0         0         0         0    0.1700
         0         0         0         0    0.2500         0         0

  Column 15

         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
```

```
    0.3300
    0.3300
    0.3300
         0
         0


% Google Matrix
G

G =

  Columns 1 through 7

    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133
    0.1013    0.1493    0.0133    0.0133    0.0133    0.0133    0.0133
    0.1013    0.1493    0.0133    0.0133    0.0133    0.0133    0.0133
    0.1013    0.0133    0.0133    0.2773    0.0133    0.0133    0.0133
    0.1013    0.0133    0.0133    0.0133    0.0133    0.1733    0.0133
    0.1013    0.0133    0.0133    0.0133    0.0133    0.1733    0.1733
    0.1013    0.1493    0.0133    0.2773    0.0133    0.0133    0.0133
    0.1013    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133
    0.1013    0.1493    0.0133    0.0133    0.0133    0.0133    0.0133
    0.1013    0.1493    0.2773    0.0133    0.0133    0.1733    0.1733
    0.0133    0.1493    0.2773    0.2773    0.8133    0.1733    0.1733
    0.0133    0.0133    0.2773    0.0133    0.0133    0.1733    0.0133
    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133    0.1733
    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133    0.1733
    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133

  Columns 8 through 14

    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133
    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133
    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133
    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133
    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133
    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133
    0.0133    0.0133    0.0133    0.0133    0.0133    0.2133    0.1493
    0.0133    0.4133    0.2773    0.0133    0.0133    0.0133    0.0133
    0.4133    0.0133    0.2773    0.0133    0.0133    0.0133    0.0133
    0.4133    0.4133    0.2773    0.2133    0.0133    0.0133    0.1493
    0.0133    0.0133    0.0133    0.2133    0.2133    0.2133    0.1493
    0.0133    0.0133    0.0133    0.2133    0.2133    0.2133    0.1493
    0.0133    0.0133    0.0133    0.2133    0.2133    0.2133    0.1493
    0.0133    0.0133    0.0133    0.0133    0.0133    0.0133    0.1493
    0.0133    0.0133    0.0133    0.0133    0.2133    0.0133    0.0133

  Column 15

    0.0133
    0.0133
    0.0133
    0.0133
```

```
    0.0133
    0.0133
    0.0133
    0.0133
    0.0133
    0.0133
    0.2773
    0.2773
    0.2773
    0.0133
    0.0133


%Final State I_inf
I_inf

I_inf =

    0.0018
    0.0022
    0.0022
    0.0026
    0.0025
    0.0034
    0.0059
    0.0159
    0.0162
    0.0285
    0.0170
    0.0131
    0.0129
    0.0031
    0.0044


echo off
```

Published with MATLAB® R2018b

Problem 3a

```
clear all
clc

A = [ 10/3 -5/3 0 5/3 2;
    -1/3 -4/3 0 1/3 4;
    2 -1 -4 -2 3;
    10 -23 -27 -5 33;
    5/3 -13/3 -6 -5/3 9];

x_guess = [-1 0 0 0 0; -1 -1 0 0 0; 0 0 1 0 0; 0 0 1 1 1; 0 0 0 1 1];

x_guess1 = [-1 0 0 0 0]';
x_guess2 = [0 -1 0 0 0]';
x_guess3 = [0 1 1 1 0]';
x_guess4 = [0 0 -1 1 1]';
x_guess5 = [0 0 0 1 1]';


% x_guess0 = [x_guess1; x_guess2; x_guess3; x_guess4; x_guess5]';

%normalize
% xg1norm = x_guess1/norm(x_guess1);
% xg2norm = x_guess2/norm(x_guess2);
% xg3norm = x_guess3/norm(x_guess3);
% xg4norm = x_guess4/norm(x_guess4);
% xg5norm = x_guess5/norm(x_guess5);


[xg1, yg1, itr_fin1, eig_S1, lambda1 ] = riterq(A, x_guess1, 1000);
[xg2, yg2, itr_fin2, eig_S2, lambda2 ] = riterq(A, x_guess2, 1000);
[xg3, yg3, itr_fin3, eig_S3, lambda3 ] = riterq(A, x_guess3, 1000);
[xg4, yg4, itr_fin4, eig_S4, lambda4 ] = riterq(A, x_guess4, 1000);
[xg5, yg5, itr_fin5, eig_S5, lambda5 ] = riterq(A, x_guess5, 1000);


%echo
echo on
diary vj_problem3a.txt
%intial xguess
x_guess1
x_guess2
x_guess3
x_guess4
x_guess5

