Numerical Linear Algebra – Norms and Conditioning

CS370 Lecture 32 – March 31, 2017

Recall: Standard Matrix Norms

$$||A||_1 = \max_j \sum_{i=1}^{n} |A_{ij}|$$ (Max absolute column sum.)

$$||A||_2 = \sqrt{\text{max magnitude eigenvalue of } A^T A}$$

$$||A||_\infty = \max_i \sum_{j=1}^{n} |A_{ij}|$$ (Max absolute row sum.)

Equivalence of Norms

Another useful property of vector/matrix norms is *equivalence*.

The norms we've looked at differ from one another by no more than a *constant factor*.

That is

$$C_1||x||_a \leq ||x||_b \leq C_2||x||_a$$

for constants C_1, C_2, and norms $||\cdot||_a$ and $||\cdot||_b$.

Conditioning of Linear Systems

Conditioning describes how the output of a function/operation/matrix changes due to changes in input.

Conditioning is indicative of how difficult a problem is to solve, *independent* of the algorithm / numerical method used.

Norms are a useful tool to help characterize the conditioning of linear systems.

Conditioning

Let’s put our norms to use in studying *conditioning* of matrices.

For a linear system $Ax = b$, we ask:

1. How much does a perturbation of b cause the solution x to change?
2. How much does a perturbation of A cause the solution x to change?

Conditioning

For a given perturbation, we say the system is

- Well-conditioned if x changes little.
- Ill-conditioned if x changes lots.

For an ill-conditioned system, small errors can be radically magnified!

This can have disastrous effects on the computed solution, e.g., due to floating point round-off error.

Conditioning Example – Perturbing b

Consider the system:

$$\begin{bmatrix} 1 & 2 \\ 2 & 3.999 \end{bmatrix} x = \begin{bmatrix} 4 \\ 7.999 \end{bmatrix}$$

And the similar system (perturbed b):

$$\begin{bmatrix} 1 & 2 \\ 2 & 3.999 \end{bmatrix} x = \begin{bmatrix} 4.001 \\ 7.998 \end{bmatrix}$$

How do the solutions differ? (e.g., try it in Matlab.)

Tiny change in b; huge difference in solution!

Conditioning Example – Perturbing A

Consider another similar system (perturbing A) :

$$\begin{bmatrix} 1.001 & 2.001 \\ 2.001 & 3.998 \end{bmatrix} x = \begin{bmatrix} 4.001 \\ 7.998 \end{bmatrix}$$

Again, compare the solutions.

Tiny change in A; huge difference in solution!

Notice A is close to the singular matrix, $\mathrm{A} = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}$.

We'll try to characterize when these effects are likely to occur.

Condition Number Summary

Condition number of a matrix A is denoted $\kappa(A) = ||A|| \cdot ||A^{-1}||$.

$\kappa \approx 1 \rightarrow A$ is well-conditioned.

$\kappa \gg 1 \rightarrow A$ is ill-conditioned.

For system $Ax = b$, $\kappa(A)$ provides upper bounds on relative change in x due to relative change in b ...

$$\frac{||\Delta x||}{||x||} \leq \kappa(A) \frac{||\Delta b||}{||b||}$$

or in A ...

$$\frac{||\Delta x||}{||x + \Delta \mathrm{x}||} \leq \kappa(A) \frac{||\Delta A||}{||A||}$$

κ Depends on the Norm

We defined the condition number as

$$\kappa(A) = ||A|| \cdot ||A^{-1}||$$

without specifying *which* norm. Different norms will give different κ.

We can specify the norm with a subscript,

e.g.,

$$\kappa_2(A) = ||A||_2 \cdot ||A^{-1}||_2$$

If unspecified, always assume the 2-norm.

Matlab support

Matlab supports matrix norms and condition numbers like so:

norm(A,1) = $||A||_1$ cond(A,1) = $\kappa_1(A)$
norm(A,2) = $||A||_2$ cond(A,2) = $\kappa_2(A)$
norm(A,inf) = $||A||_\infty$ cond(A,inf) = $\kappa_\infty(A)$

By default (no 2nd arg), Matlab will use the 2-norm.

Numerical Solutions: Residuals and Errors

Condition number plays a role in understanding/bounding the accuracy of numerical solutions (e.g., for $Ax = b$).

If we compute an approximate solution x_{approx}, how "good" is it?

We don't know! We would need the exact solution, x, for comparison.

e.g., recall: relative error = $\frac{||x - x_{approx}||}{||x||}$

Residual

As a stand-in or proxy for error, we often use the **residual** r**:**

$$r = b - A(x_{approx}).$$

i.e., by how much does our computed solution *fail* to satisfy the original problem?

This we *can* compute easily! We know $A, b,$ and our computed x_{approx}.

(But still not exactly what we want…)

Use of the Residual – Iterative Methods

Many alternate algorithms for solving $Ax = b$ are called *iterative*.

Similar to Page Rank, they iteratively improve a solution estimate.

Size of residual dictates when to stop.

e.g.

```
Do
    Improve estimate x_cur of x;
    Recompute r = b-Ax_cur;
While (norm(r) > tolerance)
```

See CS475 for more on iterative schemes!

Residual vs. Error

OK, we can compute the residual. How does it relate to error?

