Week 3: Wednesday, Jan 9

Problem du jour

1. As a function of matrix dimension, what is the asymptotic complexity of computing a determinant using the Laplace expansion (cofactor expansion) that you probably learned about when you first heard of determinants?

2. Suppose `x` is a $2 \times n$ MATLAB array in which each column represents a point in $[0, 1]^2$. How would I write efficient MATLAB code to find the minimum pairwise distance between these point?

   Note: I'm thinking of a naive $O(n^2)$ algorithm, not something based on quadtrees or similar data structures. By "efficient" I mean "making efficient use of MATLAB vector operations."

Logistics

1. HW 1 solutions are posted on CMS. Ivo is currently grading them. Remember that we do "grading by sampling" – this time, it will be 2.5 points each for problems 1 and 3.

2. When we finish grading, the scores will be posted to CMS. Homework available for pickup in 363C Upson.

3. We've been spending a lot of time talking about error analysis. It's important to understand this stuff, because while MATLAB gives you fast, robust linear solvers that you're not likely to improve upon (for problems without special structure, at least), you can get into a lot of trouble using those solvers if you don't understand when they are sensitive to errors.

   But don't worry, we'll move on soon! Monday we will talk about algorithms. The plan is still to talk about solving linear systems and least squares problem before the first prelim (in about two weeks).

4. Next week's homework will be a project involving setting up and solving a linear system.

Of sizes and sensitivities

When we talked about rounding error analysis for scalar functions, I emphasized a couple points:

- I care about *relative* error more than *absolute* error.
- Some functions are *ill-conditioned*. These are hard to solve because a small relative error in the input can lead to a large relative error in the output.
- We like algorithms that are *backward stable*: that is, they return an answer which solves slightly the wrong problem. (small *backward error*). The *forward error* for a backward stable algorithm is bounded by the product of the backward error and the condition number, so a backward stable algorithm can return results with large forward error when the problem is ill-conditioned.

Now I want to build the same machinery for talking about problems in which the inputs and outputs are vectors.

Norm!

First, we need the concept of a *norm*, which is a measure of the length of a vector. A norm is a function from a vector space into the real numbers with three properties

1. Positive definiteness: $\|x\| > 0$ when $x \neq 0$ and $\|0\| = 0$.
2. Homogeneity: $\|\alpha x\| = |\alpha| \|x\|$.
3. Triangle inequality: $\|x + y\| \leq \|x\| + \|y\|$.

One of the most popular norms is the Euclidean norm (or 2-norm):

$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2} = \sqrt{x^T x}.$$

We will also use the 1-norm and the ∞-norm (*a.k.a.* the max norm or the Manhattan norm):

$$\|x\|_1 = \sum_i |x_i|.$$
$$\|x\|_\infty = \max_i |x_i|$$

Second, we need a way to relate the norm of an input to the norm of an output. We do this with matrix norms. Matrices of a given size form a vector space, so in one way a matrix norm is just another type of vector norm. However, the most useful matrix norms are *consistent* with vector norms on their domain and range spaces, i.e. for all vectors x in the domain,

$$\|Ax\| \leq \|A\|\|x\|.$$

Given norms for vector spaces, a commonly-used consistent norm is the *induced* norm (operator norm):

$$\|A\| \equiv \max_{x \neq 0} \frac{\|Ax\|}{\|x\|} = \max_{\|x\|=1} \|Ax\|.$$

Question: Why is the second equality true?

The matrix 1-norm and the matrix ∞-norm (the norms induced by the vector 1-norm and vector ∞-norm) are:

$$\|A\|_1 = \max_j \left(\sum_i |a_{ij}| \right) \quad \text{(max abs column sum)}$$
$$\|A\|_\infty = \max_j \left(\sum_i |a_{ij}| \right) \quad \text{(max abs row sum)}$$

If we think of a vector as a special case of an n-by-1 matrix, the vector 1-norm matches the matrix 1-norm, and likewise with the ∞-norm. This is how I remember which one is the max row sum and which is the max column sum!

