MA 580; Numerical Analysis I

C. T. Kelley
NC State University
tim_kelley@ncsu.edu

Version of September 5, 2016

NCSU, Fall 2016

Part II: Notation, Background, Errors

Contents

Notation: Vectors

- $\mathbf{R}^N$ is the space of N dimensional vectors.
- $\mathbf{R}^{M \times N}$ is the space of $M \times N$ matrices.
- Vectors $\mathbf{x}$ are column vectors

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} = (x_1, \ldots, x_N)^T.$$

Notation: Matrices

- Matrices $\mathbf{A}$ are $M \times N$. The ijth component is

$$a_{ij} \text{ or, if } \mathbf{A} \text{ is a complex expression } \mathbf{A}_{ij}$$

- If $\mathbf{a}_j \in \mathbf{R}^M$ is the jth column of $\mathbf{A}$, we can write $\mathbf{A}$ as

$$\mathbf{A} = (\mathbf{a}_1, \dots, \mathbf{a}_N).$$

Matrix Products

If $\mathbf{A}$ is $M \times L$ and $\mathbf{B}$ is $L \times N$, the matrix product $\mathbf{C} = \mathbf{AB}$ is $M \times N$ and

$$c_{ij} = \sum_{l=1}^{L} a_{il} b_{lj}$$

Example: scalar product

$$\mathbf{v}^T \mathbf{u} = \sum_{i=1}^{N} v_i u_i$$

More Matrix Notation

- The transpose of $\mathbf{A} \in \mathbf{R}^{M \times N}$ is $\mathbf{A}^T \in \mathbf{R}^{N \times M}$ and $\mathbf{A}_{ij}^T = \mathbf{A}_{ji}$. So, the transpose of a column vector is a row vector.
- The identity matrix $\mathbf{I} \in \mathbf{R}^{N \times N}$: $\mathbf{I}\mathbf{x} = \mathbf{x}$ for all $\mathbf{x}$.

$$(\mathbf{I})_{ij} = \delta_{ij} = \begin{cases} 1 \text{ if } i = j \\ 0 \text{ otherwise} \end{cases}$$

- Define the inverse of $\mathbf{A} \in R^{N \times N}$, $\mathbf{A}^{-1}$: $\mathbf{A}^{-1}\mathbf{A} = \mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$

Canonical vectors

We call the columns of $\mathbf{I}$

$$\mathbf{I} = \begin{pmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \dots & \mathbf{e}_n \end{pmatrix}$$

the canonical vectors.
Warning: sometimes $\mathbf{e}$ is also an error. You'll figure it out from the context.

Canonical Vectors

$$\mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \qquad \mathbf{e}_2 = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \qquad \cdots, \qquad \mathbf{e}_N = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix}.$$

Sequences of Vectors

If $\{\mathbf{x}_k\}$ is a sequence of vectors, the ith component of $\mathbf{x}_k$ is

$$x_{ik}$$

as it would be if you regarded the sequence as a large matrix

$$\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots)$$

and used i as the row index and the iteration counter as the column index.

Vector Norms

Review the definition of norm if you don't remember it.
The three important norms are $\ell^1, \ell^2, \ell^\infty$.

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{N} |x_i|, \ \|\mathbf{x}\|_2 = \left(\sum_{i=1}^{N} |x_i|^2 \right)^{1/2}, \text{and } \|\mathbf{x}\|_\infty = \max_i |x_i|.$$

More generally, the ℓ^p norm:

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^{N} |x_i|^p \right)^{1/p}$$

for $1 \le p < \infty$.
Generally the specific norm will not be so important. If I don't say what the norm is then it is any vector norm.

Matrix Norms

We will consider matrix norms which are **induced** by vector norms. This means

$$\|\mathbf{A}\| = \max_{\|\mathbf{x}\|=1} \|\mathbf{A}\mathbf{x}\|.$$

For $\mathbf{A} \in \mathbf{R}^{N \times N}$, induced norms satisfy

$$\|\mathbf{A}\mathbf{B}\| \leq \|\mathbf{A}\|\|\mathbf{B}\|$$

and

$$\|\mathbf{A}\mathbf{x}\| \leq \|\mathbf{A}\|\|\mathbf{x}\|$$

The norm of the product is less than or equal to the product of the norms.

