# Math 170A: Homework 5

Merrick Qiu

## Q1

1. Note that $U^{-1} = U^T$ and $V^{-1} = V^T$ since these matricies are orthonormal and that $\Sigma^\dagger = (\Sigma^T\Sigma)^{-1}\Sigma^T$ since $\Sigma^T\Sigma\Sigma^\dagger = \Sigma^T$

$$\begin{aligned}
(A^TA)^{-1}A^T &= ((U\Sigma V^T)^TU\Sigma V^T)^{-1}(U\Sigma V^T)^T \\
&= (V\Sigma^TU^TU\Sigma V^T)^{-1}(V\Sigma^TU^T) \\
&= (V\Sigma^T\Sigma V^T)^{-1}(V\Sigma^TU^T) \\
&= (V(\Sigma^T\Sigma)^{-1}V^T)(V\Sigma^TU^T) \\
&= V(\Sigma^T\Sigma)^{-1}\Sigma^TU^T \\
&= V\Sigma^\dagger U^T \\
&= A^\dagger
\end{aligned}$$

2. 
$$A^\dagger A = (A^TA)^{-1}A^TA = I$$

3. 
$$\begin{aligned}
AA^\dagger &= U\Sigma V^T(V\Sigma^\dagger U^T) \\
&= U\Sigma\Sigma^\dagger U^T \\
&= U\begin{bmatrix} I_r & 0 \\ 0 & 0 \end{bmatrix}U^T \\
&= U_rU_r^T
\end{aligned}$$

## Q2

$||A||_2 = 3$, $||A^{-1}||_2 = 1$, $\kappa_2(A) = 3$, $||B||_2 = 4$.

$$B^\dagger = Q^T\begin{bmatrix} \frac{1}{4} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}P^T$$

## Q3

$$\begin{aligned} A^TA &= (U\Sigma V^T)^T U\Sigma V^T \\ &= V\Sigma^T U^T U \Sigma V^T \\ &= V(\Sigma^T\Sigma)V^T \end{aligned}$$

If $m = n$ and $A$ is full rank then

$$A^TA = V(\Sigma^T\Sigma)V^T = V\Sigma^2 V^T$$

Thus

$$||A^TA||_2 = \sigma_1^2 = ||A||_2^2$$

$$\kappa_2(A^TA) = \frac{\sigma_1^2}{\sigma_r^2} = \kappa_2(A)^2$$

## Q4

1. 1
2. $U_r$ is $3 \times 1$, $\Sigma_r$ is $1 \times 1$, $V_r^T$ is $1 \times 2$.
3. We can use the column vector as $U_r$, the row vector as $V_r^T$, and $\Sigma_r = ||U_r||||V_r^T||$

$$A = \begin{bmatrix} \frac{1}{\sqrt{6}} \\ \frac{2}{\sqrt{6}} \\ -\frac{1}{\sqrt{6}} \end{bmatrix} \begin{bmatrix} 2\sqrt{3} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$$

## Q5

```
% load image
A = imread('street2.jpg');
% convert to grascale
A = rgb2gray(A);
% B uses doubles instead of ints
B = double(A);

% print size of B
size(B)
% print rank of B
r = rank(B)
% compute SVD of B
[U,S,V] = svd(B);

% number of ranks for each approximation
ranks = [1 2 4 8 16 32 64 r];
% number of ranks to test
l = length(ranks);

% loop through each rank
for i = 1:l
    % get current rank
    k = ranks(i);
    % use the first k singular values for approximation
    approxB = U(:,1:k)*S(1:k,1:k)*V(:,1:k)';
    % round to nearest integer
    approxA = uint8(approxB);

    % make figure window
    figure(1)
    % go to the ith place in grid
    subplot(2,4,i)
    % show the approximated matrix
    imshow(approxA);
    % set title
    title(sprintf('rank %d approximation', k))
end
```

The code approximates a grayscale image using powers of 2 up to the actual rank of the image and then displays them in a plot. Using $k = 32$ is a way to compress the image, but we lose some detail.