Numerical Conditioning

1. Forward and Backward Error
   - nearby solutions for nearby problems
2. Multiple Roots
   - where functions are very flat
3. The Wilkinson Polynomial
   - a historical example in numerical analysis
4. Condition Number
   - the error magnification factor

MCS 471 Lecture 4
Numerical Analysis
Jan Verschelde, 20 January 2021

Numerical Conditioning

1. Forward and Backward Error
   - nearby solutions for nearby problems

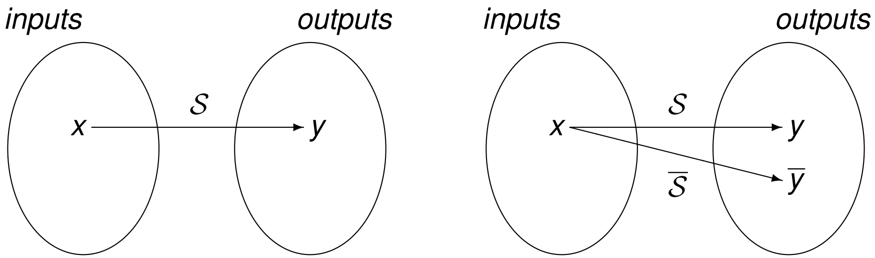
2. Multiple Roots
   - where functions are very flat

3. The Wilkinson Polynomial
   - a historical example in numerical analysis

4. Condition Number
   - the error magnification factor

nearby solutions for nearby problems

Consider a solution map $\mathcal{S}$ between the space of *inputs* and *outputs*.



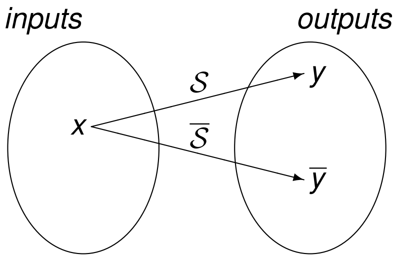
Because of floating-point arithmetic, we obtain $\overline{y} = \overline{S}(x)$.

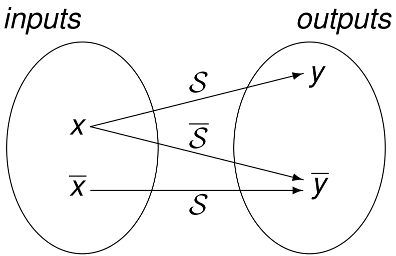
Definition

If $y = \mathcal{S}(x)$ is the exact solution for input x and $\overline{y} = \overline{\mathcal{S}}(x)$ is the approximate solution for the input x, then *the forward error* is $|y - \overline{y}|$.

forward error and backward error

For $y = \mathcal{S}(x)$ and $\overline{y} = \overline{\mathcal{S}}(x)$, the forward error is $|y - \overline{y}|$.
Modify the input x into $\overline{x}$ so that $\overline{y} = \mathcal{S}(\overline{x})$.





Definition

If $y = \mathcal{S}(x)$ is the exact solution for input x, $\overline{y} = \overline{\mathcal{S}}(x)$ is the approximate solution for the input x, and $\overline{x}$ is the modified input so that $\overline{y} = \mathcal{S}(\overline{x})$, then *the backward error* is $|x - \overline{x}|$.

application to the root finding problem

Solve the equation $f(x) = 0$ to find a root r.

We find an approximation $\overline{r}$: $f(\overline{r}) \approx 0$.
Denote Δr such that $\Delta r = r - \overline{r}$ or $r = \overline{r} + \Delta r$.
We have $f(r) = 0$ and thus $f(\overline{r} + \Delta r) = 0$, $|\Delta r|$ is the forward error.

Denote Δf such that $\Delta f = f - \overline{f}$, where $\overline{f}(\overline{r}) = 0$.

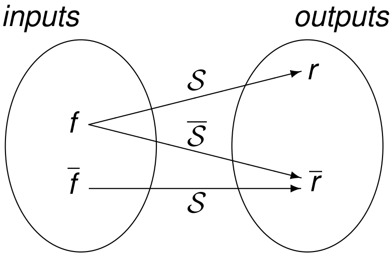
$$\begin{aligned} \overline{f}(\overline{r}) = 0 \quad & \Leftrightarrow \quad (f - \Delta f)(\overline{r}) = 0 \\ & \Leftrightarrow \quad f(\overline{r}) = \Delta f(\overline{r}) \end{aligned}$$

The backward error is $|\Delta f(\overline{r})| = |f(\overline{r})|$.

