COE 352 Project 1

This file and the README.md contain all the answers to questions asked by Dr. Trahan in the Project 1 Assignment. I recommend the internet view of the GitHub page as the README is formatted in a nicer way than this.

Comparison of SVD results:

The following array was used for the comparison:

```
A = np.array([[5, 2, 3],
              [4, 5, 6],
              [7, 8, 9],])
```

The following is output from the ipynb provided in the GitHub. This compares my solver (Custom) versus the Numpy results.

Custom U:

[[0.32717864 0.93556421 -0.13294265]
 [0.50256733 -0.29141461 -0.81394324]
 [0.8002376 -0.19949221 0.56552864]]

NumPy U:

[[-0.32717864 0.93556421 -0.13294265]
 [-0.50256733 -0.29141461 -0.81394324]
 [-0.8002376 -0.19949221 0.56552864]]

Custom Vt

[[0.5317045   0.55017589 0.64389193]
 [0.83522779 -0.46657511 -0.29103642]
 [0.14030273 0.69254181 -0.70760228]]

NumPy Vt

[[-0.5317045 -0.55017589 -0.64389193]
 [0.83522779 -0.46657511 -0.29103642]
 [0.14030273 0.69254181 -0.70760228]]

Custom S (Sigma):

[[17.39279188 0.   0.   ]
 [0.   2.53310192 0.   ]
 [0.   0.   0.27237001]]

NumPy S (Sigma):
[17.39279188 2.53310192 0.27237001]

Custom Reconstructed Matrix (U @ S @ Vt):
[[5. 2. 3.]
 [4. 5. 6.]
 [7. 8. 9.]]

Numpy Matrix A reconstructed
[[5. 2. 3.]
 [4. 5. 6.]
 [7. 8. 9.]]

As we can see on the page above, not every value of every matrix is exactly the same. However, this does not mean that either solution is incorrect. There is simply more than one functional SVD decomposition for a valid matrix.

It is clear at the end that both solutions work as the reconstructed matrices are identical to the given A matrix.

Additionally the inverse matrix and condition number of my custom solver versus Numpy's solution matched up as well.

Custom SVD solver condition number and inverse matrix:
63.85722159418202
[[0.25   -0.5   0.25   ]
 [-0.5   -2.   1.5   ]
 [0.25   2.16666667 -1.41666667]]

Numpy condition number and inverse matrix:
63.85722159418491
[[0.25   -0.5   0.25   ]
 [-0.5   -2.   1.5   ]
 [0.25   2.16666667 -1.41666667]]

Two Free End Examination:

As discussed in lecture 8, having two free ends of such a system does not make physical sense. This is seen in the example where you have 3 masses and 2 springs in a free-free system. This creates a stiffness matrix which is a 3x3 with a rank of two. This means that it is not invertible and therefore is not solvable. The issue stems from having more unknowns than equations. This means that there will not be a unique solution to the system. The stiffness matrix must have full rank (in this case rank of 3) in order to be solvable. This is only achievable with one of the boundary conditions that I have implemented in the SVD algorithm.