%yguess history
yg1
yg2
yg3
yg4
```

```
yg5

%best guess xk final
eig_S1
eig_S2
eig_S3
eig_S4
eig_S5

Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.106200e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  5.018128e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.106200e-18.
diary vj_problem3a.txt
%intial xguess
x_guess1

x_guess1 =

    -1
     0
     0
     0
     0

x_guess2

x_guess2 =

     0
    -1
     0
     0
     0

x_guess3

x_guess3 =

     0
     1
     1
     1
     0

x_guess4

x_guess4 =
```

```
     0
     0
    -1
     1
     1

x_guess5

x_guess5 =

     0
     0
     0
     1
     1


%yguess history
yg1

yg1 =

  Columns 1 through 7

    3.3333    3.8366    3.5553    3.2873    2.9662    2.9994    3.0000

  Column 8

    3.0000

yg2

yg2 =

   -1.3333   -1.2569   -1.0159   -1.0001   -1.0000   -1.0000

yg3

yg3 =

  -21.0000  -14.3344   -9.0795   -7.2711   -7.0057   -7.0000   -7.0000

yg4

yg4 =

  Columns 1 through 7

   21.1111   13.6219    7.8920    5.6692    5.1041    5.0045    5.0000

  Columns 8 through 10

    5.0000    5.0000    5.0000
```

```
yg5

yg5 =

  Columns 1 through 7

   17.6667   10.3264    6.5796    5.3298    5.0361    5.0006    5.0000

  Columns 8 through 10

    5.0000    5.0000    5.0000


%best guess xk final
eig_S1

eig_S1 =

    0.5774
   -0.5774
    0.0000
   -0.0000
   -0.5774

eig_S2

eig_S2 =

   -0.0000
   -0.5774
    0.5774
   -0.5774
    0.0000

eig_S3

eig_S3 =

   -0.2085
   -0.2085
    0.4170
    0.8341
    0.2085

eig_S4

eig_S4 =

    0.7071
    0.0000
    0.0000
    0.7071
    0.0000
```

```
eig_S5
```

```
eig_S5 =

    0.7071
    0.0000
    0.0000
    0.7071
    0.0000
```

Published with MATLAB® R2018b

Problem 3b

```
clear all


A = [2 -3 1 3;
    1 4 -3 -3;
    5 3 -1 -1;
    3 -6 -3 1];

xguess2 = [-1 + 1i 0 0 0 ]';
xguess1 = [0 -1 0 0 ]';
xguess4 = [0 1 0 1 ]';
xguess3 = [0 0 -1 1 + 1i ]';

[xg1, yg1 ] = riterq_im(A, xguess1, 100);
[xg2, yg2 ] = riterq_im(A, xguess2, 100);
[xg3, yg3 ] = riterq_im(A, xguess3, 100);
[xg4, yg4 ] = riterq_im(A, xguess4, 100);

eig_S1 = xg1(:,100);
eig_S2 = xg2(:,100);
eig_S3 = xg3(:,100);
eig_S4 = xg4(:,100);

echo on
diary vj_problem3b.txt
%intial xguess
xguess1
xguess2
xguess3
xguess4


%yguess history
yg1
yg2
yg3
yg4


%best guess xk final
eig_S1
eig_S2
eig_S3
eig_S4


echo off
```

Warning: Matrix is close to singular or badly scaled. Results may be inaccurate.

```
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.063738e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.243959e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  9.514365e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  3.872066e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  9.514365e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.942155e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  9.514365e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.378591e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
```

```
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.243959e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  9.514365e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  9.514365e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.523042e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.523042e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  9.514365e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  9.514365e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
```

```
RCOND =  1.961437e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.063738e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.961437e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.378591e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.961437e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.378591e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.378591e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.378591e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.917682e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
```

```
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.378591e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  9.514365e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.378591e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  3.813172e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.063738e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.243959e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  9.514365e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.378591e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.961437e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.243959e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.063738e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
```

```
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.063738e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.665014e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  9.514365e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.378591e-18.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.243959e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.665014e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.243959e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
```

