Assignment 2

Lokesh Mohanty (SR no: 21014)

September 2022

Problem 1

(a) $(1.f)_2 \times 2^{e-1}$, where f is 3 bit and e is 2 bit (0, 1, 2)

f can vary from 000 to 111 (i.e., $2^3 = 8$ values)
$(e-1)$ can be -1, 0, 1 (i.e., e can be 0, 1, 2 $\rightarrow$ 3 values)

$\therefore$ Number of numbers this toy system can describe is $8 \times 3 = 24$

(b) Numbers Representation

Binary Scientific Notation	Binary Representation	Decimal Representation
$1.000 \times 2^{(0-1)}$	00000	0.5000
$1.001 \times 2^{(0-1)}$	00001	0.5625
$1.010 \times 2^{(0-1)}$	00010	0.6250
$1.011 \times 2^{(0-1)}$	00011	0.6875
$1.100 \times 2^{(0-1)}$	00100	0.7500
$1.101 \times 2^{(0-1)}$	00101	0.8125
$1.110 \times 2^{(0-1)}$	00110	0.8750
$1.111 \times 2^{(0-1)}$	00111	0.9375

Binary Scientific Notation	Binary Representation	Decimal Representation
$1.000 \times 2^{(1-1)}$	01000	1.000
$1.001 \times 2^{(1-1)}$	01001	1.125
$1.010 \times 2^{(1-1)}$	01010	1.250
$1.011 \times 2^{(1-1)}$	01011	1.375
$1.100 \times 2^{(1-1)}$	01100	1.500
$1.101 \times 2^{(1-1)}$	01101	1.625
$1.110 \times 2^{(1-1)}$	01110	1.750
$1.111 \times 2^{(1-1)}$	01111	1.875

Binary Scientific Notation	Binary Representation	Decimal Representation
$1.000 \times 2^{(2-1)}$	10000	2.00
$1.001 \times 2^{(2-1)}$	10001	2.25
$1.010 \times 2^{(2-1)}$	10010	2.50
$1.011 \times 2^{(2-1)}$	10011	2.75
$1.100 \times 2^{(2-1)}$	10100	3.00
$1.101 \times 2^{(2-1)}$	10101	3.25
$1.110 \times 2^{(2-1)}$	10110	3.50
$1.111 \times 2^{(2-1)}$	10111	3.75

(c) From the table above, we can see that the minimum real number we can store is 1.000×2^{-1} i.e., 0.5 and the maximum real number is 1.111×2^{1} i.e., 3.75.

(d) Gaps between numbers
We can see that,

$$\begin{aligned}
\text{for } e = 0, \qquad & gap = (0.001)_2 \times 2^{-1} && = (0.0001)_2 = 0.0625 \\
e = 1, \qquad & \quad = (0.001)_2 \times 2^{0} && = (0.001)_2 = 0.125 \\
e = 1, \qquad & \quad = (0.001)_2 \times 2^{1} && = (0.01)_2 = 0.25
\end{aligned}$$

$\therefore$ The gaps change with the magnitude of the number we are representing

(e) $\epsilon_{machine} \geqslant \frac{|x-x'|}{|x|}$, where $x \in \mathbb{R}$, $x' \in \mathbb{F}$
As we have seen above, the maximum gap is 0.25, which implies that the maximum value of $|x - x'|$ is 0.25/2 and the minimum value of $|x|$ to get that gap is mid of 2 and 2.25 i.e., 2.125

$$\therefore \epsilon_{machine} = \frac{0.125}{2.125} \approx 0.059$$

Problem 2

$\mathbf{Ax} = \mathbf{b}$, $\mathbf{A} \in \mathbb{R}^{n \times n}$ and $\mathbf{A}$ is invertible, $\mathbf{b} \in \mathbb{R}^n \implies \mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$

