Nonlinear Rootfinding (Introduction)

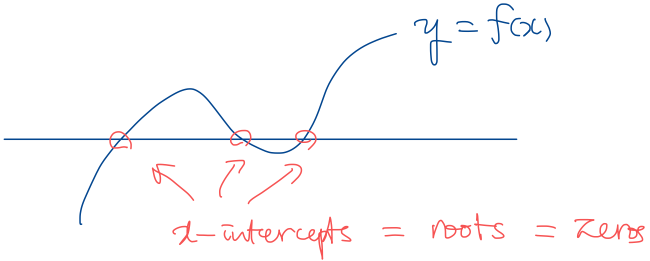
Introduction

Problem Statement

Rootfinding Problem

Given a continuous scalar function f of a scalar variable, find a real number r such that $f(r) = 0$.

- r is a **root** (zero) of the function f.
- The formulation $f(x) = 0$ is general enough; *e.g.*, to solve $g(x) = h(x)$, set $f = g - h$ and find a root of f.



Iterative Methods

- Unlike the earlier linear problems, the root cannot be produced in a finite number of operations.
- Rather, a sequence of approximations that formally converge to the root is pursued.

Iteration Strategy for Rootfinding. To find the root of f:

1. Start with an initial iterate, say x_0.
2. Generate a sequence of iterates $x_1, x_2, \ldots$ using an *iteration algorithm* of the form
$$x_{k+1} = g(x_k), \quad k = 0, 1, \ldots$$
3. Continue the iteration process until you find an x_i such that $f(x_i) = 0$. (In practice, continue until some member of the sequence seems to be "good enough".)

MATLAB's FZERO

`fzero` is MATLAB's general purpose rootfinding tool.

Syntax:

```
x_zero = fzero( <function>, <initial iterate> )
x_zero = fzero( <function>, <initial interval> )
[x_zero, fx_zero] = ....
```

root r

$f(r)$

(want it to be close to 0.)

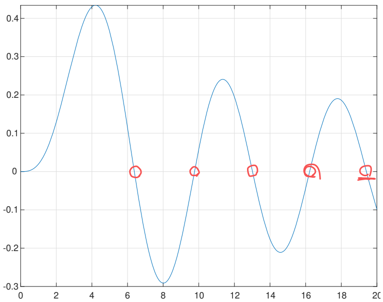
Example

Solve $x^2 \frac{d^2y}{dx^2} + x\frac{dy}{dx} + (x^2 - m^2)y = 0$

The roots of J_m, a Bessel function of the first kind, is found by

- Plot the function.
- Find approximate locations of roots.

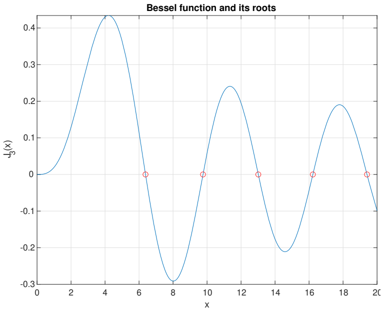
```
J3 = @(x) besselj(3,x);
fplot(J3,[0 20])
grid on
guess = [6,10,13,16,19];
```



Example (cont')

- Then use `fzero` to locate the roots:

```
omega = zeros(size(guess));
for j = 1:length(guess)
  omega(j) = fzero(J3,guess(j));
end
hold on
plot(omega,J3(omega),'ro')
```



Conditioning

Rootfinding [• input : a function f
• output : a number r (roots / zeros)

- Sensitivity of the rootfinding problem can be measured in terms of the condition number:

$$\text{" (absolute condition number)} = \frac{|\text{abs. error in output}|}{|\text{abs. error in input}|}, \quad = \frac{|\Delta r|}{|\epsilon\, g(\cdot)|}\text{"}$$

where, in the context of finding roots of f,

  - input: f (function)
  - output: r (root)

- Denote the changes by:

  # function — original — perturbed

  - error/change in input: ϵg, where $\epsilon > 0$ is small $\quad (f \mapsto f + \epsilon g)$
  - error/change in output: $\Delta r \quad (r \mapsto r + \Delta r)$

Recall Taylor series.

Let f be a smooth function (at x).

Then

$$f(x+h) = \frac{f(x)}{0!}h^0 + \frac{f'(x)}{1!}h^1 + \frac{f''(x)}{2!}h^2 + \cdots$$

center perturbation

$$= \sum_{k=0}^{\infty} \frac{f^{(k)}(x)}{k!} h^k$$

Note

$$f(x+h) = f(x) + f'(x)h + \underbrace{O(h^2)}_{\text{ignore all higher order terms}} \quad \text{as } h \to 0.$$

because $r+\Delta r$ is a root of $f+\epsilon g$.

- The *perturbed equation*

$f(r)+f'(r)\Delta r + O((\Delta r)^2)$

$$f(r+\Delta r)+\epsilon g(r+\Delta r)=0$$

$g(r)+g'(r)\Delta r + O((\Delta r)^2)$

is linearized to (Taylor expansion)

$$\cancel{f(r)}^{\,0}+f'(r)\Delta r+g(r)\epsilon+g'(r)\epsilon\Delta r\approx 0,$$

$(f'(r)+\epsilon g'(r))\Delta r \approx -\epsilon g(r)$

ignoring $O((\Delta r)^2)$ terms[1].

- Since $f(r)=0$, we solve for Δr to get

$$\Delta r\approx -\epsilon\frac{g(r)}{f'(r)+\epsilon g'(r)}\approx -\epsilon\frac{g(r)}{f'(r)},$$

for small ϵ compared with $f'(r)$.