```
RCOND =  1.189296e-17.
Warning: Matrix is close to singular or badly scaled. Results may be
 inaccurate.
RCOND =  2.378591e-18.
diary vj_problem3b.txt
%intial xguess
xguess1

xguess1 =

     0
    -1
     0
     0

xguess2

xguess2 =

  -1.0000 - 1.0000i
   0.0000 + 0.0000i
   0.0000 + 0.0000i
   0.0000 + 0.0000i

xguess3

xguess3 =

   0.0000 + 0.0000i
   0.0000 + 0.0000i
  -1.0000 + 0.0000i
   1.0000 - 1.0000i

xguess4

xguess4 =

     0
     1
     0
     1



%yguess history
yg1

yg1 =

  Columns 1 through 7

    4.0000    4.1165    3.1734    0.7835    2.8328    1.6131   -0.5326

  Columns 8 through 14
```

```
  0.8183    2.7213    1.6026   -0.4342    0.9070    2.5612    1.6705

Columns 15 through 21

 -0.3718    1.0537    1.8628    1.5056   -0.4391    0.7365    2.7516

Columns 22 through 28

  1.4640   -0.3990    0.6822    2.7227    1.3565   -0.1242    0.6344

Columns 29 through 35

  2.6768    1.2531    0.4174    0.7881    2.7474    1.5523   -0.4507

Columns 36 through 42

  0.8129    2.7313    1.5884   -0.4403    0.8805    2.6291    1.6579

Columns 43 through 49

 -0.3854    1.0259    2.0373    1.5733   -0.4464    0.8515    2.6844

Columns 50 through 56

  1.6340   -0.4086    0.9739    2.3121    1.6513   -0.3923    1.0113

Columns 57 through 63

  2.1214    1.6015   -0.4331    0.9064    2.5631    1.6703   -0.3720

Columns 64 through 70

  1.0533    1.8654    1.5067   -0.4397    0.7381    2.7520    1.4670

Columns 71 through 77

 -0.4030    0.6856    2.7253    1.3637   -0.1512    0.6321    2.6743

Columns 78 through 84

  1.2480    0.4507    0.8020    2.7396    1.5731   -0.4465    0.8511

Columns 85 through 91

  2.6850    1.6336   -0.4090    0.9730    2.3162    1.6521   -0.3914

Columns 92 through 98

  1.0130    2.1114    1.5983   -0.4350    0.9001    2.5807    1.6681

Columns 99 through 101

 -0.3744    1.0486    1.8966
```

```
yg2

yg2 =

  Columns 1 through 7

    2.0000    1.6719   -0.3524    1.4243   -0.1162    0.6739    2.7117

  Columns 8 through 14

    1.3620   -0.1335    0.6349    2.6775    1.2553    0.4040    0.7826

  Columns 15 through 21

    2.7496    1.5437   -0.4508    0.7979    2.7423    1.5671   -0.4482

  Columns 22 through 28

    0.8398    2.7017    1.6216   -0.4190    0.9478    2.4233    1.6685

  Columns 29 through 35

   -0.3740    1.0493    1.8920    1.5177   -0.4452    0.7549    2.7542

  Columns 36 through 42

    1.4974   -0.4336    0.7246    2.7478    1.4416   -0.3637    0.6597

  Columns 43 through 49

    2.7031    1.3089    0.0914    0.6755    2.7172    1.3425   -0.0670

  Columns 50 through 56

    0.6417    2.6848    1.2693    0.3152    0.7478    2.7537    1.4848

  Columns 57 through 63

   -0.4230    0.7074    2.7397    1.4081   -0.2905    0.6372    2.6799

  Columns 64 through 70

    1.2593    0.3772    0.7718    2.7527    1.5262   -0.4481    0.7684

  Columns 71 through 77

    2.7533    1.5205   -0.4463    0.7592    2.7541    1.5049   -0.4387

  Columns 78 through 84

    0.7354    2.7513    1.4621   -0.3965    0.6801    2.7210    1.3522

  Columns 85 through 91
```

```
   -0.1071    0.6362    2.6789    1.2572    0.3909    0.7773    2.7513

  Columns 92 through 98

    1.5352   -0.4500    0.7832    2.7494    1.5447   -0.4509    0.7995

  Columns 99 through 101

    2.7413    1.5695   -0.4476

yg3

yg3 =

  Columns 1 through 4

   1.6667 + 0.6667i  -1.0211 + 1.4972i  -0.2174 + 0.6925i   1.4638 +
 1.4140i

  Columns 5 through 8

   1.0460 + 2.0709i   1.1469 + 2.0057i   1.1516 + 2.0075i   1.1516 +
 2.0075i

  Columns 9 through 12

   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
 2.0075i

  Columns 13 through 16

   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
 2.0075i

  Columns 17 through 20

   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
 2.0075i

  Columns 21 through 24

   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
 2.0075i

  Columns 25 through 28

   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
 2.0075i

  Columns 29 through 32

   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
 2.0075i
```

```
 Columns 33 through 36

  1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

 Columns 37 through 40

  1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

 Columns 41 through 44

  1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

 Columns 45 through 48

  1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

 Columns 49 through 52

  1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

 Columns 53 through 56

  1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

 Columns 57 through 60

  1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

 Columns 61 through 64

  1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

 Columns 65 through 68

  1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

 Columns 69 through 72

  1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

 Columns 73 through 76
```

```
   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

  Columns 77 through 80

   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

  Columns 81 through 84

   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

  Columns 85 through 88

   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

  Columns 89 through 92

   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

  Columns 93 through 96

   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

  Columns 97 through 100

   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 + 2.0075i   1.1516 +
2.0075i