(a) Relative condition number $(\kappa(x)) = \max_{\delta x} \frac{\frac{||\delta f||}{||f||}}{\frac{||\delta x||}{||x||}}$

$$\begin{aligned}
\implies \kappa(b) &= \max_{\delta b} \frac{\frac{||\delta f||}{||f||}}{\frac{||\delta b||}{||b||}} = \max_{\delta \mathbf{b}} \frac{\frac{||\delta \mathbf{A}^{-1}\mathbf{b}||}{||\mathbf{A}^{-1}\mathbf{b}||}}{\frac{||\delta \mathbf{b}||}{||\mathbf{b}||}} \\
&= \max_{\delta \mathbf{b}} \frac{\frac{||\mathbf{A}^{-1}\delta \mathbf{b}||}{||\mathbf{A}^{-1}\mathbf{b}||}}{\frac{||\delta \mathbf{b}||}{||\mathbf{b}||}} \\
&= \max_{\delta \mathbf{b}} \frac{\frac{||\mathbf{A}^{-1}\delta \mathbf{b}||}{||\delta \mathbf{b}||}}{\frac{||\mathbf{A}^{-1}\mathbf{b}||}{||\mathbf{b}||}} = \max_{\delta \mathbf{b}} \left(\frac{||\mathbf{A}^{-1}\delta \mathbf{b}||}{||\delta \mathbf{b}||}\right)\left(\frac{||\mathbf{b}||}{||\mathbf{A}^{-1}\mathbf{b}||}\right)
\end{aligned}$$

From the definition of induced matrix norm, we know that

$$\begin{aligned}
||\mathbf{A}|| &= \max_{\mathbf{x}} \frac{||\mathbf{Ax}||}{||\mathbf{x}||} \\
\implies ||\mathbf{A}^{-1}|| &= \max_{\mathbf{x}} \frac{||\mathbf{A}^{-1}\mathbf{x}||}{||\mathbf{x}||} \\
\implies \kappa(\mathbf{b}) &= \frac{||\mathbf{A}^{-1}||||\mathbf{b}||}{||\mathbf{A}^{-1}\mathbf{b}||}
\end{aligned}$$

(b) Tight lower bound of $\kappa(\mathbf{b})$

From the definition of induced matrix norm, we know that

$$\begin{aligned}
||\mathbf{A}^{-1}|| &\geqslant \frac{||\mathbf{A}^{-1}\mathbf{b}||}{||\mathbf{b}||} \\
\implies \frac{||\mathbf{A}^{-1}||||\mathbf{b}||}{||\mathbf{A}^{-1}\mathbf{b}||} &\geqslant 1 \\
\implies \kappa(\mathbf{b}) &\geqslant 1
\end{aligned}$$

$\therefore$ Tight lower bond of relative condition number of $\mathbf{b}$ is 1

Problem 3

Given that, $K(\mathbf{A})||\Delta\mathbf{A}|| < ||\mathbf{A}||$ and $(\mathbf{A} + \Delta\mathbf{A})(\mathbf{x} + \Delta\mathbf{x}) = \mathbf{b} + \Delta\mathbf{b}$

$$\begin{aligned}
&\implies (\Delta\mathbf{A})\mathbf{x} + \mathbf{A}\Delta\mathbf{x} = \Delta\mathbf{b} \\
&\implies \mathbf{A}\Delta\mathbf{x} = \Delta\mathbf{b} - \Delta\mathbf{A}\mathbf{x} \\
&\implies \Delta\mathbf{x} = \mathbf{A}^{-1}(\Delta\mathbf{b} - \Delta\mathbf{A}\mathbf{x}) \\
&\implies ||\Delta\mathbf{x}|| = ||\mathbf{A}^{-1}(\Delta\mathbf{b} - \Delta\mathbf{A}\mathbf{x})|| \\
&\implies ||\Delta\mathbf{x}|| \leqslant ||\mathbf{A}^{-1}||(||\Delta\mathbf{b}|| + ||\Delta\mathbf{A}\mathbf{x}||)
\end{aligned}$$

We know that $K(\mathbf{A}) = ||\mathbf{A}||||\mathbf{A}^{-1}||$

$$\begin{aligned}
&\implies \quad ||\Delta\mathbf{x}|| \leqslant \frac{K(\mathbf{A})}{||\mathbf{A}||}(||\Delta\mathbf{b}|| + ||\Delta\mathbf{A}\mathbf{x}||) \\
&\implies \quad \frac{||\Delta\mathbf{x}||}{||\mathbf{x}||} \leqslant K(\mathbf{A})\left(\frac{||\Delta\mathbf{b}||}{||\mathbf{b}||} + \frac{||\Delta\mathbf{A}||}{||\mathbf{A}||}\right)
\end{aligned}$$