The matrix 2-norm is very useful, but it is actually much harder to compute than the 1-norm or the ∞-norm. There is a related matrix norm, the Frobenius norm, which is much easier to compute:

$$\|A\|_F = \sqrt{\sum_{i,j} |a_{ij}^2|}.$$

The Frobenius norm is consistent, but it is not an operator norm[1]

Absolute error, relative error, conditioning

Suppose I want to compute $y = Ax$, where A is a square matrix, but I don't know the true value of x. Instead, I only know $\hat{x} = x + e$ and some bound on $\|e\|$. What can I say about the error in $\hat{y} = A\hat{x}$ as an approximation to y? I know the absolute error in $\hat{y}$ is

$$\hat{y} - y = A\hat{x} - Ax = A(\hat{x} - x) = Ae,$$

and if I have a consistent matrix norm, I can write

$$\|\hat{y} - y\| \leq \|A\|\|e\|$$

Remember, though, I usually care about the relative error:

$$\frac{\|\hat{y} - y\|}{\|y\|} \leq \|A\| \frac{\|x\|}{\|y\|} \frac{\|e\|}{\|y\|}$$

If A is invertible, then $x = A^{-1}y$, and so I can write

$$\frac{\|x\|}{\|y\|} = \frac{\|A^{-1}y\|}{\|y\|} \leq \|A^{-1}\|.$$

Therefore,

$$\frac{\|\hat{y} - y\|}{\|y\|} \leq \|A\|\|A^{-1}\| \frac{\|e\|}{\|x\|}.$$

The quantity $\kappa(A) = \|A\|\|A^{-1}\|$ is the condition number for matrix multiplication. It is also the condition number for multiplication by the inverse (solving a linear system).

Ill-conditioned matrices are "close to" singular in a well-defined sense: if $\kappa(A) \gg 1$, then there is a perturbation E, $\|E\| \ll \|A\|$, such that $A + E$ is singular. An exactly singular matrix (which has no inverse) can be thought of as infinitely ill-conditioned. That is because if x is a null vector for A, an arbitrarily small relative error in x can lead to an "infinite" relative error in Ax (since no nonzero vector can be a good approximation to a zero vector).

[1] The first half of this sentence is basically Cauchy-Schwarz; the second half of the sentence can be seen by looking at $\|I\|_F$. If you don't understand this footnote, no worries.

Matrix perturbation and a useful power series

This seems like a good place to mention a Taylor expansion that we will see more than once in the coming weeks. For $|x| < 1$, we can write an absolutely convergent series:

$$(1-x)^{-1} = \sum_{j=0}^{\infty} x^j.$$

As a concrete example of this, we have that if δ is around ϵ_{mach},

$$(1+\delta)^{-1} \approx 1 - \delta$$

The utility of this power series is not restricted to real numbers! When E is a matrix such that $\|E\| < 1$ in some consistent norm[2], we have the absolutely convergent matrix power series

$$(I-E)^{-1} = \sum_{j=0}^{\infty} E^j.$$

The linear approximation $(I+E)^{-1} \approx I - E$ is particularly useful.

Now, suppose I had a routine that *exactly* computed $\hat{x} = \hat{A}^{-1}b$ where $\hat{A} = A+E$. How would I estimate the relative error in $\hat{x}$ as an approximation to $x = A^{-1}b$? Let's work through the analysis step by step:

1. **Write an equation for the absolute error**

   The absolute error $\hat{y} - y$ in this case is

   $$\begin{aligned} \hat{y} - y &= (A+E)^{-1}b - A^{-1}b \\ &= (I + A^{-1}E)^{-1}A^{-1}b - A^{-1}b \\ &= \left((I + A^{-1}E)^{-1} - I\right) y \end{aligned}$$

   I'm happy to have the error expressed as a function of y, because I ultimately plan to "divide through by y" to get the relative error.

2. **Simplify using first-order Taylor expansions**

[2] Strictly speaking, all we need to guarantee convergence is that all the eigenvalues of E are inside the unit circle in the complex plane. But it is usually more convenient to test the norm condition.

If A is nonsingular and E is small enough, then

$$(I + A^{-1}E)^{-1} - I \approx -A^{-1}E$$

using the Taylor series that I just introduced a moment ago. Therefore,

$$\hat{y} - y \approx -A^{-1}Ey.$$

3. **Introduce norms**

   Assuming we are using a consistent matrix norm,

$$\|\hat{y} - y\| \lesssim \|A^{-1}\|\|E\|\|y\|.$$

4. **Get bounds on relative error**

   Dividing through by the size of A (since E perturbs A) and the size of y, we have

$$\frac{\|\hat{y} - y\|}{\|y\|} \lesssim \|A^{-1}\|\|A\| \frac{\|E\|}{\|A\|} = \kappa(A)\frac{\|E\|}{\|A\|}.$$