Although we do not know Δr, $f(\overline{r})$ is a simple evaluation.

relation between forward and backward error?

For the root finding problem: $r = \mathcal{S}(f)$, $\overline{r} = \overline{\mathcal{S}}(f)$, and $\overline{r} = \mathcal{S}(\overline{f})$.



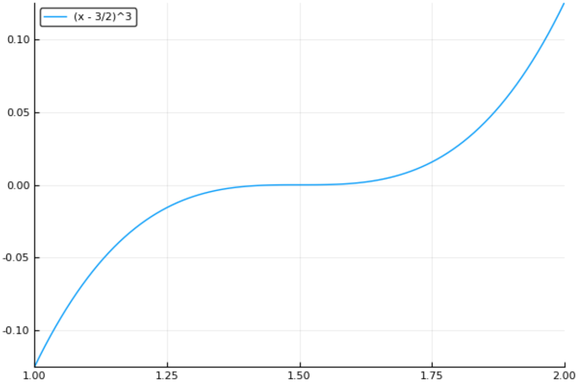
As the backward error $|\Delta f| = |f(\overline{r})|$, is $|\Delta f| \approx |\Delta r|$?

Numerical Conditioning

1. Forward and Backward Error
   - nearby solutions for nearby problems
2. Multiple Roots
   - where functions are very flat
3. The Wilkinson Polynomial
   - a historical example in numerical analysis
4. Condition Number
   - the error magnification factor

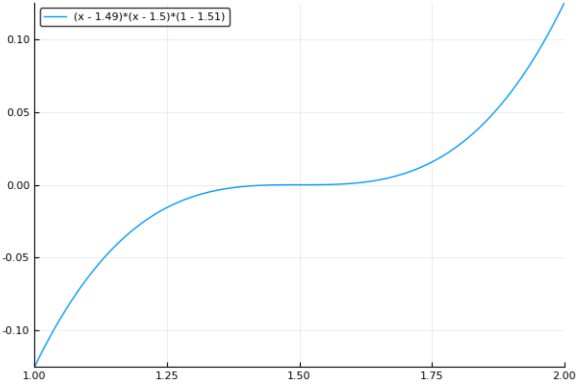
a multiple root

Consider the plot of $(x - 3/2)^3$:



close to a multiple root

Consider the plot of $(x - 1.49)(x - 1.5)(x - 1.51)$:



relation between forward and backward error?

Consider $p(x) = (x - 1.5)^3$ and $q(x) = (x - 1.51)(x - 1.5)(x - 1.49)$.
The forward error is the error in the roots: $|\Delta r| = 0.01 = 10^{-2}$.

Consider the difference in the coefficients of p and q:

$$\begin{array}{rcl} p(x) & = & x^3 - 4.5x^2 + 6.75x - 3.375 \\ q(x) & = & x^3 - 4.5x^2 + 6.7499x - 3.37485 \end{array}$$

$|6.75 - 6.7499| = 10^{-4}$ and $|3.375 - 3.37484| = 1.50 \cdot 10^{-4}$.
Adding up the errors, we find $|\Delta p| = 2.50 \cdot 10^{-4}$.

Now we compare: $|\Delta p| = 2.50 \cdot 10^{-4} \ll 10^{-2} = |\Delta r|$.

The magnitude of the error on the root (the output) is much larger than the magnitude of the error on the coefficients (the input).

sensitivity formula for roots

The equation $f(x) = 0$ has the root r: $f(r) = 0$.
Denote Δr: $\Delta r = r - \bar{r}$ or $r = \bar{r} + \Delta r$, Δr is the forward error.
Denote Δf: $\Delta f = f - \bar{f}$ or $f = \bar{f} + \Delta f$, Δf is the backward error.

$$(f + \Delta f)(r + \Delta r) = 0 \Leftrightarrow f(r + \Delta r) + \Delta f(r + \Delta r) = 0$$