We know that $K(\mathbf{A})||\Delta\mathbf{A}|| < ||\mathbf{A}||$

$$\begin{aligned}
&\implies 1 - K(\mathbf{A})\frac{||\Delta\mathbf{A}||}{||\mathbf{A}||} \geqslant 0 \\
&\implies \frac{||\Delta\mathbf{x}||}{||\mathbf{x}||} \leqslant \frac{K(\mathbf{A})}{(1 - K(\mathbf{A})\frac{||\Delta\mathbf{A}||}{||\mathbf{A}||})}\left(\frac{||\Delta\mathbf{b}||}{||\mathbf{b}||} + \frac{||\Delta\mathbf{A}||}{||\mathbf{A}||}\right)
\end{aligned}$$

Problem 4

$fl(x) = x(1+\epsilon)$
$x \odot y = (x.y)(1+\epsilon)$

Stable: $\frac{||\tilde{f}(x) - f(\tilde{x})||}{||f(\tilde{x})||} = O(\epsilon_{machine})$
Backward Stable: $\tilde{f}(x) - f(\tilde{x}) = 0$ for some $\tilde{x}$

(a) $x \oplus x$

$$\begin{aligned}
\tilde{f}(x) &= fl(x) \oplus fl(x) = (x(1+\epsilon_1) + x(1+\epsilon_2))(1+\epsilon_3) \\
&= (2x + x(\epsilon_1 + \epsilon_2))(1+\epsilon_3) \\
&= 2x + x(\epsilon_1 + \epsilon_2 + 2\epsilon_3) \\
&= 2x(1 + 0.5\epsilon_1 + 0.5\epsilon_2 + \epsilon_3) \\
&= 2x(1+\epsilon_4) = 2\tilde{x} = f(\tilde{x}) \\
\implies \tilde{f}(x) &= f(\tilde{x})
\end{aligned}$$

Hence, it is both stable and backward stable.

(b) $x \otimes x$

$$\begin{aligned}
\tilde{f}(x) &= fl(x) \otimes fl(x) = (x(1+\epsilon_1) \times x(1+\epsilon_2))(1+\epsilon_3) \\
&= x^2(1+\epsilon_1)(1+\epsilon_2)(1+\epsilon_3) \\
&= x^2(1+\epsilon_1+\epsilon_2+\epsilon_3) \\
&= (x(1+\epsilon_4))^2 = \tilde{x}^2 = f(\tilde{x}) \\
\implies \tilde{f}(x) &= f(\tilde{x})
\end{aligned}$$

Hence, it is both stable and backward stable.

(c) $x \oslash x$

$$\begin{aligned}
\tilde{f}(x) &= fl(x) \oplus fl(x) = (x(1+\epsilon_1)/x(1+\epsilon_2))(1+\epsilon_3) \\
&= \frac{1+\epsilon_1}{1+\epsilon_2}(1+\epsilon_3) \\
&= (1+\epsilon_1)(1-\epsilon_2)(1+\epsilon_3) \\
&= 1+\epsilon_4 = f(\tilde{x})(1+O(\epsilon_{machine})) \neq f(\tilde{x})
\end{aligned}$$

$\therefore$ It is stable but not backward stable.

(d) SVD of $\mathbf{A}$: $\mathbf{A} = \mathbf{U\Sigma V}^T$
From Problem 5 we know that to compute the SVD of $\mathbf{A}$, we need to find $\mathbf{U}$, $\mathbf{\Sigma}$ and $\mathbf{V}$. And since $\mathbf{V}$ and $\mathbf{\Sigma}$ are the colleciton of eigen vectors and square of eigen values of $\mathbf{A}^T\mathbf{A}$ computing which is unstable.