Assuming $x_{approx} = x + \Delta x$, we have

$$r = b - A(x + \Delta x)$$

or

$$A(x + \Delta x) = b - r.$$

(r looks like a perturbation of b.)

$$\boxed{\frac{||\Delta x||}{||x||} \leq \kappa(A)\frac{||\Delta b||}{||b||}}$$

So, applying our earlier bound, we find that

$$\frac{||\Delta x||}{||x||} \leq \kappa(A)\frac{||r||}{||b||}.$$

Interpreting This Bound

$$\frac{||\Delta x||}{||x||} \leq \kappa(A)\frac{||r||}{||b||}.$$

The solution's relative error, $\frac{||\Delta x||}{||x||}$, is bounded by the condition number times the *relative* size of residual r w.r.t. to rhs b.

Moral: If we (roughly) know A's condition number, the computed residual indicates how large the error might be (at worst).

Interpreting This Bound

$$\frac{||\Delta x||}{||x||} \leq \kappa(A) \frac{||r||}{||b||}.$$

Can't avoid that $\frac{||r||}{||b||} \approx \epsilon_{machine}$ (at least).

If $\kappa \approx 1$, small residual indicates a fairly small relative error.

But, if κ is large, residual could be small while error is very large!

Gaussian Elimination & Error

In floating point, Gaussian elimination with pivoting on $Ax = b$ is quite stable and accurate.

A valid interpretation is that GE's numerical result $\hat{x}$ gives the ***exact solution to a "nearby" problem*** (i.e. with perturbed A),

$$(A + E)\hat{x} = b,$$

where $||E|| = ||A|| \cdot \epsilon_{machine}$.

$$\boxed{\frac{||\Delta x||}{||x + \Delta \mathrm{x}||} \le \kappa(A) \frac{||\Delta A||}{||A||}}$$

Again, applying our earlier bound gives

$$\frac{||x - \hat{x}||}{||\hat{x}||} \le \kappa(A)\epsilon_{machine}$$

Reminder:
Conditioning is algorithm-independent!

Recall that conditioning is a property of the problem itself - *not* a property of a particular algorithm.

i.e., A system $Ax = b$ is well- or ill-conditioned, independent of how we choose to solve it.

Even an "ideal" numerical algorithm can't guarantee a solution with small error if $\kappa \gg 1$.

Summary: Conditioning & Norms

Norms let us measure the sizes of vectors and matrices.

They can be used to derive *condition numbers* for linear algebra problems.

The matrix condition number $\kappa(A) = ||A|| \cdot ||A^{-1}||$ plays a key role in bounding the error of numerical solutions.

Examples – Condition Numbers

Find the condition numbers $\kappa_1(A)$ and $\kappa_\infty(A)$ for the matrix

$$A = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

with

$$A^{-1} = \begin{bmatrix} 1/3 & 0 & 0 \\ 0 & 2/3 & -1/3 \\ 0 & -1/3 & 2/3 \end{bmatrix}.$$

$$\kappa(A) = ||A|| \cdot ||A^{-1}||$$

$$||A||_1 = \max_j \sum_{i=1}^{n} |A_{ij}|$$

$$||A||_\infty = \max_i \sum_{j=1}^{n} |A_{ij}|$$

Examples – Condition Numbers

$$A = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$
and
$$A^{-1} = \begin{bmatrix} 1/3 & 0 & 0 \\ 0 & 2/3 & -1/3 \\ 0 & -1/3 & 2/3 \end{bmatrix}.$$

What is $\kappa_2(A)$, if we know the eigenvalues λ_i of $A^T A$ are {1,9,9}?

First, $||A||_2 = \max_i \sqrt{|\lambda_i|} = \sqrt{|9|} = 3.$

For $||A^{-1}||_2$, need eigenvalues of $A^{-T} A^{-1} = (AA^T)^{-1}$.

Since A happens to be symmetric, so that $AA^T = A^T A$, these two have the same eigenvalues.

Invert the eigenvalues of $A^T A$, giving {1,1/9,1/9}.

Then $||A^{-1}||_2 = \sqrt{|1|} = 1.$

Result: $\kappa_2(A) = 3 \cdot 1 = 3.$

Reminder: $||A||_2 = \sqrt{\text{max mag eigenvalue of } A^T A}$

Examples – Condition Numbers

Are the following matrices (relatively) *ill*-conditioned or *well*-conditioned? Why?

1. $\begin{bmatrix} 10^{10} & 0 \\ 0 & 10^{10} \end{bmatrix}$

2. $\begin{bmatrix} 10^{10} & 0 \\ 0 & 10^{-10} \end{bmatrix}$

3. $\begin{bmatrix} 10^{-10} & 0 \\ 0 & 10^{-10} \end{bmatrix}$

Examples – Condition Numbers

$$||A|| = 0 \leftrightarrow A_{ij} = 0 \; \forall i, j.$$
$$||\alpha A|| = |\alpha| \cdot ||A|| \text{ for scalar } \alpha$$
$$||A + B|| \le ||A|| + ||B||$$
$$||A\boldsymbol{x}|| \le ||A|| \cdot ||\boldsymbol{x}||$$
$$||AB|| \le ||A|| \cdot ||B||$$
$$||I|| = 1$$

Using our properties of norms…

- Why is it true that $\kappa(\alpha A) = \kappa(A)$?
- Why is it true that $\kappa(A) \ge 1$?