Lecture 8 Relative and matrix condition numbers

Recall that a *problem* is a function $f : X \to Y$ from a normed space X to a normed space Y, where the X-norm measures the error in the input and the Y-norm measures the error in the output.

Relative condition number

The *(relative) condition number* of a problem is a measure of how much the relative error in the input is magnified to cause relative error in the output. The mathematical definition of the relative condition number is

$$\kappa_f(\mathbf{x}, \epsilon) \triangleq \sup_{\|\Delta \mathbf{x}\|_X \leq \epsilon} \frac{\frac{\|f(\mathbf{x}+\Delta\mathbf{x}) - f(\mathbf{x})\|_Y}{\|f(\mathbf{x})\|_Y}}{\frac{\|\Delta\mathbf{x}\|_X}{\|\mathbf{x}\|_X}} = \sup_{\|\Delta \mathbf{x}\|_X \leq \epsilon} \frac{\|f(\mathbf{x}+\Delta\mathbf{x}) - f(\mathbf{x})\|_Y}{\|f(\mathbf{x})\|_Y} \frac{\|\mathbf{x}\|_X}{\|\Delta\mathbf{x}\|_X}$$

This gives us a bound on relative errors: if $\|\Delta\mathbf{x}\|_X \leq \epsilon$ we have

$$\frac{\|f(\mathbf{x}+\Delta\mathbf{x}) - f(\mathbf{x})\|_Y}{\|f(\mathbf{x})\|_Y} \leq \kappa_f(\mathbf{x}, \epsilon) \frac{\|\Delta\mathbf{x}\|_X}{\|\mathbf{x}\|_X}$$

Matrix condition numbers

In this section we will always use the 2-norm. Define the *condition number of a matrix* as

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

If A is not invertible, then the condition number is infinite. We see that this gives a bound on the condition numbers of several matrix-vector problems:

Example 1

For a given $A \in \mathbb{R}^{n \times n}$, consider the matrix-vector problem where we measure the error in the vector:

$$f(\mathbf{x}) = A\mathbf{x}$$

The condition number is

$$\kappa_f(\mathbf{x}, \epsilon) = \sup_{\|\Delta\mathbf{x}\| \leq \epsilon} \frac{\|A(\mathbf{x}+\Delta\mathbf{x}) - A\mathbf{x}\|}{\|A\mathbf{x}\|} \frac{\|\mathbf{x}\|}{\|\Delta\mathbf{x}\|} = \frac{\|\mathbf{x}\|}{\|A\mathbf{x}\|} \sup_{\|\Delta\mathbf{x}\| \leq \epsilon} \frac{\|A\Delta\mathbf{x}\|}{\|\Delta\mathbf{x}\|} = \frac{\|\mathbf{x}\|}{\|A\mathbf{x}\|} \|A\|$$

We can bound this by the condition number:

$$\kappa_f(\mathbf{x}, \epsilon) = \frac{\|A^{-1}A\mathbf{x}\|}{\|A\mathbf{x}\|} \|A\| \leq \frac{\|A^{-1}\| \|A\mathbf{x}\|}{\|A\mathbf{x}\|} \|A\| = \|A^{-1}\| \|A\| = \kappa(A)$$

Example 2

For a given $\mathbf{x} \in \mathbb{R}^n$, consider the matrix-vector problem where we measure the error in the *matrix*:

$$f(A) = A\mathbf{x}$$

We can bound the condition number of the problem by the condition number of the matrix

$$\kappa_f(A, \epsilon) = \sup_{\|\Delta A\| \leq \epsilon} \frac{\|(A + \Delta A)\mathbf{x} - A\mathbf{x}\|}{\|A\mathbf{x}\|} \frac{\|A\|}{\|\Delta A\|} = \frac{\|A\|}{\|A\mathbf{x}\|} \sup_{\|\Delta A\| \leq \epsilon} \frac{\|\Delta A\mathbf{x}\|}{\|\Delta A\|} \leq \frac{\|A\|}{\|A\mathbf{x}\|} \sup_{\|\Delta A\| \leq \epsilon} \frac{\|\Delta A\|\|\mathbf{x}\|}{\|\Delta A\|}$$

Example 3

The matrix-inverse problem

$$f(\mathbf{x}) = A^{-1}\mathbf{x}$$

has a condition number also bounded by $\kappa(A)$.