Computing the ℓ^∞ matrix norm: I

Do the math (absolute value of sum $\le$ sum of absolute values) to get

$$|(\mathbf{Ax})_i| = |\sum_{j=1}^{N} a_{ij}x_j| \le \sum_{j=1}^{N} |a_{ij}||x_j| \le \|\mathbf{x}\|_\infty \max_i \sum_{j=1}^{N} |a_{ij}|$$

for all $\mathbf{x}$ and all i. So,

$$\|\mathbf{A}\|_\infty \le \max_i \sum_{j=1}^{N} |a_{ij}| = \text{ maximum absolute row sum}$$

We'll show that the norm is the maximum absolute row sum.

Computing the ℓ^∞ matrix norm: II

To show that the $\leq$ is really $=$. You'll need to find a vector that attains the bound. Here it is.
Let i^* be the row of $\mathbf{A}$ with max norm

$$\sum_{j=1}^{N} |a_{i^*j}| = \max_i \sum_{j=1}^{N} |a_{ij}|$$

and let $x_j = sign(a_{i^*j})$. Then $\|\mathbf{x}\| = 1$ and

$$(\mathbf{Ax})_{i^*} = \sum_{j=1}^{N} |a_{i^*j}|$$

SHAZAM!

ℓ^1 matrix norm?

$$\|\mathbf{A}\|_1 = \max_j \sum_{i=1}^{N} |a_{ij}|$$

the max absolute **column** sum.
You should be able to prove this.
We'll do ℓ^2 later.

λ is an eigenvalue of $\mathbf{A}$ with corresponding eigenvector $\mathbf{x}$ if

$$\mathbf{A}\mathbf{x} = \lambda \mathbf{x}$$

Any eigenvalue is a root of the characteristic polynomial of $\mathbf{A}$

$$p_C(z) = \det(z\mathbf{I} - \mathbf{A})$$

p_C has N roots by the fundamental theorem of algebra. So there are N eigenvalues, counted with multiplicity.

Multiplicity

The algebraic multiplicity of an eigenvalue λ is the number of times it appears in the factorization of p_C into linear terms

$$p_C(z) = \prod_{i=1}^{N} (z - \lambda_i)$$

The geometric multiplicity of λ is the dimension of the null space of $\lambda \mathbf{I} - \mathbf{A}$.

The two notions of multiplicity are not the same. Take

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix},$$

please.

Diagonal Matrices

A matrix $\mathbf{D}$ is diagonal if $d_{ij} = 0$ unless $i = j$.

$$\mathbf{D} = \begin{pmatrix} d_{11} & & \\ & \ddots & \\ & & d_{nn} \end{pmatrix}.$$

In this case we sometimes abbreviate d_{ii} as d_i and write

$$\mathbf{D} = \mathrm{diag}(d_1, \dots, d_N)$$

where d_i is the ith entry in the diagonal.
Example: $\mathbf{I} = \mathrm{diag}(1, 1, \dots, 1)$

Diagonalizable Matrices

A matrix is diagonalizable if it has a basis of eigenvectors $\{\mathbf{v}_i\}_{i=1}^N$. In this case the matrix $\mathbf{V} = (\mathbf{v}_1, \dots, \mathbf{v}_N)$ is nonsingular and

$$\mathbf{A} = \mathbf{V}\Lambda\mathbf{V}^{-1}$$

where $\Lambda = \mathrm{diag}(\lambda_1, \dots, \lambda_N)$.

Triangular Matrices

$\mathbf{A} \in R^{N\times N}$ is **upper triangular** if $a_{ij} = 0$ for $i > j$. That is,

$$\mathbf{A} = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ & \ddots & \vdots \\ & & a_{nn} \end{pmatrix}.$$

$\mathbf{A}$ is lower triangular if $\mathbf{A}^T$ is upper triangular.

Orthogonal Matrices

An $N \times N$ matrix $\mathbf{U}$ is orthogonal if

$$\mathbf{U}^{-1} = \mathbf{U}^T$$

This means that the columns of $\mathbf{U}$ form an orthonormal basis. Orthogonal matrices are a very big deal in this course.

Symmetric Matrices

$\mathbf{A}$ is symmetric is $\mathbf{A} = \mathbf{A}^T$.

