Part I:

Question 1 (40 points)

Compute the LU decomposition of the following matrix by hand by setting the diagonal entries of the matrix L to 1. Make sure to show as many steps are possible.

$$\begin{bmatrix} 8 & 5 \\ 10 & 5 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ l_{21} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} \\ 0 & u_{22} \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} \\ l_{21}u_{11} & l_{21}u_{12} + u_{22} \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 \\ 5/4 & 1 \end{bmatrix} \begin{bmatrix} 5 & 5 \\ 0 & -5/4 \end{bmatrix}$$

Work for the Matrix LU decomposition

$u_{11} = 8$, $u_{12} = 5$, $l_{21}u_{11} = 10$, $l_{21}u_{12} + u_{22} = 5$,
$8l_{21} = 10 = l_{21} = 10/8 = 5/4$,
$5/4(5) + u_{22} = 5 = 25/4 + u_{22} = 5 = u_{22} = 5 - 25/4 = -5/4$

Question 2

$$\begin{bmatrix} 10 & -7 & 0 \\ -3 & 2 & 6 \\ 5 & -1 & 5 \end{bmatrix}$$

Question 2.1

Compute the LU decomposition of the matrix above and write down your solution. It is possible that obtain a permutation matrix, P, from the result; if so, include it in your answer.

Permutation Matrix

[[1. 0. 0.]

[0. 0. 1.]

[0. 1. 0.]]

Question 2.2

Compute the SVD of the matrix above and write down the singular values.

Singular values = [13.57864052 7.90802895 1.44346799]

Question 2.3

Compute the QR decomposition and write down the Q and R matrices.

Q

```
[[-0.86386843  0.42560398 -0.26943013]
 [ 0.25916053 -0.08312578 -0.96225045]
 [-0.43193421 -0.90108343 -0.03849002]]
```

R

```
[[-11.5758369    6.99733425  -0.6047079 ]
 [  0.          -2.24439601  -5.00417184]
 [  0.           0.          -5.96595278]]
```

Question 3

Computing Singular Values

Goal: The goal of this assignment is to compute the singular values of the given matrix yourself and compare those results with what numpy’s function computes.

Build in library function

Σ=

[3.06028845 1.07904872 0.40852445 0.14168031 0.05053199]

my version of the singular

Σ=

[3.06028845 1.07904872 0.40852445 0.14168031 0.05053199]

My version from singular value was more complex than the built-in library. My version was for use in my columns U. In my transposition multiplication, the final two rows were swapped, as were the columns V. And to get the diagonal matrix knows as Singular Value Matrix you need to the square root of the eigenvalues of AAT or ATT. The Singular Value Matrix I computed was correct, but it did not sort appropriately. I had to sort to make sure the values aligned correctly when compared to the SVD library.

CODE IMPLEMENTION:

```
Garcia_Milord_Q2_1

A = np.matrix([[10, -7, 0], [-3, 2, 6], [5, -1, 5]])

def Q2pt1():
    P, L, U = scipy.linalg.lu(A)
    print("Lower Matrix\n" +str(L))
    print("Upper Matrix\n" +str(U))
    print('Permutation Maxtrix\n'+str(P))
Q2pt1()
```

```
Garcia_Milord_Q2_2

def Garcia_Milord_Q2_P2():
    u, s, vt = np.linalg.svd(A)
    return print('Singular values\n '+ str(s))
Garcia_Milord_Q2_P2()
```

```
Garcia_Milord_Q2_3

def Garcia_Milord_Q2_P3():
    Q, R = np.linalg.qr(A)
    print('Q')
    print(Q)
    print('R')
    print(R)

Garcia_Milord_Q2_P3()
```

```
Garcia_Milord_Q3
    # SVD_DA_1.JSO
with open('SVD_DA_1.JSO') as f:
    d = json.load(f)


print('Build in library function')
u, s, vt = np.linalg.svd(d)
print("Σ=\n "+str(s))
print()
print('U=\n'+str(u))
print()
print('Vt=\n'+str(vt))
print()

print("My function for the SVD")
A = np.array(d)
#line 37-42 doing the code swap column thing
'''The eigenvectors of AAT are the columns of U'''
AAT= np.linalg.eig( A @ A.T)[1]
'''The eigenvectors of ATA are the columns of V'''
ATA = np.linalg.eig(A.T @ A)[1]
'''The square root of the eigenvalues of AAT , or ATA, make up the diagonal
matrix Σ'''
Sigma = np.sqrt(np.linalg.eig(A.T @ A)[0])

#Singular values was to sort my last work row
# and know there is better way
Sigma[[4,3]] = Sigma[[3,4]]
print('U=\n'+ str(AAT))
print()
print('V=\n'+ str(ATA))
print()
print('Σ=\n'+ str(Sigma))
```