For an algorithm which finds SVD to be stable/backward stable, we can assume that the input to the algorithm is the matrix $\mathbf{A}$ and the output is a vector with three elements $\mathbf{U}$, $\mathbf{\Sigma}$ and $\mathbf{V}$. And as per the definition of stablity and backward stability, we can deduce their stability/backward stability.

(e) $f(x) = \mathbf{x}^T\mathbf{y}$

$$\tilde{f}(x) = (fl(x_1) \otimes fl(y_1)) \oplus (fl(x_2) \otimes fl(y_2))...$$

We know that $\otimes$, $\oplus$ are backward stable

$$\begin{aligned}
\implies \tilde{f}(x) &= x_1y_1(1+\epsilon_1) \oplus x_2y_2(1+\epsilon_2) \oplus ... \\
&= (x_1y_1 + x_2y_2)(1+\epsilon_{n+1}) \oplus x_3y_3(1+\epsilon_2) \oplus ... \\
&\vdots \\
&= (x_1y_1 + x_2y_2 + \cdots + x_ny_n)(1+\epsilon_{2n-1}) \\
&= f(\tilde{x}) \\
\implies \tilde{f}(x) &= f(\tilde{x})
\end{aligned}$$

Hence, it is both stable and backward stable.

(f) Characteristic Polynomial: $det(\mathbf{A} - \lambda\mathbf{I}) = 0$

Assuming perturbed matrix $\mathbf{A}$

$$
\begin{aligned}
det(\mathbf{A} + \delta\mathbf{A} - \lambda\mathbf{I}) &= 0 \\
\implies \left| \begin{pmatrix} 1+\epsilon_1-\lambda & \epsilon_2 & \epsilon_3 \\ \epsilon_4 & 1+\epsilon_5-\lambda & \epsilon_6 \\ \epsilon_7 & \epsilon_8 & 1+\epsilon_9-\lambda \end{pmatrix} \right| &= 0 \\
\implies (1+\epsilon_1-\lambda)(\lambda^2+1+(1-\lambda)(\epsilon_5+\epsilon_9)-2\lambda) - \epsilon_2(1-\lambda)\epsilon_4 + \epsilon_3(\lambda-1)\epsilon_7 &= 0 \\
\implies -\lambda^3 + (1+\epsilon_1+2+\epsilon_5+\epsilon_9)\lambda^2 + (-3-2\epsilon_5-2\epsilon_9-2\epsilon_1)\lambda + (1+\epsilon_1+\epsilon_5+\epsilon_9) &= 0 \\
\implies \lambda^3 - \lambda^2(3+\epsilon_{10}) + \lambda(3+2\epsilon_{10}) - (1+\epsilon_{10}) &= 0
\end{aligned}
$$

If $\lambda_1, \lambda_2, \lambda_3$ are the roots of the above equation(eigen values), then we know that

$$
\begin{aligned}
\implies \lambda_1 + \lambda_2 + \lambda_3 &= 3 + \epsilon_{10} \\
\lambda_1\lambda_2 + \lambda_2\lambda_3 + \lambda_3\lambda_1 &= 3 + 2\epsilon_{10} \\
\lambda_1\lambda_2\lambda_3 &= 1 + \epsilon_{10} \\
\text{Let } \lambda_3 &= 3 + \epsilon_{10} - \lambda_1 - \lambda_2 \\
\implies \lambda_1\lambda_2(\lambda_1+\lambda_2) &= 1 + \epsilon_{10} \text{ and } \lambda_1\lambda_2 + (\lambda_1+\lambda_2)(3-\lambda_1-\lambda_2) = 3 + 2\epsilon_{10}
\end{aligned}
$$

We can see that this forms a quadratic equation with roots λ_1 and λ_2 for which $\frac{||\tilde{f}(x)-f(\tilde{x})||}{||f(\tilde{x})||} = O(\sqrt{\epsilon_{machine}}) \neq O(\epsilon_{machine})$.
Hence it is neither stable not backward stable.