A few facts:

- The eigenvalues $\{\lambda_i\}_{i=1}^N$ of $\mathbf{A}$ are real.
- There's an orthonormal basis $\{\mathbf{u}_i\}_{i=1}^N$ of eigenvectors of $\mathbf{A}$.
- $\mathbf{U} = (\mathbf{u}_1, \dots, \mathbf{u}_N)$ is an orthogonal matrix.
- $\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^T$ (spectral theorem for symmetric matrices)

Prove some of this stuff.

Symmetric Positive (semi)Definite Matrices

A symmetric matrix $\mathbf{A}$ is symmetric positive definite (spd) if

$$\mathbf{x}^T \mathbf{A} \mathbf{x} > 0 \text{ for all } \mathbf{x} \neq 0$$

Alternatively, $\mathbf{A}$ is spd if all its eigenvalues are positive.
Positive semidefinite: all eigenvalues nonnegative or

$$\mathbf{x}^T \mathbf{A} \mathbf{x} \geq 0 \text{ for all } \mathbf{x}$$

Rank-one or Outer Product Matrices

Recall the **inner product** of $\mathbf{v}$ and $\mathbf{u}$

$$\mathbf{u}^T\mathbf{v} = (1 \times N)(N \times 1) = (1 \times 1) = \sum_{i=1}^{N} u_i v_i.$$

The **outer product** is

$$\mathbf{u}\mathbf{v}^T = (N \times 1)(1 \times N) = N \times N,$$

and $(\mathbf{u}\mathbf{v}^T)_{ij} = u_i v_j$

Rank One Matrices

- $\mathbf{A}$ is rank-one matrix if the range $R(\mathbf{A})$ of $\mathbf{A}$ has dimension one.
- Fact: $\mathbf{A}$ is rank-one if and only if $\mathbf{A} = \mathbf{u}\mathbf{v}^T$ for vectors $\mathbf{u}$ and $\mathbf{v}$. **You get to prove this in the homework.**

Eigen-decomposition of symmetric matrices

If $\mathbf{A} = \mathbf{A}^T$ then

$$\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^T = \sum_{i=1}^{N} \lambda_i \mathbf{u}_i \mathbf{u}_i^T.$$

From this you can prove, for symmetric $\mathbf{A}$,

$$\|\mathbf{A}\|_2 = \max_i |\lambda_i|$$

You'll see things like this on exams.

Eigenvalues/vectors of Rank-One matrices

$$\mathbf{A} = \mathbf{x}\mathbf{y}^T$$

Any vector in $E = \{\mathbf{z} \mid \mathbf{y}^T\mathbf{z} = 0\}$ is a null-vector of $\mathbf{A}$ ($\mathbf{Ax} = 0$).
If $\mathbf{y}^T\mathbf{x} \neq 0$ ($\mathbf{x} \notin E$) then

$$\mathbf{Ax} = (\mathbf{y}^T\mathbf{x})\mathbf{x}$$

so $\lambda = \mathbf{y}^T\mathbf{x}$ is an eigenvalue with eigenvector $\mathbf{x}$.
Algebraic multiplicity = geometric multiplicity!
What if $\mathbf{y}^T\mathbf{x} = 0$?

ℓ^2 norm of symmetric matrix $\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^T$

Since $\mathbf{U}$ is orthogonal then every $\mathbf{x} \in R^N$ has the expansion

$$\mathbf{x} = \sum_i (\mathbf{u}_i^T \mathbf{x})\mathbf{u}_i,$$

which implies that

$$\mathbf{A}\mathbf{x} = \sum_i \lambda_i (\mathbf{u}_i^T \mathbf{x})\mathbf{u}_i.$$

So,

$$\|\mathbf{A}\mathbf{x}\|_2^2 = \sum_i \lambda_i^2 (\mathbf{u}_i^T \mathbf{x})^2 \le \lambda_N^2 \sum_i (\mathbf{u}_i^T \mathbf{x})^2 = \lambda_N^2 \|\mathbf{x}\|^2,$$

with equality if $\mathbf{x} = \mathbf{u}_N$. That's it.