[1] That is, terms involving $(\Delta r)^2$ and higher powers of Δr

Conditioning (cont')

$$\left|\frac{\Delta r}{\epsilon\, g(r)}\right| \approx \left|-\frac{\cancel{\epsilon}\, \cancel{g(r)}}{f'(r)}\right| \cdot \left|\frac{1}{\cancel{\epsilon}\, \cancel{g(r)}}\right| = \frac{1}{|f'(r)|}$$

- Therefore, the absolute condition number of the rootfinding problem is

$$\kappa_{f\mapsto r} = \frac{1}{|f'(r)|},$$

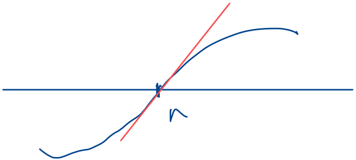
syn. ill-conditioned

  which implies that the problem is highly sensitive whenever $f'(r) \approx 0$.

- In other words, if $|f'|$ is small at the root, a computed *root estimate* may involve large errors.

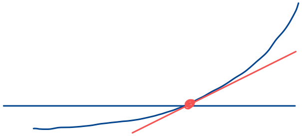
Illustration

$|f'(r)|$ is "mild"



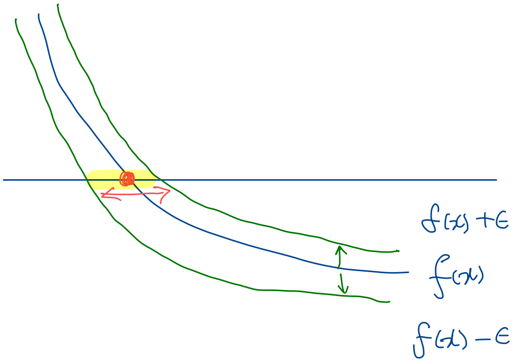
OK

$|f'(r)|$ is small.

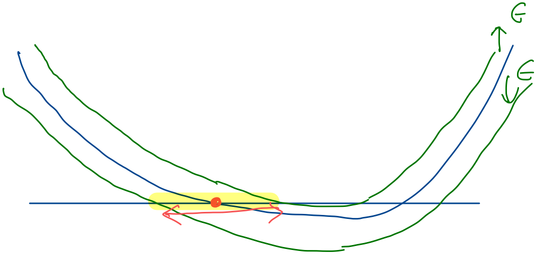


Bad

reasonably - conditioned



ill - conditioned



Residual and Backward Error

- Without knowing the exact root, we cannot compute the error.
- But the **residual** of a root estimate $\tilde{r}$ can be computed:

$$\text{(residual)} = f(\tilde{r}).$$

- Small residual *might* be associated with a small error.
- The residual $f(\tilde{r})$ is the *backward error* of the estimate.

Recall Least square.

$A\vec{x} \text{ "=" } \vec{b}$

$\text{residual} = A\vec{x} - \vec{b}$

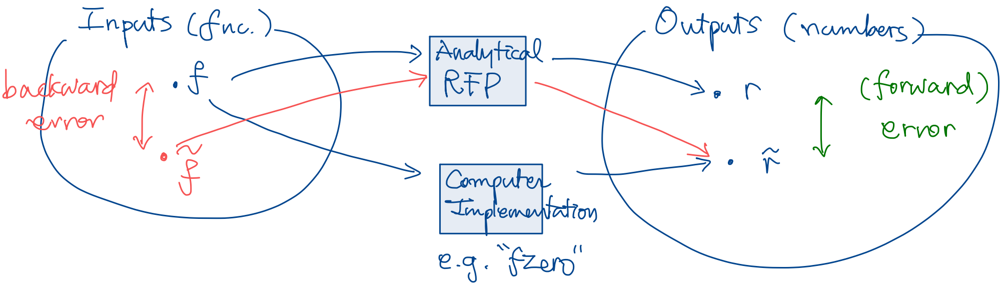
Root finding

Exact: $f(r) = 0$

Approx: $f(\tilde{r}) \approx 0$

$$\boxed{\begin{aligned}\text{residual} &= f(\tilde{r}) - 0 \\ &= f(\tilde{r})\end{aligned}}$$

Backward error



Note that $\tilde f(x)$ defined as

$$\tilde f(x) = f(x) - f(\hat r)$$

will do.

why?

$$\tilde f(\hat r) = f(\hat r) - f(\hat r) = 0$$

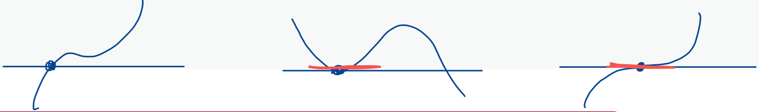
So why (back. error) = (residual) ?

$$(\text{back. error}) = f(x) - \hat{f}(x)$$

$$= f(x) - \left(f(x) - f(\tilde{r}) \right)$$

$$= f(\tilde{r}) = (\text{residual})$$

Multiple Roots



Definition 1 (Multiplicity of Roots)

Assume that r is a root of the differentiable function f. Then if

$$0 = f(r) = f'(r) = \cdots = f^{(m-1)}(r) \quad \text{but} \quad f^{(m)}(r) \neq 0,$$

we say that f has a root of **multiplicity** m at r.

- We say that f has a **multiple root** at r if the multiplicity is greater than 1.
- A root is called **simple** if its multiplicity is 1.
- If r is a multiple root, the condition number is infinite.

$$\kappa = \frac{1}{|f'(r)|}$$

- Even if r is a simple root, we expect difficulty in numerical computation if $f'(r) \approx 0$.