  Column 101

   1.1516 + 2.0075i

yg4

yg4 =

  Columns 1 through 7

   -2.0000   -0.5994    0.3499    0.4078    0.4694    2.1801    0.4434

  Columns 8 through 14

    2.4333    0.8548    2.6940    1.6372   -0.4060    0.9816    2.2770

  Columns 15 through 21

    1.6440   -0.3994    0.9954    2.2065    1.6264   -0.4151    0.9578

  Columns 22 through 28
```

```
    2.3829    1.6635   -0.3795    1.0382    1.9624    1.5457   -0.4509

  Columns 29 through 35

    0.8012    2.7401    1.5720   -0.4469    0.8490    2.6883    1.6315

  Columns 36 through 42

   -0.4109    0.9685    2.3365    1.6559   -0.3875    1.0215    2.0631

  Columns 43 through 49

    1.5823   -0.4431    0.8686    2.6539    1.6493   -0.3942    1.0070

  Columns 50 through 56

    2.1449    1.6087   -0.4284    0.9212    2.5184    1.6729   -0.3690

  Columns 57 through 63

    1.0592    1.8266    1.4902   -0.4279    0.7146    2.7435    1.4223

  Columns 64 through 70

   -0.3246    0.6450    2.6883    1.2767    0.2705    0.7315    2.7502

  Columns 71 through 77

    1.4548   -0.3858    0.6723    2.7146    1.3359   -0.0381    0.6465

  Columns 78 through 84

    2.6899    1.2799    0.2514    0.7248    2.7479    1.4421   -0.3646

  Columns 85 through 91

    0.6602    2.7035    1.3098    0.0865    0.6742    2.7162    1.3398

  Columns 92 through 98

   -0.0553    0.6435    2.6868    1.2734    0.2902    0.7386    2.7521

  Columns 99 through 101

    1.4680   -0.4043    0.6867


%best guess xk final
eig_S1

eig_S1 =
```

```
    0.1831
   -0.1795
    0.7076
   -0.6584

eig_S2

eig_S2 =

  -0.1938 - 0.1938i
   0.0663 + 0.0663i
  -0.5430 - 0.5430i
   0.4040 + 0.4040i

eig_S3

eig_S3 =

   0.3033 - 0.0360i
   0.1214 + 0.2383i
   0.4741 - 0.4775i
  -0.0983 + 0.6105i

eig_S4

eig_S4 =

   -0.3216
   -0.5061
    0.1264
   -0.7902



echo off
```

Published with MATLAB® R2018b

```
%%Problem 4
close all
clear all

A = imread('Avengers_Endgame_BW.tif');
nbits = 8;
A = single(A);
A = -(A - (2^nbits-1)/2);

s = svd(A);

[U, S, V] = svd(A);

f0 = figure('Name', 'Semilogy sigmai');

h1 = semilogy(s);
xlabel('nth singular value of A')
ylabel('Singular values')
title('Singular Values corresponding with nth value of A')

grid on
A1 = [];
A2 = [];
A3 = [];
N1 = 0;
N2 = 0;
N3 = 0;
N1a = 70;
N2a = 200;
N3a = 600;
for kk = 1:70

    A1 = s(kk) * U(:,kk) * V(:,kk)';
    N1 = N1 + A1;

end

for kk = 1:200

    A2 = s(kk) * U(:,kk) * V(:,kk)';
    N2 = N2 + A2;

end

 for kk = 1:600

    A3 = s(kk) * U(:,kk) * V(:,kk)';
    N3 = N3 + A3;

 end

N1g = mat2gray(N1);
N2g = mat2gray(N2);
N3g = mat2gray(N3);

figure
imshow(N1g)
title('N1 sums')
```

```
figure
imshow(N2g)
title('N2 sums')

figure
imshow(N3g)
title('N3 sums')

