Lab 3 Write-up:

Please see accompanying files for code and generated charts/images

1) 1. **Vector in S:** [0.16065981 0.67354565 0.48197942 0.99486526]

Vector not in S: [0.67081582 0.2428012   0.78650936 0.01165721]

You can check if a new vector **v** is in S by creating an augmented matrix composed of all the column vectors in a basis for S on the left and **v** on the right. After row reducing the augmented matrix, if the right side is all 0's, then **v** is in S; otherwise, **v** is not in S.

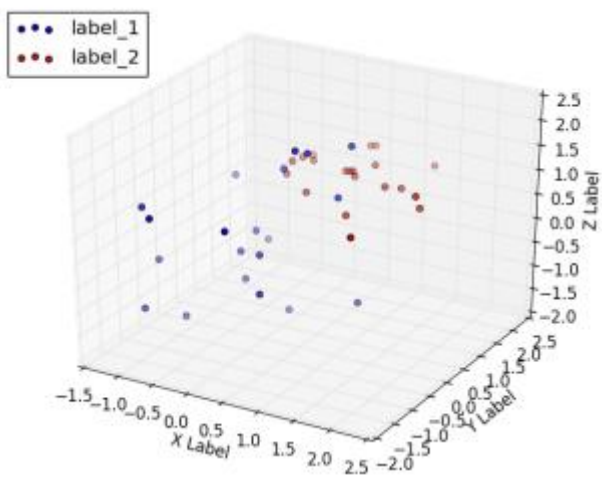
2. dim(S) = 2

3. **Orthonormal basis for S:**

[[-0.24011927 -0.05990306 -0.35992538 -0.89955994]
 [0.8581727 -0.29094143 0.27628983 -0.32024463]]

4. For all vectors x in S, x takes the form Beta * v0. For a randomly generated vector v0 in S, where v 0 = [0.14376971 0.70545992 0.43130913 0.99299934], Beta = 0.085049764085389987, which we calculated using the equation we derived in class for a generic vector in a particular subspace using t he orthogonality principle. (See the bottom of problem1.py)

2) 1. See attached plot (problem2a.png):



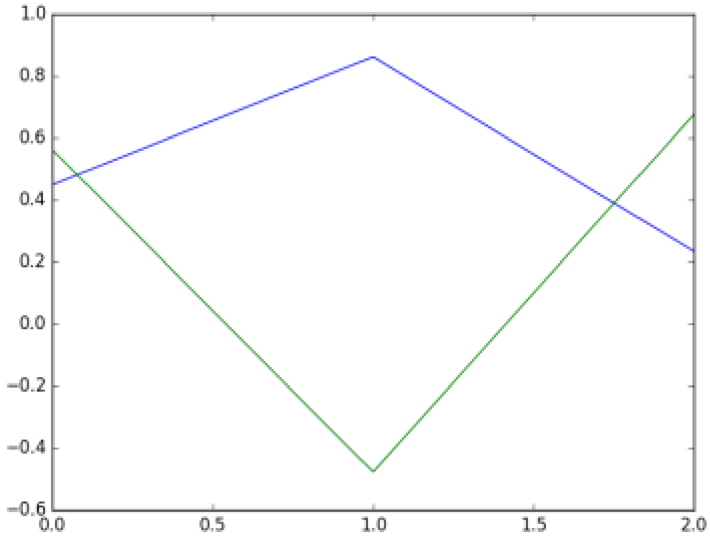
2. The points from each of the two data sets are fairly close together, appearing to almost form two distinct clusters.

3. For one particular run of the procedure for this problem, the covariance matrix we calculated was:

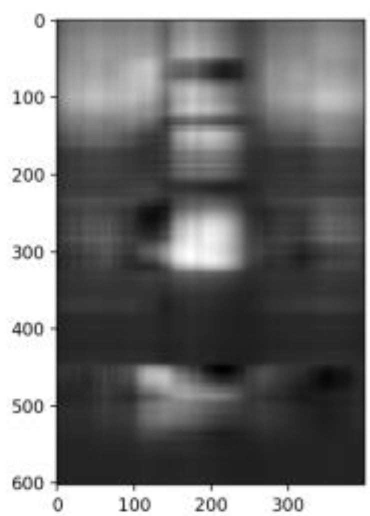
[[0.86561594203446424, 0.44395345359304211, 0.39180096534572695],
[0.44395345359304211, 1.0986407996325145, 0.35471541995899974],
[0.39180096534572695, 0.35471541995899974, 0.50224413971835202]],

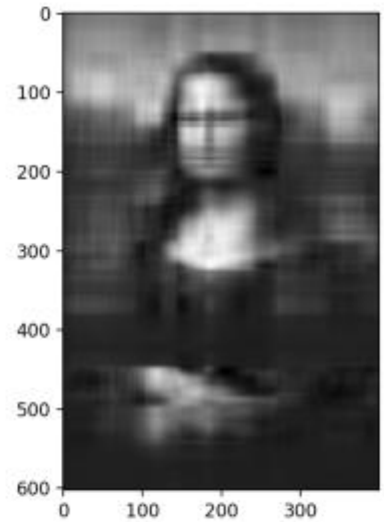
which was consistent with the result of np.cov() for the same procedure. For verification, run another trial using the attached code (problem2.py).

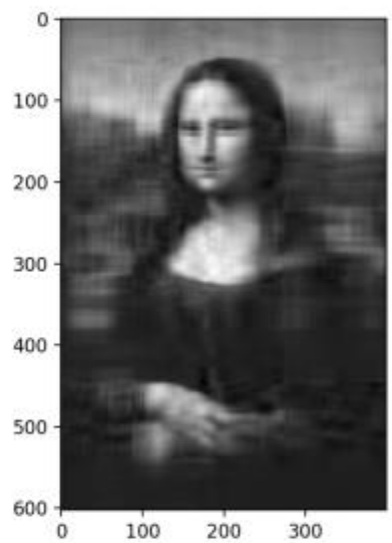
4. Yes, performing PCA made the two labels more distinguishable in two dimensions. In the resulting plot below (problem2d.png in zip file), the two labels form two distinct curves with opposing “concavities.”



3) 1. See attached images below, named mona_lisa_k.png in the zip file, where k is the corresponding value of k.







2. Given that the original image is 603 pixels tall and 400 pixels wide, the entire image is represented in approximately 241200 pixels. If each pixel is represented by two bytes, then the entire compressed images are about **3,859,200 bits.**

4) 1. Usernames -
Shamma: ShammaKabir
Tony: adepalatis

2. **RMSE:** 0.13029

3. **Best RMSE:** 0.12242. We got this by raising the value of alpha to increasingly higher values, with the best results being at alpha = 7.5. We ran out of submissions before we could go any higher, but we suspect we could have further improved if we tried different solvers and (probably to a lesser degree) if we fiddled with the tolerance.