Homework 1: Linear Algebra, Images, & Operators

1. Entire images can be manipulated as vectors. You are given examples of T1-weighted brain slices from the FastMRI dataset (please see the course website). Use an inner product to compute the angle between each possible pairing of those images and comment on the relation between the angle and visual similarity. *Hint*: load the images into a dictionary or NumPy array and have code do the pairing for you.

2. Convince yourself that operators can "do" imaging. You are given matrix operator H as a compressed NumPy file (please see the course website). Pretend that the images from the previous exercise are real objects and apply the H operator to them. *Hint*: print the shape and dtype of H after you load the file to make sure H things are reasonable.

a. Describe in words what the operator does to the objects.

b. Characterize this imaging system. E.g., $H : \mathbb{U} \to \mathbb{V}$ where you fill-in $\mathbb{U}$ and $\mathbb{V}$.

c. Answer in words: do you think that H has a null space? Why or why not?

3. The expression of an operator depends on the basis. Given is a linear transform $T : \mathbb{C}^2 \to \mathbb{C}^3$,

$$T(x, y) = (x + iy, x - iy, 2x) \tag{1}$$

and new basis sets for the domain and range, respectively: $\mathcal{B}_1 = \{(3, 4), (4, -3)\}$ and $\mathcal{B}_2 = \{(1, 2, 2), (6, 3, -6), (6, -6, 3)\}$

a. Compute a matrix S_1 that transforms standard basis vectors into $\mathcal{B}_1$ vectors. Compute another matrix S_2 which transforms standard basis vectors into $\mathcal{B}_2$ vectors. *Hint*: don't forget to normalize.

b. Compute the matrix operator T in terms of the standard bases. Use S_1 and S_2 to compute a new matrix operator T_{new} in terms of $\mathcal{B}_1$ and $\mathcal{B}_2$.

c. Given $u = (1, i)$ written in terms of $\mathcal{B}_1$, use T_{new} to transform u into a $v \in \mathbb{C}^3$.

d. Write u and v in terms of the standard bases and verify that $T(u_e) = v_e$.

4. An important basis set (that will be used in the future). Given all functions of the form

$$f(N) = \sum_{n=0}^{N-1} a_n \sin(n\pi x) + b_n \cos(n\pi x) \tag{2}$$

where $n \in \mathbb{Z}$ and a_n, b_n are constants, show that $\{u_n = \frac{1}{\sqrt{2}} \exp(in\pi x)\}$ is an orthonormal basis of the space of all $f(N)$ where the inner product is defined as

$$\langle f, g \rangle = \int_{-1}^{1} \overline{f(x)} g(x) dx \tag{3}$$

and compute the dimension of this space. *Hint*: write trigonometric functions in terms of complex exponentials and consider that you change the order and index of summation whenever you like.

5. The derivative operator (D). For the vector space in the previous exercise, show that the operator $A = iD$ is Hermitian. *Hint*: Think about the Hermitian condition terms of an inner product $\langle f, g\rangle$ and apply A to a generic f without writing everything out. Then show that the pesky term "preventing" A from being Hermitian vanishes in this particular space.

6. Characterize an imaging system by the forward operator. Show (trivially) that the the idealized imaging system:

$$g(x', y') = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \operatorname{sinc}(x' - x)\operatorname{sinc}(y' - y) f(x, y) dx dy \tag{4}$$

is a linear operator. Then compute the adjoint of this operator and thus determine whether it is Hermitian. What are the domain and range of this operator?

7. The nullity of a Toeplitz matrix operator. The operator in exercise 2 is a kind of Toeplitz matrix. To keep to computations manageable, assume an image size of 64 x 64 pixels and compute a Toeplitz matrix operator $H : \mathbb{R}^{4096} \rightarrow \mathbb{R}^{4096}$ with elements given by:

$$H(\sigma)_{r,c} = \exp(-\frac{(r - c)^2}{\sigma}) \tag{5}$$

a. Let $\sigma = 8$ and plot your operator as a heatmap so that you see what it looks like.

b. Use numpy.linalg.matrix_rank(H) to compute the rank of H and then use that to compute the nullity of H. *Note*: there are more elegant and more informative ways to do this but, for now, it's ok to just let NumPy handle the calculation.

c. Repeat part b for $\sigma \in \{8, 16, 32, 64, 128, 256\}$ and make a plot of nullity versus σ. Comment in words what's going on here and why this makes sense in the context of forward imaging operators and the key subspaces discussed in class.

Note: this exercise can be done straightforwardly with nested loops, however, that code could take tens of minutes to run, per operator, on an old desktop computer. If you already know how to speed this up via vectorized NumPy, Numba and/or parallelization, that's great. If not, don't spend hours learning now; just let the slow code run and consider starting on 8x8 test objects while you're debugging. FYI: this sort of thing is covered in BIOE 488.

8. Projection operators and subspaces in imaging. For this exercise, let the forward operator be $H(\sigma = 32)$ from the previous exercise.

a. Use scipy.linalg.orth(H) to orthonormalize $H(32)$. Call this new operator A and convince yourself the columns of A indeed are orthogonal (to with some reasonable numerical error). *Note*: One could orthonormalize via the Gram-Schmidt algorithm, however, SciPy employs some fast SVD trickery to do this for us. We'll explore how that works in the next assignment.

b. Compute the projection operator $P = A(A^T A)^{-1}A^T$ and verify that $\sum_r \sum_c P_{r,c}$ is what you expect.

c. Compute the null space projection operator, P_{null}, by using a very simple identity.

d. Apply P and P_{null} (separately) to the test object from the course website and display the results side-by-side. Describe, in the language of imaging science, what each result "is."

e. Compute the angle between the results. How can you *know* that your answer is correct?

BONUS: why is orthonormalization important and what would happen if we didn't do part a?