A = A';
x1 = A(:);
B = N1';
B2 = N2';
B3 = N3';
x2 = B(:);
x3 = B2(:);
x4 = B3(:);
x1m = mean(x1);
x2m = mean(x2);
x3m = mean(x3);
x4m = mean(x4);
x1mre = repmat(x1m, length(x1),1);
d1 = x1 - x1mre;
d1norm = d1/norm(d1);

x2mre = repmat(x2m, length(x2),1);
d2 = x2 - x1mre;
d2norm = d2/norm(d2);

x3mre = repmat(x3m, length(x3),1);
d3 = x3 - x1mre;
d3norm = d3/norm(d3);

x4mre = repmat(x4m, length(x4),1);
d4 = x4 - x1mre;
d4norm = d4/norm(d4);

zz = 100;

figure
plot(x1,x2,'.')
title('N1 Reconstructed vs Original');
xlabel('original image intensities');
ylabel('SVD-reconstructed pixel intensities');
hold on
D_n1 = [d1 d2];
S_n1 = 1/(length(D_n1)-1) * D_n1' * D_n1;
[V_n1, l_n1] = eig(S_n1);
ev1 = line([x1m zz*V_n1(1,1)+x1m x2m  ], [x2m zz*V_n1(2,1)+x2m x1m ]);
ev1.Color = 'm';
% ev1.LineWidth = zz;
ev2 = line([x1m zz*V_n1(1,2)+x1m x2m  ], [x2m zz*V_n1(2,2)+x2m x1m ]);
ev2.Color = 'b';
% ev2.LineWidth = zz;
axis square
axis([-255 255 -255 255])
lamsum_n1 = l_n1(1,1) + l_n1(2,2);
pvar_n1v1 = l_n1(1,1)/lamsum_n1;
pvar_n1v2 = l_n1(2,2)/lamsum_n1;
R_n1 = D_n1'*D_n1;
```

```
figure
plot(x1,x3,'.')
title('N2 Reconstructed vs Original');
xlabel('original image intensities');
ylabel('SVD-reconstructed pixel intensities');
D_n2 = [d1 d3];
S_n2 = 1/(length(D_n2)-1) * D_n2' * D_n2;
[V_n2, l_n2] = eig(S_n2);
hold on
ev1 = line([x1m zz*V_n2(1,1)+x1m  x3m ], [x3m zz*V_n2(2,1)+x3m  x1m]);
ev1.Color = 'm';
% ev1.LineWidth = zz;
ev2 = line([x1m zz*V_n2(1,2)+x1m  x3m ], [x3m zz*V_n2(2,2)+x3m x1m ]);
ev2.Color = 'b';
% ev2.LineWidth = zz;
axis square
axis([-255 255 -255 255])
lamsum_n2 = l_n2(1,1) + l_n2(2,2);
pvar_n2v1 = l_n2(1,1)/lamsum_n2;
pvar_n2v2 = l_n2(2,2)/lamsum_n2;
R_n2 = D_n2'*D_n2;

figure
plot(x1,x4,'.')
title('N3 Reconstructed vs Original');
xlabel('original image intensities');
ylabel('SVD-reconstructed pixel intensities');
D_n3 = [d1 d4];
S_n3 = 1/(length(D_n3)-1) * D_n3' * D_n3;
[V_n3, l_n3] = eig(S_n3);
hold on
ev1 = line([x1m zz*V_n3(1,1)+x1m x4m  ], [x4m zz*V_n3(2,1)+x4m x1m ]);
ev1.Color = 'm';
% ev1.LineWidth = zzz;
ev2 = line([x1m zz*V_n3(1,2)+x1m  x4m ], [x4m zz*V_n3(2,2)+x4m x1m ]);
ev2.Color = 'b';
% ev2.LineWidth = zz;
axis square
axis([-255 255 -255 255])
lamsum_n3 = l_n3(1,1) + l_n3(2,2);
pvar_n3v1 = l_n3(1,1)/lamsum_n3;
pvar_n3v2 = l_n3(2,2)/lamsum_n3;
R_n3 = D_n3'*D_n3;


diary vjprob4.txt
echo on
disp('N1 Spectra sums: ');
N1a
disp('N2 Spectra sums: ');
N2a
disp('N3 Spectra sums: ');
N3a

disp('N1 related Covariance etc.')
V_n1
l_n1
S_n1
disp('% variance explained due to the variations along the v1 principal axis: ')
pvar_n1v1
disp('% variance explained due to the variations along the v2 principal axis: ')
```

```
pvar_n1v2
disp('Pearson Correlation Coefficient matrix R:')
R_n1

disp('N2 related Covariance etc.')
V_n2
l_n2
S_n2
disp('% variance explained due to the variations along the v1 principal axis: ')
pvar_n2v1
disp('% variance explained due to the variations along the v2 principal axis: ')
pvar_n2v2
disp('Pearson Correlation Coefficient matrix R:')
R_n2

disp('N3 related Covariance etc.')
V_n3
l_n3
S_n3
disp('% variance explained due to the variations along the v1 principal axis: ')
pvar_n3v1
disp('% variance explained due to the variations along the v2 principal axis: ')
pvar_n3v2
disp('Pearson Correlation Coefficient matrix R:')
R_n3
```

