Solving linear systems of equations

§3.6 Condition Numbers

MA385 – Numerical Analysis 1

November 2019

Numerical solutions to some linear systems are adversely affected by round-off errors.

This phenomenon is related the matrices in the linear systems. Those matrices for which the issue is particularly prevalent are referred to as being ***ill-conditioned***.

For any matrix, we can assign a numerical score that gives an indication of whether it is ill-conditioned. That score is called the ***condition number***, and is the subject of these section.

The condition number is defined in terms of matrix norms.

Consistency of matrix norms

Suppose we have a vector norm, $\|\cdot\|$ and associated subordinate matrix norm. It is not hard to see that

$$\|A\boldsymbol{u}\| \leq \|A\|\|\boldsymbol{u}\| \quad \text{for } \textit{any } \boldsymbol{u} \in \mathbb{R}^n, A \in \mathbb{R}^{n \times n}.$$

Here is why:

There is an analogous statement for the product of two matrices:

Definition 3.26 (Consistent matrix norm)

A matrix norm $\|\cdot\|$ is ***consistent*** (or "*sub-multiplicative*" if

$$\|AB\| \leq \|A\|\|B\|, \qquad \text{for all } A, B \in \mathbb{R}^{n\times n}.$$

Theorem 3.27

Any subordinate matrix norm is consistent.

The proof is left to Exercise 3.17. That exercises also demonstrates that there are matrix norms which are *not* consistent.

[Please read this slide in your own time!]

Modern computers don't store numbers in decimal (base 10), but in binary (base 2) "floating point numbers" of the form :

$$x = \pm a \times 2^{b-M}.$$

Most use *double precision*, where 8 bytes (64 bits or *binary digits*) are used to store

- the sign (1 bit),
- a, called the "significand" or "mantissa" (52 bits)
- and the exponent, $b - 1023$ (11 bits)

Note that a has roughly 16 decimal digits.

(Some older computer systems sometimes use *single precision* where a has 23 bits — giving 8 decimal digits — and b has 7; so too do many new GPU-based systems).

[OK, you can start reading again!] When we try to store a real number x on a computer, we actually store the nearest floating-point number. That is, we end up storing $x + \delta x$, where δx is the "round-off" error.

But the quantity we are mainly interested in is the **relative error:** $|\delta x|/|x|$.

Since this is not a course on computer architecture, we'll simplify a little and just take it that single and double precision systems lead to a relative error of 10^{-8} and 10^{-16} respectively.

(Sew p68–70 of Süli and Mayers for a thorough development of the concept of a condition number).

Suppose we use, say, LU-factorization and back-substitution on a computer to solve

$$A\boldsymbol{x} = \boldsymbol{b}.$$

Because of the "round-off error" we actually solve

$$A(\boldsymbol{x} + \boldsymbol{\delta x}) = (b + \boldsymbol{\delta b}).$$

Our problem now is, for a given A, if we know the (relative) error in $\boldsymbol{b}$, can we find an upper-bound on the relative error in $\boldsymbol{x}$?

Definition 3.28

The *condition number* of a matrix, with respect to a particular matrix norm $\|\cdot\|_\star$ is

$$\kappa_\star(A) = \|A\|_\star \|A^{-1}\|_\star.$$

If $\kappa_\star(A) \gg 1$ then we say A is *ill-conditioned*.

Example: Find the condition number κ_∞ of

$$A = \begin{pmatrix} 10 & 12 \\ 0.08 & 0.1 \end{pmatrix}.$$

Theorem 3.29

Suppose that $A \in \mathbb{R}^{n\times n}$ is nonsingular and that $\boldsymbol{b}, x \in \mathbb{R}^n$ are non-zero vectors. If $A\boldsymbol{x} = \boldsymbol{b}$ and $A(\boldsymbol{x} + \boldsymbol{\delta x}) = (\boldsymbol{b} + \boldsymbol{\delta b})$ then

$$\frac{\|\boldsymbol{\delta x}\|}{\|\boldsymbol{x}\|} \leq \kappa(A)\frac{\|\boldsymbol{\delta b}\|}{\|\boldsymbol{b}\|}.$$

Example 3.30

Suppose we are using a computer to solve $A\boldsymbol{x} = \boldsymbol{b}$ where

$$A = \begin{pmatrix} 10 & 12 \\ 0.08 & 0.1 \end{pmatrix} \quad \text{and } \boldsymbol{b} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

But, due to round-off error, right-hand side has a relative error (in the ∞-norm) of 10^{-6}. Give a bound for the relative error in $\boldsymbol{x}$ in the ∞-norm.

For every matrix norm we get a different condition number.

Example 3.31

Let A be the $n \times n$ matrix

$$A = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 0 & 1 & \dots & 0 \\ \vdots & & & & \\ 1 & 0 & 0 & \dots & 1 \end{pmatrix}.$$

What are $\kappa_1(A)$, and $\kappa_\infty(A)$?

First we compute $\|A\|_1$ and $\|A_\infty\|$.

For this very special example, it is easy to write down the inverse of A:

$$A^{-1} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ -1 & 1 & 0 & \dots & 0 \\ -1 & 0 & 1 & \dots & 0 \\ \vdots & & & & \\ -1 & 0 & 0 & \dots & 1 \end{pmatrix}.$$

To compute $\kappa_1(A)$ and $\kappa_\infty(A)$, we need to know A^{-1}, which is usually not practical. However, for κ_2, we are able to *estimate* the condition number of A without knowing A^{-1}.

Recall that $\|A\|_2 = \sqrt{\lambda_n}$ where λ_n is the largest eigenvalue of $B = A^T A$.

We can also show that $\|A^{-1}\|_2 = \dfrac{1}{\sqrt{\lambda_1}}$ where λ_1 is the smallest eigenvalue of B (see Section 3.6.5 of notes). So

$$\kappa_2(A) = \Big(\lambda_n/\lambda_1\Big)^{1/2}.$$

. . . .

Motivated by this, we'll finish MA385, by studying an easy way of estimating the eigenvalues of a matrix.

Exercise 3.17

(i) Prove that, if $\|\cdot\|$ is a subordinate matrix norm, then it is *consistent*, i.e., for any pair of $n \times n$ matrices, A and B, we have $\|AB\| \leq \|A\|\|B\|$.

(ii) One might think it intuitive to define the "max" norm of a matrix as follows:

$$\|A\|_{\underset{\infty}{\sim}} = \max_{i,j} |a_{ij}|.$$

Show that this is indeed a norm on $\mathbb{R}^{n \times n}$. Show that, however, it is not consistent.

Exercise 3.18

Let A be the matrix

$$A = \begin{pmatrix} 0.1 & 0 & 0 \\ 10 & 0.1 & 10 \\ 0 & 0 & 0.1 \end{pmatrix}$$

Compute $\kappa_\infty(A)$. Suppose we wish to solve the system of equations $A\boldsymbol{x} = \boldsymbol{b}$ on *single precision* computer system (i.e., the relative error in any stored number is approximately 10^{-8}). Give an upper bound on the relative error in the computed solution $\boldsymbol{x}$.