Homework 5

Please answer the following questions in complete sentences in a clearly prepared manuscript and submit the solution by the due date on Gradescope, around 4am on Monday November 22nd.

Remember that this is a graduate class. There may be elements of the problem statements that require you to fill in appropriate assumptions. You are also responsible for determining what evidence to include. An answer alone is rarely sufficient, but neither is an overly verbose description required. Use your judgment to focus your discussion on the most interesting pieces. The answer to “should I include ‘something’ in my solution?” will almost always be: Yes, if you think it helps support your answer.

Problem 0: Homework checklist

- Please identify anyone, whether or not they are in the class, with whom you discussed your homework. This problem is worth 1 point, but on a multiplicative scale.
- Make sure you have included your source-code and prepared your solution according to the most recent Piazza note on homework submissions.

Problem 1: Updating factorizations of linear systems of equations

In class, we showed how to solve \((A + uv^T)x = b\) when given a fast factorization method to solve \(Ax = b\). In this problem, we will address the question of how to update the factorization itself. Suppose that we are given a Cholesky factorization of \(A = LDL^T\) as we saw in class. Show how to update this factorization to produce \(L'D'L'^T = (A + uu^T)\). Your algorithm should do no more than \(O(n^2)\) work.

Problem 2: Trig identities with matrices.

This question asks you to understand how one might generalize a mathematical result and test a conjecture. In this case, the result is known, so feel free to look it up and justify it that way; however, see the ‘skeptical note’ below. Alternatively, it’s fairly simple to get it yourself. So feel free to with that route too. (You’ll learn more.)

The idea: A well known relationship is \(\cos^2 \theta + \sin^2 \theta = 1\). Since we can define cos and sin for matrices as well, shouldn’t we have:

\[
\cos^2 A + \sin^2 A = I
\]

or perhaps

\[
\cos^2 A + \sin^2 A = \text{ones}
\]

where \(A\) is a square matrix?

1. Prove or disprove one of these conjectures. Also, show numerical evidence on at least a 10x10 orthogonal matrix that your result is correct. (In this case,
you are presenting to someone who has taken this class and has seen how many “proof ideas” fall on their face with simple numerical tests because the “proof” was really wrong.)

2. Prove that \( \tan A = (\cos A)^{-1} \sin A = \sin A (\cos(A))^{-1} \).

**Problem 3: Weighted orthogonality for SVD.**

One possible weighted generalization of the SVD involves producing a factorization

\[
A = U \Sigma V^T
\]

where \( U^T R U = I \) and \( V^T S V = I \), where \( R \) and \( S \) are symmetric positive definite matrices.

A common theme in solving more advanced problems is converting or translating them into problems that we know how to solve.

Show how to use a standard SVD computation to produce this weighted SVD factorization.

**Problem 4: Implementing a rank-k solution update.**

Given \( x \) and an LU factorization of \( A \), then updating the solution of a linear system \( Ax = b \) to a new solution \( (A + UV^T)y = b \) can be done fairly efficiently.

Write code or fairly detailed pseudo-code to do this. This uses the Julia factorization \( F \) which enables you to solve systems with \( A \) without recomputing the factorization.

```julia
function update(x::Vector, b::Vector, F::LUFact, U::Matrix, V::Matrix)
end
```

**Problem 5: Adding and deleting an equation**

Recall that a linear system represents the simultaneously solution of a set of linear equations

\[
a_1^T x = b_1 \\
a_2^T x = b_2 \\
\vdots \\
a_n^T x = b_n.
\]

Consider a set of \( n \) equations and \( n \) unknowns with a unique solution for any possible set of values \( b_1, \ldots, b_n \).

Let \( \text{alg}(b) \) be an algorithm to solve for \( x \) when given \( b_1, \ldots, b_n \).

Let \( \text{alg}(b) \) be an algorithm to solve for \( x \) when given \( b_1, \ldots, b_n \).

Show how to use \( \text{alg} \) in order to solve for \( y \) in the system of equations

\[
c_1^T y = c_1 \\
a_2^T y = b_2 \\
\vdots \\
a_n^T y = b_n.
\]

That is, we deleted the equation with \( a_1 \) and added a new equation with \( c_1 \) instead. If you use \( \text{alg} \) too many times, you may lose points. Be efficient!

(Hint: this is essentially a special case of other problems on this homework or other things we’ve seen in class.)