```
Warning: Image is too big to fit on screen; displaying at 67%
Warning: Image is too big to fit on screen; displaying at 67%
Warning: Image is too big to fit on screen; displaying at 67%
disp('N1 Spectra sums: ');
N1 Spectra sums:
N1a

N1a =

    70

disp('N2 Spectra sums: ');
N2 Spectra sums:
N2a

N2a =

   200

disp('N3 Spectra sums: ');
N3 Spectra sums:
N3a

N3a =

   600


disp('N1 related Covariance etc.')
N1 related Covariance etc.
V_n1

V_n1 =
```

```
  2×2 single matrix

    0.6878   -0.7259
   -0.7259   -0.6878

l_n1

l_n1 =

  2×2 single matrix

   1.0e+03 *

    0.1325         0
         0    5.1771

S_n1

S_n1 =

  2×2 single matrix

   1.0e+03 *

    2.7909    2.5186
    2.5186    2.5186

disp('% variance explained due to the variations along the v1 principal axis: ')
% variance explained due to the variations along the v1 principal axis:
pvar_n1v1

pvar_n1v1 =

  single

    0.0249

disp('% variance explained due to the variations along the v2 principal axis: ')
% variance explained due to the variations along the v2 principal axis:
pvar_n1v2

pvar_n1v2 =

  single

    0.9751

disp('Pearson Correlation Coefficient matrix R:')
Pearson Correlation Coefficient matrix R:
R_n1

R_n1 =

  2×2 single matrix

   1.0e+09 *

    2.2310    2.0133
    2.0133    2.0133
```

```
disp('N2 related Covariance etc.')
```

```
N2 related Covariance etc.
```

```
V_n2
```

```
V_n2 =

  2×2 single matrix

    0.7030   -0.7112
   -0.7112   -0.7030
```

```
l_n2
```

```
l_n2 =

  2×2 single matrix

   1.0e+03 *

    0.0313         0
         0    5.4876
```

```
S_n2
```

```
S_n2 =

  2×2 single matrix

   1.0e+03 *

    2.7909    2.7280
    2.7280    2.7280
```

```
disp('% variance explained due to the variations along the v1 principal axis: ')
```

```
% variance explained due to the variations along the v1 principal axis:
```

```
pvar_n2v1
```

```
pvar_n2v1 =

  single

    0.0057
```

```
disp('% variance explained due to the variations along the v2 principal axis: ')
```

```
% variance explained due to the variations along the v2 principal axis:
```

```
pvar_n2v2
```

```
pvar_n2v2 =

  single

    0.9943
```

```
disp('Pearson Correlation Coefficient matrix R:')
```

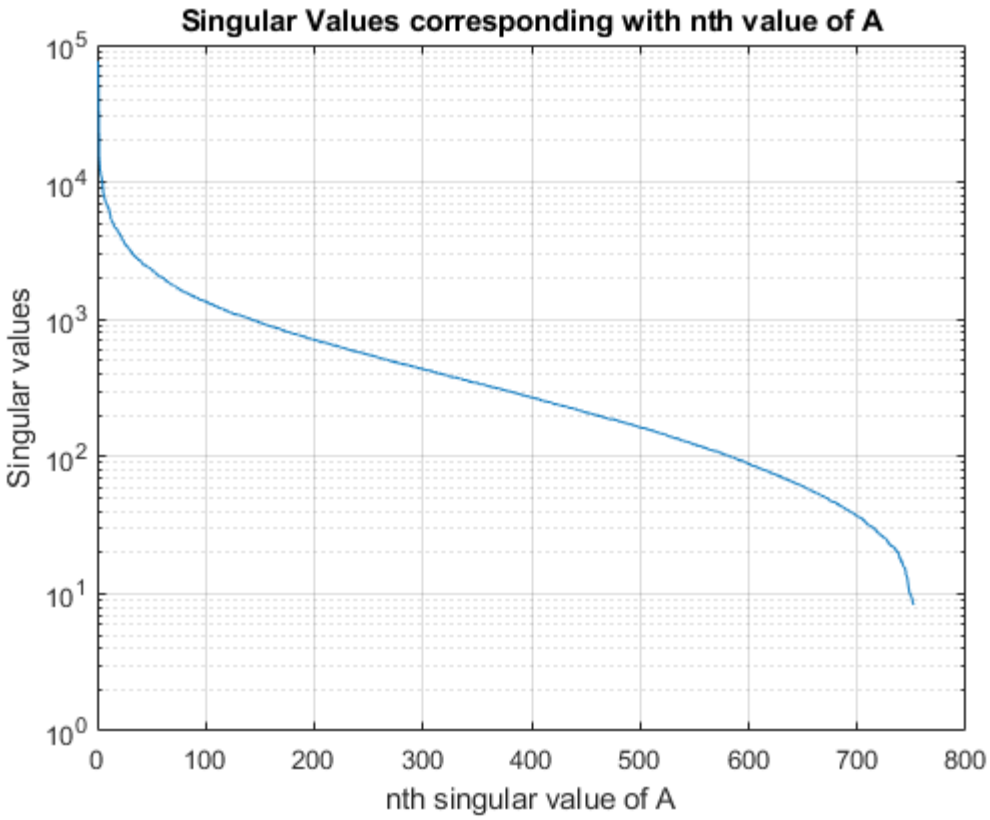
```
Pearson Correlation Coefficient matrix R:
```

