Math 170A, Fall 2023 ***HOMEWORK #5*** ***due Friday, Nov 17***

Homework problems that will be graded (Q1 - Q5, 30pts in total):

Q1. Let $A \in \mathbb{R}^{n\times m}$, $n \geq m$, $\mathrm{rank}(A) = m$. Compute the SVD of the matrix $(A^T A)^{-1}A^T$ in terms of the SVD of $A = U\Sigma V^T$, and explain what the parts are.

a) Based on the definitions in class, conclude that the pseudoinverse of A has the format $A^\dagger = (A^T A)^{-1}A^T$.

b) Use (a) to calculate $A^\dagger A$.

c) Use the SVDs of A and $A^\dagger$ to calculate $AA^\dagger$, simplifying as much as possible.

Q2. Let

$$A = U \cdot \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot V,$$

$$B = P \cdot \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot Q,$$

where $U \in \mathbb{R}^{3\times3}, V \in \mathbb{R}^{3\times3}$, $P \in \mathbb{R}^{3\times3}, Q \in \mathbb{R}^{4\times4}$ are all given orthogonal matrices.

Compute $\|A\|_2$, $\|A^{-1}\|_2$, $\kappa_2(A)$, $\|B\|_2$ and the pseudoinverse $B^\dagger$ (the answer will be in terms of P and Q).

Q3. Let $A \in \mathbb{R}^{n\times m}$, $n \geq m$.

a) Use the SVD of A to deduce the SVD of $A^T A$.

b) If $m = n$ and A is full-rank, use a) to show that $\|A^T A\|_2 = \|A\|_2^2$ and that $\kappa_2(A^T A) = \kappa_2(A)^2$.

Q4. Work this exercise using pencil and paper. You can use MATLAB to check your work. Let A be the following exterior (or outer) product of two vectors:

$$A = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \end{bmatrix}$$

Note that A is 3×2. Answer the questions below; you can do so ***without ever forming*** A ***explicitly***.

a) What is the rank r of A?

b) Think about the "sum of rank one matrices" expression for the SVD of A, then consider the reduced SVD of A: $A = U_r \Sigma_r V_r^T$. What are the sizes of U_r, Σ_r, and V_r?

c) Use the fact that the columns of U_r and V_r are orthonormal to figure out U_r, V_r, and Σ_r.

Q5. Run the attached `low_rank_approximation.m` MATLAB code (**on the next page**).

a) Explain line by line what the code does (you might need to google some of the commands).

b) Explain what the algorithm, as a whole, does.

c) Note that the approximation gets better as we increase k. Even when $k = 32$, the resulting approximation looks reasonable. What is the advantage to use/store the $k = 32$ approximation instead of the original image? What is the disadvantage?

```
% load image

A = imread('street2.jpg');
A = rgb2gray(A);

B = double(A);

% compute SVD

size(B)
r = rank(B)
[U,S,V] = svd(B);

% approximate image

ranks = [1 2 4 8 16 32 64 r];
l = length(ranks);

for i = 1:l

    % compute rank i approximation

    k = ranks(i);

    approxB = U(:,1:k)*S(1:k,1:k)*V(:,1:k)';

    approxA = uint8(approxB);

    % plot images

    figure(1)
    subplot(2,4,i)
    imshow(approxA);
    title(sprintf('rank %d approximation',k))


end
```