Problem 5

$\mathbf{A} \in \mathbb{R}^{m \times n}$ and $rank(\mathbf{A}) = r$

(a) $\mathbf{G} = \mathbf{A}^T\mathbf{A}$

Prove $\mathbf{x}^T\mathbf{G}\mathbf{x} \geqslant 0 \ \forall \mathbf{x} \in \mathbb{R}^n$

$$\begin{aligned}
\mathbf{x}^T\mathbf{G}\mathbf{x} &= \mathbf{x}^T(\mathbf{A}^T\mathbf{A})\mathbf{x} \\
&= (\mathbf{A}\mathbf{x})^T\mathbf{A}\mathbf{x} \\
&= ||\mathbf{A}\mathbf{x}||^2 \geqslant 0 \\
\implies \mathbf{x}^T\mathbf{G}\mathbf{x} &\geqslant 0
\end{aligned}$$

Prove all eigen values of $\mathbf{G}$ are non-negative
We know that for any eigen value λ and the respective eigen vector $\mathbf{u}$ of $\mathbf{G}$

$$\begin{aligned}
&\mathbf{G}\mathbf{u} = \lambda\mathbf{u} \\
\implies &\mathbf{u}^T\mathbf{G}\mathbf{u} = \lambda\mathbf{u}^T\mathbf{u} = \lambda||\mathbf{u}||^2 \\
\implies &\frac{\mathbf{u}^T\mathbf{G}\mathbf{u}}{||\mathbf{u}||^2} = \lambda
\end{aligned}$$

Since $\mathbf{u}^T\mathbf{G}\mathbf{u} \geqslant 0$ and $||\mathbf{u}|| \geqslant 0$, the eigen value $\boxed{\lambda \geqslant 0}, \forall\lambda$

(b) **Prove A** and $\mathbf{A}^T\mathbf{A}$ have the same rank

Let $N(\mathbf{A})$ be the null space of $\mathbf{A}$ and $R(\mathbf{A})$ be the range of column space of $\mathbf{A}$

If $\mathbf{x} \in N(\mathbf{A}) \implies \mathbf{x} \in N(\mathbf{A}^T\mathbf{A})$ as $\mathbf{A}^T(0) = 0$. Hence $N(\mathbf{A}) \subseteq N(\mathbf{A}^T\mathbf{A})$

Let $\mathbf{x} \notin N(\mathbf{A})$ but $\mathbf{x} \in N(\mathbf{A}^T\mathbf{A}) \implies \mathbf{A}^T(\mathbf{A}\mathbf{x}) = 0$. Here $\mathbf{A}\mathbf{x} \in R(\mathbf{A}), \mathbf{A}\mathbf{x} \in N(\mathbf{A}^T)$, but we know that $N(\mathbf{A}^T)$ is orthogonal to $R(\mathbf{A})$. Hence $N(\mathbf{A}) = N(\mathbf{A}^T\mathbf{A})$

We know that (number of columns) = rank + (dimension of null space). And since the number of columns in $\mathbf{A}$ and $\mathbf{A}^T\mathbf{A}$ are same,

$$\begin{aligned}
rank(\mathbf{A}) + dim(N(\mathbf{A})) &= rank(\mathbf{A}^T\mathbf{A}) + dim(N(\mathbf{A}^T\mathbf{A})) \\
\implies rank(\mathbf{A}) &= rank(\mathbf{A}^T\mathbf{A})
\end{aligned}$$

(c) $\mathbf{v}$ and σ^2 form the eigen vector, eigen value pair of $\mathbf{A}^T\mathbf{A}$.

Prove that $\mathbf{u}$ is a unit eigen vector of $\mathbf{A}\mathbf{A}^T$ and is of the form $\mathbf{A}\mathbf{v}/\sigma$

Since $\mathbf{v}$ and σ^2 are the eigen vector, eigen value of $\mathbf{A}^T\mathbf{A}$,

$$
\begin{aligned}
\mathbf{A}^T\mathbf{A}\mathbf{v} &= \sigma^2\mathbf{v} \\
\implies \mathbf{A}\mathbf{A}^T\mathbf{A}\mathbf{v} &= \sigma^2\mathbf{A}\mathbf{v} \\
\implies \mathbf{A}\mathbf{A}^T\left(\frac{\mathbf{A}\mathbf{v}}{\sigma}\right) &= \sigma^2\left(\frac{\mathbf{A}\mathbf{v}}{\sigma}\right), \text{ as } \sigma \neq 0 \\
\implies \mathbf{A}\mathbf{A}^T\mathbf{u} &= \sigma^2\mathbf{u},\ \mathbf{u} = \mathbf{A}\mathbf{v}/\sigma
\end{aligned}
$$