```
R_n2
```

```
R_n2 =

  2×2 single matrix
```

```
   1.0e+09 *

    2.2310    2.1807
    2.1807    2.1807


disp('N3 related Covariance etc.')
N3 related Covariance etc.
V_n3

V_n3 =

  2×2 single matrix

    0.7071   -0.7071
   -0.7071   -0.7071

l_n3

l_n3 =

  2×2 single matrix

   1.0e+03 *

    0.0003         0
         0    5.5811

S_n3

S_n3 =

  2×2 single matrix

   1.0e+03 *

    2.7909    2.7904
    2.7904    2.7904

disp('% variance explained due to the variations along the v1 principal axis: ')
% variance explained due to the variations along the v1 principal axis:
pvar_n3v1

pvar_n3v1 =

  single

  4.6782e-05

disp('% variance explained due to the variations along the v2 principal axis: ')
% variance explained due to the variations along the v2 principal axis:
pvar_n3v2

pvar_n3v2 =

  single

    1.0000
```

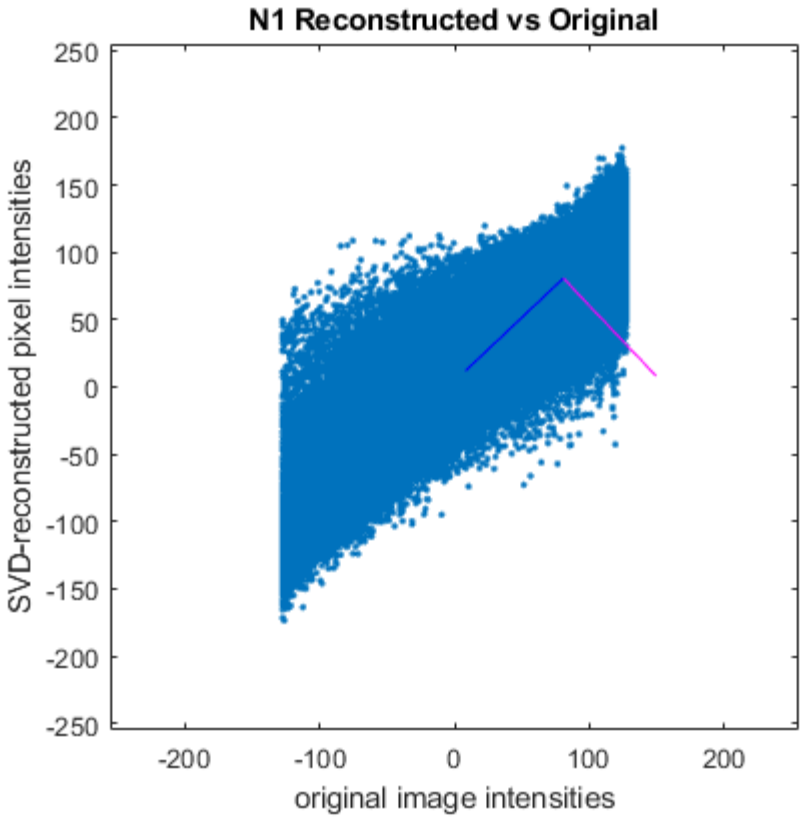
```
disp('Pearson Correlation Coefficient matrix R:')
Pearson Correlation Coefficient matrix R:
R_n3