ℓ^2 Matrix Norm

Start with

$$\|\mathbf{Ax}\|_2^2 = (\mathbf{Ax})^T(\mathbf{Ax}) = \mathbf{x}^T(\mathbf{A}^T\mathbf{A})\mathbf{x}.$$

$\mathbf{A}^T\mathbf{A}$ is symmetric positive semidefinite, so

$$\mathbf{A}^T\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^T \text{ where } 0 \le \lambda_1 \le \lambda_2 \le \cdots \le \lambda_N.$$

This means (you should fill in the details) that

$$\|\mathbf{Ax}\|_2^2 \le \lambda_N \|\mathbf{x}\|^2 \text{ with equality if } \mathbf{x} = \mathbf{u}_N.$$

So $\|\mathbf{A}\|_2 = \sqrt{\lambda_N}$ where λ_N is the largest eigenvalue of $\mathbf{A}^T\mathbf{A}$.

The Sherman-Morrison Formula: Rank-one changes

Assume $\mathbf{A} \in R^{N \times N}$; $\mathbf{v}, \mathbf{u} \in R^N$ and

- $\mathbf{A}$ is nonsingular,
- $1 + \mathbf{v}^T \mathbf{A}^{-1} \mathbf{u} \neq 0$.

Then $A + \mathbf{u}\mathbf{v}^T$ is nonsingular and

$$(\mathbf{A} + \mathbf{u}\mathbf{v}^T)^{-1} = \left(\mathbf{I} + \frac{(\mathbf{A}^{-1}\mathbf{u})\mathbf{v}^T}{1 + \mathbf{v}^T \mathbf{A}^{-1} \mathbf{u}} \right) \mathbf{A}^{-1}.$$

How do you prove this? It's your problem in the homework.

Things to think about

- In what sense is the spectral decomposition for a symmetric matrix $\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^T$ unique?
  What about the 1×1 case? What about the 2×2 identity?
- What are the eigenvalues of an orthogonal matrix?
- Is an orthogonal matrix diagonalizable?
- What are the orthogonal spd matrices?

Relative and Absolute Errors

Let $\mathbf{x}$ be a vector, matrix, scalar, ...
Suppose you approximate $\mathbf{x}$ by $\tilde{\mathbf{x}}$. The **absolute** error is

$$\|\mathbf{x} - \tilde{\mathbf{x}}\| \text{ sensitive to units}$$

The **relative** error (for $\mathbf{x} \neq 0$) is

$$\frac{\|\mathbf{x} - \tilde{\mathbf{x}}\|}{\|\mathbf{x}\|} \text{ not sensitive to units}$$

It's ok to think of a relative error of 10^{-k} as meaning that $\mathbf{x}$ and $\tilde{\mathbf{x}}$ agree to k decimal digits.

Floating Point Arithmetic

A real number is an abstraction. It does not physically exist.

- There are uncountably many real numbers.
- There are uncountably many real numbers between any two real numbers.
- You cannot store all the real numbers in a physical device.

On the other hand . . .

A floating point number is like a dog.

It is not an abstraction. It is realer that a real number.

- It consumes energy.
- It radiates heat.
- It occupies space.
- It makes noise.

Geography of Floating Point Numbers: Radix

- Floating point numbers are expressed in three fields in radix (or base) P arithmetic.
- Modern computers use radix 2 (binary) arithmetic.
  The standard is IEEE-754 (1985).
- We will use radix 10 (decimal) for simple examples in this part of the course.
- Historical systems have used 16 (hexadecimal) and 8 (octal).

Geography of Floating Point Numbers: The Fields

$\pm$	Exponent	Mantissa

- Sign bit.
- Exponent field
- Mantissa, significand, or fraction.
- Conventions for uniqueness: normalized numbers
  - The point is before the leading digit.
  - The leading digit of a non-zero number is non-zero.

You can think of this as scientific notation with limited range.

How big is an IEEE float?

	Width (bits)	Sign	Exponent	Manitssa
Double	64	1	11	52
Single	32	1	8	23

- Matlab default is 64 bit IEEE double precision.
- Special objects:
  - INF $1/0, 10^{1000}, \dots$
  - NaN $0/0$, $INF * 0$, missing data in statistics

If you get these, you have a bug!

I have a base 10 brain. What does this mean to me?