We apply Taylor series, ignoring second order terms:

$$\begin{aligned} f(r + \Delta r) &= f(r) + f'(r)\Delta r + \cdots \\ \Delta f(r + \Delta r) &= \Delta f(r) + \Delta f'(r)\Delta r + \cdots \end{aligned}$$

Note that $f(r) = 0$ and $\Delta f'(r)\Delta r$ is of second order.

$$(f + \Delta f)(r + \Delta r) = 0 \Leftrightarrow f'(r)\Delta r + \Delta f(r) \approx 0.$$

Theorem (sensitiviy of a root)

For an equation $f(x) = 0$ with root r: $|\Delta r| \approx \left| \dfrac{\Delta f(r)}{f'(r)} \right|$.

Numerical Conditioning

The Wilkinson Polynomial

The roots of the Wilkinson polynomials are consecutive integers $1, 2, \ldots$. For example, the 20-th Wilkinson polynomial is

$$p(x) = (x - 1)(x - 2)\cdots(x - 20).$$

To test a root finder, this polynomial seems like an ideal test problem.

When expanded, the constant coefficient of p is
$20! = 2432902008176640000$
and $20! =$ `2.43290200817664e18` as a 64-bit float.

Evaluating p correctly is the main difficulty.

evaluating Wilkinson polynomials

```
julia> import Pkg; Pkg.add("SymPy")
julia> using SymPy
julia> x = Sym("x")
julia> w20 = prod([(x - Float64(r)) for r = 1:20])
julia> e20 = expand(w20)
```

Observe the conversion `Float64(r)` of the integer `r`.

```
julia> subs(w20, x => 1.0)
0
julia> subs(e20, x => 1.0)
-2560.00000000000
```

Exercise 1:
Do the above calculation for the other 19 roots
of the 20-th Wilkinson polynomial. What do you observe?

Numerical Conditioning

1. Forward and Backward Error
   - nearby solutions for nearby problems
2. Multiple Roots
   - where functions are very flat
3. The Wilkinson Polynomial
   - a historical example in numerical analysis
4. Condition Number
   - the error magnification factor

condition number

Every problem in numerical analysis has a condition number.

Definition

For a numerical problem with input x and output y,

the condition number κ is $\dfrac{|\Delta y|}{|y|} \leq \kappa \dfrac{|\Delta x|}{|x|}$.

The condition number measures how errors in the input are magnified to errors in the output.

A problem is

- *well-conditioned* if κ is small, and
- *ill-conditioned* if κ is large.

application to the root finding problem

For an equation $f(x) = 0$ with root r,
we derived the sensitiviy formula for the root:

$$|\Delta r| \approx \left| \frac{\Delta f(r)}{f'(r)} \right|.$$

To derive the condition number, divide both sides by $|r|$ and insert $|f|$:

$$\left| \frac{\Delta r}{r} \right| \approx \frac{1}{|f'(r)|} \left| \frac{\Delta f(r)}{r} \right| = \frac{|f|}{|r\,f'(r)|} \left| \frac{\Delta f(r)}{f} \right|,$$

where $|f|$ measures the size (of the coefficients) of f.

Theorem (condition of the root finding problem)

The equation $f(x) = 0$ with root r has condition number $\kappa = \dfrac{|f|}{|r\,f'(r)|}$.

interpretation of the condition number

For an equation $f(x) = 0$ with root r, the condition number is

$$\kappa = \frac{|f|}{|r f'(r)|}.$$

Three factors in the magnification of the backward error:

1. $|f|$ is the size of f, some examples are
   - the largest coefficient of f if a polynomial, or
   - the largest value $f(x)$ can take in the neighborhood of r.

   This factor captures the numerical difficulty of evaluating f.
2. $1/|r|$: the smaller the root, the larger the relative error.
3. $1/|f'(r)|$: if r is close to a multiple root, then $f'(r) \approx 0$.

the Wilkinson polynomials again

Exercise 2:

Consider $p(x) = \prod_{r=1}^{d} (x - r)$, for some finite degree d.

Evaluate the formula $\kappa = \dfrac{|p|}{|r\,p'(r)|}$ at the first root $r = 1$ of p,

for increasing values of the degree $d = 2, 3, \ldots, 20$,
using the absolute value of the largest coefficient of p
as the value for $|p|$.

For which value of d does κ become larger than 10^8?