Condition numbers in Julia

We now do some experiments on condition numbers. First consider a random matrix `A`, using the command `cond` to calculate the condition number:

In [74]:

```
A=rand(50,50);
cond(A)    # condition number A, using induced 2-norm
```

Out[74]:

```
6770.353833509765
```

This is equivalent to the following:

In [75]:

```
norm(A)*norm(inv(A))
```

Out[75]:

```
6770.353833509197
```

In [76]:

```
norm(A,2)*norm(inv(A),2)
```

Out[76]:

```
6770.353833509197
```

We can bound the relative error of matrix-vector multiplication using the condition number:

In [78]:

```
n=50
x=rand(n)
Δx=0.0001*rand(n)

norm(A*x-A*(x+Δx))/norm(A*x)   , cond(A)*norm(Δx)/norm(x)
```

Out[78]:

```
(0.00010672805085198968,0.6895570969146974)
```

The bound in this example is very pessimistic. We can also use it to bound the error to perturbation in `A`:

In [79]:

```
ΔA=0.0001*rand(n,n)


norm(A*x-(A+ΔA)*x)/norm(A*x) , cond(A)*norm(ΔA)/norm(A)
```

Out[79]:

```
(0.00010068814777527139,0.6797522163177864)
```

Hilbert matrix

A notorious matrix with extremely bad conditioning is the *Hilbert matrix*, which is constant on the anti-diagonals:

$$H = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots & \frac{1}{n+1} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \cdots & \frac{1}{n+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \frac{1}{n+2} & \cdots & \frac{1}{2n-1} \end{pmatrix}$$

We can create it with the following `for` loop:

In [80]:

```
n=5

H=zeros(n,n)
for k=1:(2n-1)
    for j=1:k
        if (k-j+1)≤n && (j ≤ n)
            H[k-j+1,j]=1/k
        end
    end
end

H
```

Out[80]:

```
5x5 Array{Float64,2}:
 1.0       0.5       0.333333  0.25      0.2
 0.5       0.333333  0.25      0.2       0.166667
 0.333333  0.25      0.2       0.166667  0.142857
 0.25      0.2       0.166667  0.142857  0.125
 0.2       0.166667  0.142857  0.125     0.111111
```

Even for a moderate value of `n`, the condition number is very bad:

In [82]:

```
cond(H)
```

Out[82]:

```
476607.2502422621
```

Again, the condition number gives a (very pecimmistic bound)

In [93]:

```
x=ones(n)
Δx=0.00001*(-1.0).^(1:n)

norm(H*(x+Δx)-H*x)/norm(H*x) , cond(H)*norm(Δx)/norm(x)
```

Out[93]:

```
(3.012468075974052e-6,4.766072502422621)
```

But certain vectors produce much worse error, still bounded by the condition number:

In [94]:

```
x=[0.006173863456720333,-0.11669274684770821,0.50616365835256,-0.76719119308
51485,0.3762455454339546]

Δx=0.00001*(-1.0).^(1:n)


norm(H*(x+Δx)-H*x)/norm(H*x) , cond(H)*norm(Δx)/norm(x)
```

Out[94]:

```
(2.8753568306615302,10.657262101109513)
```

Simple 2x2 example

This simple 2x2 example demonstrates why special vectors can cause large relative errors:

In [95]:

```
A=[1. 0.0001;
   1.3 0.0001]

x=[0.,1.]

Δx=0.0001rand(2)

norm(A*x-A*(x+Δx))/norm(A*x)  , cond(A)*norm(Δx)/norm(x)
```

Out[95]:

```
(0.2401683222870867,3.2384200040168505)
```

A generic vector doesn't have the same error, however: condition number is always just an upper bound.

In [96]:

```
A=[1. 0.0001;
   1.3 0.0001]

x=[1.,1.]

Δx=0.0001rand(2)

norm(A*x-A*(x+Δx))/norm(A*x)  , cond(A)*norm(Δx)/norm(x)
```

Out[96]:

```
(6.957056820336793e-5,7.684024504106144)
```