W.D. Henshaw **Math 6800: Problem Set 2** **Due:** Thurs. Sept. 22, 2022.

NLA = the text-book *Numerical Linear Algebra*, by Trefethen and Bau

1. NLA exercise 4.1 *Determine SVDs of...*

2. NLA exercise 4.4 *Two matrices ...*

3. NLA exercise 5.1 *In example 3.1 ...*

4. NLA exercise 5.3 *Consider the matrix ...*

5. Use the SVD to show that if $A \in \mathbb{C}^{m\times n}$ has rank n then

$$\|A(A^*A)^{-1}A^*\|_2 = 1.$$

6. Create your own Matlab code to compress an image using the SVD by typing in the code below. Replace line 8 with the name of an interesting image (in gif format) that you would like to compress. Make plots of the original image and some compressed images by choosing 2 or 3 different values for k (the number of terms in the SVD to keep) to illustrate how the quality varies. Also save a plot of the size of the singular values versus k (figure 3 in the code below).

Listing 1: svdCompress.m

```
1  %
2  %  Compress an image with the SVD
3  %
4
5  clear;
6
7  k=100;                          % number of singular values to keep
8  image = 'heatedRoom2dT18.gif';  % Here is an image
9
10 % Set defaults for plotting
11 fontSize=18; lineWidth=2; markerSize=8;
12 fontSize2=24;
13 set(0,'DefaultLineMarkerSize',markerSize);
14 set(0,'DefaultLineLineWidth',lineWidth);
15 set(0,'DefaultAxesFontSize',fontSize2);
16 set(0,'DefaultLegendFontSize',fontSize2);
17
18 %-- read the image ---
19 [A,map] = imread(image);
20
21 figure(1);
22 imshow(A,map);  % plot original image
23
24 % convert to an array or rgb values B(1:m,1:m,1:3)
25 if ~isempty(map)
26   B = ind2rgb(A,map);
27 else
28   B = im2double(A);
29 end;
```

```
30
31 [m,n,p] = size(B);
32
33 figure(3);
34 rgbc{1}='r'; rgbc{2}='g'; rgbc{3}='b';
35
36 for rgb=1:3
37   % ---- COMPUTE THE SVD ----
38   [U,S,V] = svd(B(1:m,1:n,rgb));
39
40   % -- plot singular values ---
41   Sd = diag(S);
42   semilogy( 1:k, Sd(1:k)+1e-20, rgbc{rgb} ); hold on;
43
44   % Form the rank k approximation
45   %  C = sigma1*u1*v1^* + ... + sigmak*uk*vk^*
46   C(1:m,1:n,rgb) = U(1:m,1:k)*S(1:k,1:k)*(V(1:n,1:k)');
47
48 end;
49
50 hold off;
51 title('Singular␣values␣of␣RGB'); legend('r','g','b'); grid on;
52
53 % -- plot the rank k approximation ---
54 figure(2);
55 imshow(C);
56
57 fprintf('Image␣size:␣m=%d,␣n=%d,␣␣Keep␣k=%d␣singular␣values.\n',m,n,k);
```