CSx80 Matrix Algorithms for Data Science Fall 2022

Johnstone **Homework 1** (due Monday, 09.12.22, 8pm)

Reading:

Trefethen Lecture 1 (Matrix-Vector Mult), 2 (Orthogonal), 3 (Norms), 12 (Condition), 13 (Floating Point).

Solomon Chapters 1 (Mathematics Review) and 2 (Numerics and Error Analysis).

Also review your **undergraduate linear algebra text**.

Honor code: Study groups are allowed. Students may discuss homework problems at a high algorithmic level with other members of this class, but not write anything down. Each student must write down the solutions alone and independently. In other words, each student must understand the solution well enough to reconstruct it by him/herself. In addition, each student should write on the problem set the set of people with whom s/he collaborated. It is not legal to search the web for solutions. Because we occasionally reuse problem sets from previous years, we also expect students not to copy, refer to, or look at solutions from previous years in preparing their answers. It would be an honour code violation to either search the web, to use solutions developed by others, or to refer to previous year solutions. This is also counterproductive, since you will not learn for the exams, which have the bulk of the weight in this course.

I declare that I have completed this assigment in accordance with the UAB Academic Integrity Code. I have read the UAB Academic Integrity Code and understand that any breach of the Code may result in severe penalties.

Your signature: Adrian Hilton

Date: 9 Sep 2022

1. Leveraging and preserving structure is important in matrix algorithms. Upper triangular matrices are an important special matrix with added structure. A square or rectangular matrix (a_{ij}) is upper triangular if $a_{ij} = 0$ whenever $i > j$. That is, all the elements below the diagonal are zero.

   Show that the product of two upper triangular matrices is upper triangular. Only a concise proof will earn full grades. (Aim for tight proofs, as you aim for tight code.)

2. 1.1, Trefethen

3. Give an $O(n^2)$ algorithm for computing $C = (xy^t)^k$, where x and y are n-vectors.

4. Positive definite matrices are important in convex optimization (a central topic for ML), since a twice-differentiable function $f : \Re^n \to \Re$ is convex near p if its Hessian matrix is positive definite.

   A matrix A is positive definite (resp., semidefinite) if it is symmetric and $x^t Ax > 0$ (resp., $x^t Ax \geq 0$) for all nonzero $x \in \Re^n$ (resp., all $x \in \Re^n$).

   (a) **Show that** zz^t **is positive semi-definite**, where $z \in \Re^n$ is an n-vector.

   (b) Let $A \in \Re^{n\times n}$ be positive semidefinite. Let $B \in \Re^{m\times n}$ be arbitrary. **Is** BAB^t **positive semidefinite?** If so, prove it. If not, give a specific counterexample.

   (c) (780 only) Let $A = zz^t$, where $z \in \Re^n$ is a *nonzero* n-vector. **What is the rank of** A (and why)? **Characterize the null space of** A concisely and intuitively (and explain why it has this form).

5. (680 only) Vectors will be a standard data structure in this class, and their size will be an important measurement, often representing error. The Pythagorean Theorem states that the Euclidean length of a 2-vector (a, b) is $\sqrt{a^2 + b^2}$ (although it is traditionally posed less helpfully: that the side lengths of a right triangle in 2-space satisfy $a^2 + b^2 = c^2$, where c is the hypotenuse). Prove that the generalization to 3-space holds. That is, **prove that the length of a 3-vector** (a, b, c) **is** $\sqrt{a^2 + b^2 + c^2}$**.** (Hint: reduce to the 2-space version of Pythagoras.)

6. Solomon 1.7 (AA^t and A^tA in terms of rows/columns)

7. Solomon 2.7a (error)

8. (780 only) Solomon 2.12 a,b (robust geometric primitives)