In double precision:

- Exponent Range:
- Largest > 0 floating point number: $\approx 1.8 \times 10^{308}$
- Smallest normalized > 0 floating point number:
  $\approx 2.22 \times 10^{-308}$ =`2.22e-302`
  denormalized: `4.9407e-324`, `eps(0)`
- Unit roundoff: $\epsilon_u \approx 1.11 \times 10^{-16}$
  relative error bound for conversion from real to float
  and elementary operations.
- Difference between 1 and floating point number > 1 nearest
  to 1: `2.2204e-16`
  `eps` or `eps(1)`

Why would anyone use single precision?

Ideas?

What you need to remember

- There are finitely many floating point numbers.
- Floating point numers are not equally spaced.
- There's a role (but not in MA580) for single (and lower) precision.
  `help single`
- If computation arrives at
  - something larger than the largest float: **overflow**, get INF
  - something $0 < x < eps(0)$: **underflow**
    gradual or zero
- Overflow is a problem. Underflow is usually not.

Rounding and Floating Point Operations

- Real number: x, Floating point representation $fl(x) = x(1+\epsilon)$, $|\epsilon| \le \epsilon_u$
- IEEE 754 default: round to nearest.
- Operations: Given $x, y \in R$ and elementary operation $\circ$

$$fl(x \circ y) = fl(fl(x) \circ fl(y)) = (x \circ y)(1 + \epsilon_u)$$

Given something to evaluate $f(x)$ you hope that

$$\|fl(f(x)) - f(x)\| = \epsilon_f \|f(x)\|, \text{ where } \epsilon_f \approx sizeof(computation)\epsilon_u$$

A Toy Floating Point Number System

Here's a radix 10 system.

- Two digit mantissa
- Exponent range: -2:2
- Largest float $= .99 * 10^2 = 99$
- eps(0) $= .10 * 10^{-2}$
- eps(1) $= 1.1 - 1 = .1$
- Round to nearest.

Overflow in inner product

Given $\mathbf{x} = (10, 10)^T$ compute $\|\mathbf{x}\|_2$.
Answer $= \sqrt{100 + 100} = \sqrt{200}$ which rounds to $.14 \times 10^2$, a legal floating point number.
Wrong way: Add $x_1 y_1$ (Overflow) to $x_2 y_2$ (Overflow) and take the square root.
Right way:

- Normalize $\mathbf{x}$ to get $\mathbf{w} = \mathbf{x}/\|\mathbf{x}\|_\infty = (1, 1)^T$
- Compute $\|\mathbf{w}\|_2 = \sqrt{2}$ rounds to $.14 \times 10^1$.
- Multiply $\|\mathbf{w}\|$ by $\|\mathbf{x}\|_\infty = 10$ to get $\|\mathbf{x}\|_2 = .14 \times 10^2$

Order of summation

Let's compute

$$\sum_{i=1}^{101} a_i, \text{ where } a_1 = 90, a_2 = a_3 = \dots a_{101} = .01$$

If do we this in order

$$fl(a_1 + a_2) = fl(90.01) = 90, \dots$$

and the summation stagnates with a result of 90 (wrong).
Do it backwards and

$$fl(\sum_{i=101}^{2} .01) = fl(1) = 1, \text{ and } fl(1 + 90) = 91.$$

Moral: sum the small things first.

The Quadratic Formula

From your early childhood, the solutions of

$$ax^2 + bx + c = 0$$

are

$$r_{\pm} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Let's break it.

Catastrophic Cancellation or Loss of Signifcance

Let's find the roots of

$$x^2 + 4x + .1$$

with the quadratic formula. The answer (computed in IEEE double) is

$$r_\pm = \frac{-4 \pm \sqrt{16 - .4}}{2} \approx \begin{cases} -.02515 & + \\ -3.975 & - \end{cases}$$

In the toy floating point system, these round to

$$-.25 \times 10^{-1} \text{ and } - .40 \times 10^{1}$$

Apply the Formula in Toy Floats

Since

$$fl(16 - .4) = fl(15.6) = 16$$

we may be in trouble. When we compute r_- we get

$$r_- = (-4 - 4)/2 = -4 \text{ right! } .$$

But for r_+ we are lost

$$r_+ = (-4 + 4)/2 = 0 \text{ wrong!}$$

The subtraction $16 - .4$ lost all information.