Hence $\mathbf{u}$ is an eigen vector of $\mathbf{A}\mathbf{A}^T$ and $\mathbf{v}$ can be picked such that $\mathbf{u}$ is a unit eigen vector

(d) show that a full rank matrix $\mathbf{A}$ can be written as $\mathbf{A} = \mathbf{U}^T$

Let $\mathbf{v}_i$, σ_i^2 be eigen vector and eigen value pairs of the matrix $\mathbf{A}^T\mathbf{A}$

$$
\mathbf{A}^T\mathbf{A}\mathbf{v}_i = \sigma_i^2\mathbf{v}_i \tag{1}
$$

Since $\mathbf{A}^T\mathbf{A}$ is symmetric, all its eigen vectors are orthogonal and all eigen values are real. Hence all $\mathbf{v}_i$ are mutually orthogonal.

Let $\mathbf{u}_i = \mathbf{A}\mathbf{v}_i/\sigma_i$. This implies that $\mathbf{u}_i$ is a eigen vector of $\mathbf{A}\mathbf{A}^T$. And since $\mathbf{A}\mathbf{A}^T$ is also symmetric, all its eigen vectors are orthogonal to each other. Hence all $\mathbf{u}_i$ are mutually orthogonal.

$$
\begin{aligned}
\mathbf{A}^T\mathbf{A}\mathbf{v}_i &= \sigma_i^2\mathbf{v}_i \\
\mathbf{A}^T\mathbf{u}_i &= \sigma_i\mathbf{v}_i \\
\implies \mathbf{A}^T \begin{pmatrix} | & | & ... & | \\ \mathbf{u}_1 & \mathbf{u}_2 & ... & \mathbf{u}_n \\ | & | & ... & | \end{pmatrix} &= \begin{pmatrix} | & | & ... & | \\ \sigma_1\mathbf{v}_1 & \sigma_2\mathbf{v}_2 & ... & \sigma_n\mathbf{v}_n \\ | & | & ... & | \end{pmatrix} \\
\implies \mathbf{A}^T \begin{pmatrix} | & | & ... & | \\ \mathbf{u}_1 & \mathbf{u}_2 & ... & \mathbf{u}_n \\ | & | & ... & | \end{pmatrix} &= \begin{pmatrix} | & | & ... & | \\ \mathbf{v}_1 & \mathbf{v}_2 & ... & \mathbf{v}_n \\ | & | & ... & | \end{pmatrix} \begin{pmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \end{pmatrix} \\
\implies \left(\mathbf{A}^T \begin{pmatrix} | & | & ... & | \\ \mathbf{u}_1 & \mathbf{u}_2 & ... & \mathbf{u}_n \\ | & | & ... & | \end{pmatrix}\right)^T &= \left(\begin{pmatrix} | & | & ... & | \\ \mathbf{v}_1 & \mathbf{v}_2 & ... & \mathbf{v}_n \\ | & | & ... & | \end{pmatrix} \begin{pmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \end{pmatrix}\right)^T \\
\implies \begin{pmatrix} | & | & ... & | \\ \mathbf{u}_1 & \mathbf{u}_2 & ... & \mathbf{u}_n \\ | & | & ... & | \end{pmatrix}^T \mathbf{A} &= \begin{pmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \end{pmatrix} \begin{pmatrix} | & | & ... & | \\ \mathbf{v}_1 & \mathbf{v}_2 & ... & \mathbf{v}_n \\ | & | & ... & | \end{pmatrix}^T
\end{aligned}
$$

It can be written as $\mathbf{U}^T\mathbf{A} = \mathbf{\Sigma}\mathbf{V}^T$. And since $\mathbf{U}$ is orthogonal, its transpose is its inverse i.e., $\mathbf{U}^T = \mathbf{U}^{-1}$

$$
\begin{aligned}
&\implies \mathbf{U}^{-1}\mathbf{A} = \mathbf{\Sigma}\mathbf{V}^T \\
&\implies \mathbf{A}_{(m,n)} = \mathbf{U}_{(m,m)}\mathbf{\Sigma}_{(m,n)}\mathbf{V}^T_{(n,n)}
\end{aligned}
$$

Problem 6

(a) Based on observation, it seemed that with the initial 80 singular values, most of the information is contained. Hence 80 singular values are required.