R_n3 =

  2×2 single matrix

   1.0e+09 *

    2.2310    2.2306
    2.2306    2.2306
```

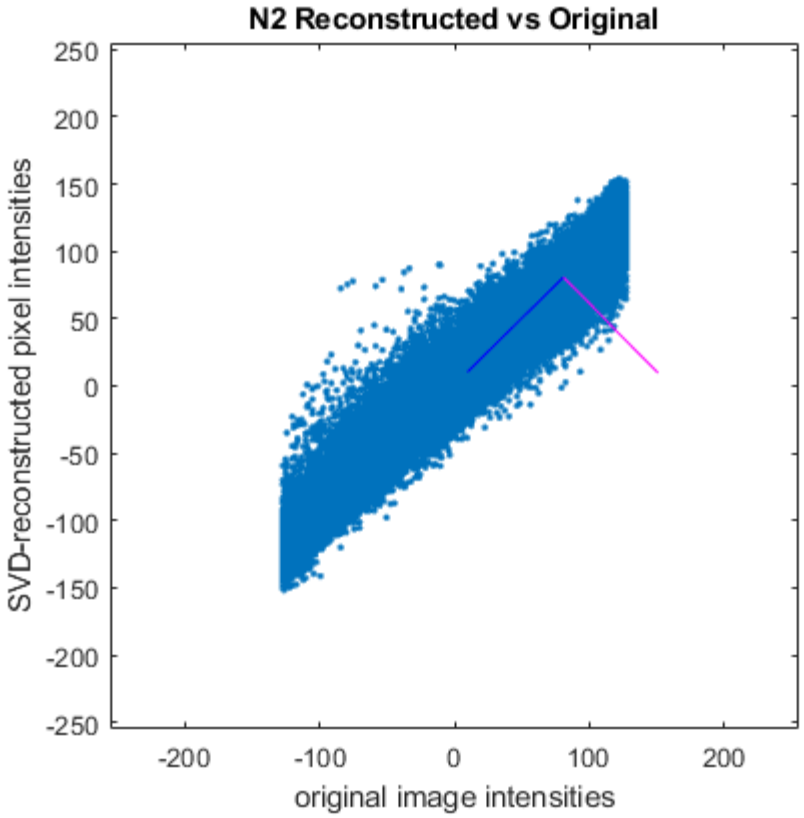


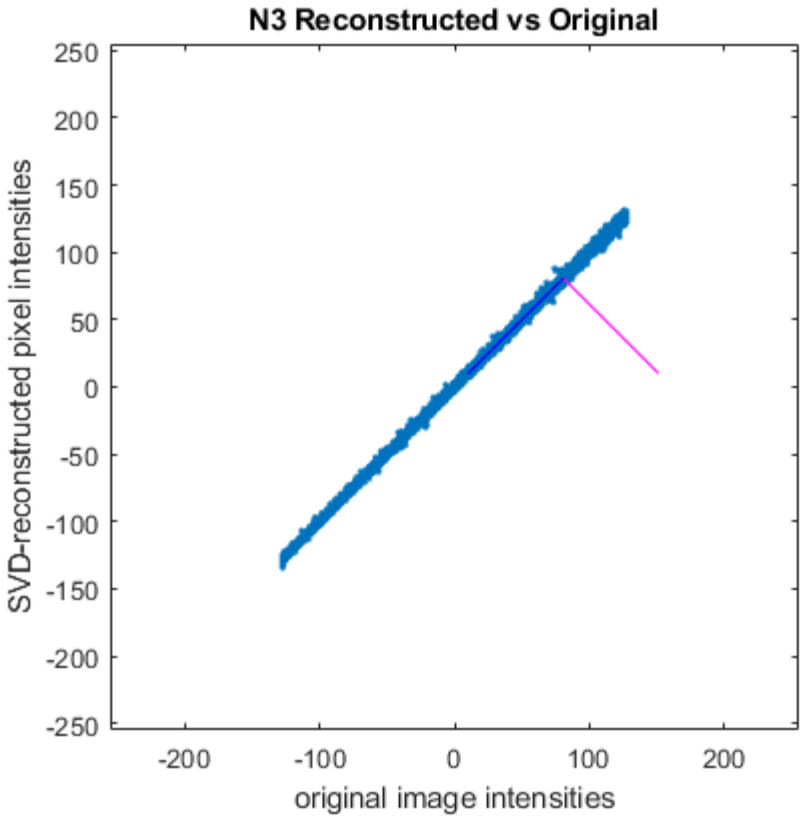
N1 sums

N2 sums

N3 sums


N1 Reconstructed vs Original
SVD-reconstructed pixel intensities
original image intensities
250
200
150
100
50
0
-50
-100
-150
-200
-250
-200
-100
0
100
200


N2 Reconstructed vs Original
SVD-reconstructed pixel intensities
original image intensities
250
200
150
100
50
0
-50
-100
-150
-200
-250
-200
-100
0
100
200



Published with MATLAB® R2018b