The Fix

Before doing anything else, rewrite the formla for r_+ to avoid the subtraction.

$$r_+ = r_+ \frac{-b - \sqrt{b^2 - 4ac}}{-b - \sqrt{b^2 - 4ac}} = -\frac{4ac}{2a(b + \sqrt{b^2 - 4ac})}$$

For our problem this is

$$-\frac{.4}{2(8)} = -\frac{.1}{4}$$

which rounds to $-.25 \times 10^{-1}$. SHAZAM!

Stability

An algorithm is unstable when applied to a problem if the size of the output grows exponentially as some measure of problem complexity increases.

This is not a precise definition, but

You'll know it when you see it.

An unstable method.

Here's a way to compute eigenvalue-eigenvector pairs.
Let **A** be symmetric with eigenvalues
$|\lambda_1| \le |\lambda_2| \cdots \le |\lambda_{N-1}| < |\lambda_N|$.

- Pick a large n and $\mathbf{x} \in R^N$
- Compute $\mathbf{w} = \mathbf{A}^n \mathbf{x}$
- Set $\mathbf{u}_n = \mathbf{w}/\|\mathbf{w}\|$; $\lambda_n = \mathbf{u}_n^T A \mathbf{u}_N$

Theorem: $\mathbf{u}_n \to \mathbf{u}$ and $\mathbf{Au} = \lambda_N \mathbf{u}$.
This algorithm is unstable even if $N = 1$! Consider $\mathbf{A} = 3$.

But why do you care?

Doesn't the huge size of $\mathbf{w}$ go away when you divide by $\|\mathbf{w}\|$?

- Yes, it does in **exact arithmetic**,
- but not in **floating point**.
  You get an overflow from an intermediate step.

But you can fix this one: Pick $\mathbf{x} \in R^N$

- While not happy
  - $\mathbf{w} = \mathbf{Ax}$
  - $\mathbf{x} = \mathbf{w}/\|\mathbf{w}\|$

Same results, but normalizing each time.

Fix for instability: change algorithm.

Conditioning

- A computation is poorly conditioned if a small relative change in the input produces a large relative change in the output.
- Define small!!!
  This is yet another "I'll know it when I see it." kind of thing.
- You'll get inaccurate results for a sufficiently poorly conditioned computation no matter what you do.
- Fix for poor conditioning: reformulate.

A problem is well conditioned if small relative changes in the input lead to small relative changes in the output.
There is, of course, a large and fuzzy middle ground.

Condition number

We can quantify it this time

$$\kappa = \text{" condition number "} = \frac{\text{norm of relative change in output}}{\text{norm of relative change in out put}}.$$

If $\mathbf{x}$ is the input, $\mathbf{S}(\mathbf{x})$ is the output, and $\delta\mathbf{x}$ is the change in $\mathbf{x}$. Then

$$\kappa = \frac{\|\mathbf{S}(\mathbf{x}+\delta\mathbf{x}) - \mathbf{S}(\mathbf{x})\| / \|\mathbf{S}(\mathbf{x})\|}{\|\delta\mathbf{x}\| / \|\mathbf{x}\|}.$$

Note: κ depends on what you're doing $\mathbf{S}$
and where you're doing it $\mathbf{x}$.

Example: subtraction

Here

$$\mathbf{x} = \begin{pmatrix} x \\ y \end{pmatrix}, \delta\mathbf{x} = \begin{pmatrix} \delta_x \\ \delta_y \end{pmatrix}, \text{ and } \mathbf{S}(\mathbf{x}) = x - y$$

So

$$|\mathbf{S}(\mathbf{x} + \delta\mathbf{x}) - \mathbf{S}(\mathbf{x})| = |\delta_x - \delta_y|,$$

Use the ℓ^1 norm and

$$\|\delta\mathbf{x}\| = |\delta_x| + |\delta_y|, \text{ and } \|\mathbf{x}\| = |x| + |y|.$$

Conditioning of Subtraction

Plug it all into the formula and ...

$$\kappa = \frac{|\delta_x - \delta_y|/|x-y|}{|\delta_x| + |\delta_y|/(|x|+|y|)}.$$

There is no reason to expect δ_x and δ_y to cancel, so the most sensible estimate is

$$\kappa = \frac{|x|+|y|}{|x-y|}$$

Wow! Subtraction of nearly equation numbers is a bad idea again!!!