(b) Based on observation in (a), we need to send the first 80 rows of $\mathbf{U}$, the first 80 elements of $\mathbf{\Sigma}$ and the first 80 columns of $\mathbf{V}$ for red, blue and green. These can be later used to reconstruct the approximate image.

(c) Empirically we found that $||\mathbf{A} - \mathbf{A}_v|| \approx \sigma_{v+1}$ and $||\mathbf{A} - \mathbf{A}_v||_F = \sqrt{\sigma_{v+1}^2 + ... + \sigma_r^2}$. I calculated this separately for the red, blue and green matrices.

Norm	Red	Blue	Green
L2	2269.045423718001	2036.66041654077	1928.180396628874
Frobenius	19333.317340145557	18587.417927015093	17127.025982851294

Code(in python):

```
import matplotlib.pyplot as plt
from PIL import Image
import numpy as np
from numpy.linalg import svd, norm

class ProcessImage:
    def __init__(self, file):
        self.img = np.array(Image.open(file), dtype='uint')
        self.red = svd(self.img.transpose()[0], False)
        self.green = svd(self.img.transpose()[2], False)
        self.blue = svd(self.img.transpose()[1], False)

    def compress(self, r):
        self.r = r;
        self.red_reduced = np.delete(
            self.red[0],
            slice(r-1, -1),
            axis=1
        ).dot(np.diag(self.red[1][0:r]).dot(self.red[2][0:r]))
        self.green_reduced = np.delete(
            self.green[0],
            slice(r-1, -1),
            axis=1
```

```
    ).dot(np.diag(self.green[1][0:r]).dot(self.green[2][0:r]))
    self.blue_reduced = np.delete(
        self.blue[0],
        slice(r-1, -1),
        axis=1
    ).dot(np.diag(self.blue[1][0:r]).dot(self.blue[2][0:r]))

    self.img_reduced = np.array(
        [self.red_reduced, self.green_reduced, self.blue_reduced],
        dtype='uint'
    ).transpose()
    return self

def compare(self):
    fig, (ax1, ax2) = plt.subplots(1,2)
    ax1.imshow(self.img)
    ax2.imshow(self.img_reduced)
    plt.show()
    return self

def error(self):
    l2_norm_red = norm(np.subtract(self.red, self.red_reduced), 2)
    l2_norm_green = norm(np.subtract(self.green, self.green_reduced), 2)
    l2_norm_blue = norm(np.subtract(self.blue, self.blue_reduced), 2)
    fro_norm_red = norm(np.subtract(self.red, self.red_reduced))
    fro_norm_green = norm(np.subtract(self.green, self.green_reduced))
    fro_norm_blue = norm(np.subtract(self.blue, self.blue_reduced))

    l2_red = self.red[1][self.r+1]
    l2_green = self.green[1][self.r+1]
    l2_blue = self.blue[1][self.r+1]

    fro_red = norm(self.red[1][self.r+1:])
    fro_green = norm(self.green[1][self.r+1:])
    fro_blue = norm(self.blue[1][self.r+1:])

    return [
        [l2_norm_red, l2_norm_green, l2_norm_blue],
        [fro_norm_red, fro_norm_green, fro_norm_blue],
        [l2_red, l2_green, l2_blue],
```

```
            [fro_red, fro_green, fro_blue],
        ]

# ProcessImage("Webb's_First_Deep_Field.png").compress(80).compare()
compressed = ProcessImage("Webb's_First_Deep_Field.png")
compressed.compress(80).compare()
print(compressed.error())
```