Conditioning of Matrix-Vector Product

Data: $\mathbf{A}$ fixed, $\mathbf{x}$, $\delta\mathbf{x}$.

$$\kappa = \frac{\|\mathbf{A}(\mathbf{x}+\delta\mathbf{x}) - \mathbf{A}\mathbf{x}\|/\|\mathbf{A}\mathbf{x}\|}{\|\delta\mathbf{x}\|/\|\mathbf{x}\|}.$$

We'll use the estimate

$$\|\mathbf{A}^{-1}\|^{-1}\|\mathbf{x}\| \le \|\mathbf{A}\mathbf{x}\| \le \|\mathbf{A}\|\|\mathbf{x}\|$$

We know the one on the right because norm of product $\le$ product of norms. The one on the left follows from

$$\|\mathbf{x}\| = \|\mathbf{A}^{-1}\mathbf{A}\mathbf{x}\| \le \|\mathbf{A}^{-1}\|\|\mathbf{A}\mathbf{x}\|.$$

Now to estimate this mess ...

Start with this

$$\kappa = \frac{\|\mathbf{A}\delta\mathbf{x}\|/\|\mathbf{A}\mathbf{x}\|}{\|\delta\mathbf{x}\|/\|\mathbf{x}\|}$$

use

$$1/\|\mathbf{A}\mathbf{x}\| \geq 1/(\|\mathbf{A}^{-1}\|\|\mathbf{x}\|) = \|\mathbf{A}^{-1}\|/\|\mathbf{x}\|$$

to get

$$\kappa \leq \frac{\|\mathbf{A}\|\|\mathbf{A}^{-1}\|\|\delta\mathbf{x}\|/\|\mathbf{x}\|}{\|\delta\mathbf{x}\|/\|\mathbf{x}\|} = \|\mathbf{A}\|\|\mathbf{A}^{-1}\|.$$

This leads to the most profound definition in the course ...

Condition number of a matrix **A**

The condition number of an $N \times N$ nonsingular matrix **A** relative to the vector norm $\| \cdot \|$ is

$$\kappa(\mathbf{A}) = \|\mathbf{A}\| \|\mathbf{A}^{-1}\| \geq 1.$$

Can you see why $\kappa(\mathbf{A}) \geq 1$?
When the norm matters, we indicate that with κ. For example

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

A matrix is well-conditioned if $\kappa(A)$ is small, ill-conditioned if $\kappa(A)$ is **HUGE**.
Once again, ill-defined with lots of middle ground.

(over) Generalized view of κ

Return to

$$\kappa = \frac{\|\mathbf{S}(\mathbf{x}+\delta\mathbf{x}) - \mathbf{S}(\mathbf{x})\|/\|\mathbf{S}(\mathbf{x})\|}{\|\delta\mathbf{x}\|/\|\mathbf{x}\|}.$$

and think of

$$\frac{\|\mathbf{S}(\mathbf{x}+\delta\mathbf{x}) - \mathbf{S}(\mathbf{x})\|}{\|\delta\mathbf{x}\|} \approx \|\mathbf{S}_{\mathbf{x}}(\mathbf{x})\mathbf{u}\|.$$

where $\mathbf{u} = \delta\mathbf{x}/\|\delta\mathbf{x}\|$ is a unit vector in the direction $\delta\mathbf{x}$
Then, since $\|\mathbf{u}\| = 1$,

$$\kappa \approx \frac{\|\mathbf{S}_{\mathbf{x}}(\mathbf{x})\|\|\mathbf{x}\|}{\|\mathbf{S}(\mathbf{x})\|}.$$

You'll hear more about this when we get to nonlinear equations.

Bottomline on stability and conditioning

- Stability is a property of the algorithm.
  Fix instability with a change in the algorithm.
- Conditioning is a property of the problem.
  Fix ill conditioning with a reformulation of the problem.

O and o Notation

- $a_n = O(b_n)$ as $n \to \infty$ means that there is K

$$\|a_n\| \leq K\|b_n\| \text{ for all sufficiently large } n.$$

- $a_n = o(b_n)$ as $n \to \infty$ means that

$$\lim_{n\to\infty} \|a_n\|/\|b_n\| = 0.$$

- Similarly $f(h) = O(h^p)$ and $f(h) = o(h^p)$ as $h \to 0$.