SFU
muraki

MONDAY

- COMPUTING REPORT DUE TODAY (CROWDMARK attach script)
- WORKSHOP OPEN 2^{30}
- MIDTERM DATE: FRI 01 MARCH (after reading week)
- classroom emergency procedures

LAST DAY

- CONTINUITY, INTERMEDIATE VALUE THM & BRACKETTING.
- BISECTION algorithm on a sign-change interval
  - interval (x_L, x_R) shrinks by every iteration
  - convergence rate $(\log_{10}(x_R - x_L)$ vs $n)$ decreased
- SECANT METHOD
  - 2 initial guesses x_0 & x_2
  - update is x-intercept of SECANT LINE

$$p_n = p_{n-1} - \frac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f(p_{n-1}) - f(p_{n-2})}. \tag{2.12}$$

This technique is called the **Secant method** and is presented in Algorithm 2.4. (See Figure 2.10.) Starting with the two initial approximations p_0 and p_1, the approximation p_2 is the x-intercept of the line joining $(p_0, f(p_0))$ and $(p_1, f(p_1))$. The approximation p_3 is the

- no bracketting, CONVERGES for guesses but guarantee of convergence
- convergence is than

SFU SAFETY & RISK SERVICES

Classroom Safety Script

Please review the following safety information with your class at the beginning of each semester.

Reporting an Emergency

In an emergency call 911.

- If possible, call 911 from a campus phone and Campus Security will be automatically notified of the call location. A phone is located in every classroom. *<Indicate where the phone is located>*

- Urgent support is provided 24/7 by Campus Security at all three campuses. Campus Security provides emergency response and support, first aid and helps guide emergency responders to the right place on campus quickly.

Emergency Line (Urgent Security/ First Aid)	**778-782-4500**
Non-Emergency Line (Security/ Safe Walk)	**778-782-7991**

- This information is noted in several places in the classroom including: on campus phones, on blue "Assistance Phone" signage, and on classroom emergency posters *<indicate the location of these items>*
- Please also take a moment to add these numbers to your mobile devices.

Evacuation Procedures

- Familiarize yourself with the evacuation maps and assembly locations posted on the walls all over campus. Burnaby campus assembly areas are can also be found at: www.sfu.ca/emergency.
- If we need to evacuate this room, the exit routes are *<indicate where exit route are located>*
- We will meet outside at this location's assembly point *<indicate where the class would meet once outside>*

Emergency Notifications

- In an emergency, visit www.sfu.ca and follow @SFU on Twitter.
- Download the SFU snap app and enable push notifications to receive messages on your mobile device: www.sfu.ca/sfualerts
- These communication channels are also used in the event of class cancellation due to an emergency

Safe Walk

- Campus Security is available to escort you anywhere on campus 24/7. Contact Campus Security's non-emergency line (778-782-7991) to request a Safe Walk.

See Something, Say Something

- Report suspicious persons or objects to Campus Security immediately. Safety is a shared responsibility.

Hazard Specific Procedures

- Review the emergency procedures poster in our classroom that detail specific procedures in the event of a fire, earthquake, active threat, cardiac arrest, severe weather event, or a hazardous outdoor environment.
- AEDs have been installed throughout all three campuses. AED campus locations can be found at www.sfu.ca/aed. *<indicate where the nearest AEDs are located>*

For more information visit:
www.sfu.ca/emergency

SFU
muraki

COMPUTING REPORT #3

- sequence of roots by

$$f(x_j, \quad) = 0 \rightarrow (\quad)$$

- FOUR scripts: CONTOUR FOLLOWING. CODE (fzero)
  BiSsqrt.m & SMsqrt.m
  topographic function

CONVERGENCE RATE &

- HOW DOES ITERATION ?
- BISECTION (p 51-52)
  - bracketted root. $x_L^0 < \quad < x_R^0$

exact zero

$$|x_B^1 - \quad| < \frac{x_R^0 - x_L^0}{2} \quad \text{after } 1^{st} \text{ bisection,}$$

$$|x_B^k - \quad| < \frac{x_R^{k-1} - x_L^{k-1}}{2} \quad \text{" } k^{th} \text{ "}$$

SFU
muraki

and since $x_R - x_L$ decreases by $\frac{1}{}$ each iteration

$$|x_R^k - x_e| < \left(\frac{}{}\right)^{k} (x_R - x_L)$$

$$= (\ \ ) \cdot 10^{(\ \ \ )k}$$

exponential decay of error

· an additional decimal place of accuracy for every $1/.301 \approx$. iterations

CONVERGENCE ORDER (p78)

Order of Convergence

Suppose $\{p_n\}_{n=0}^{\infty}$ is a sequence that converges to p, with $p_n \neq p$ for all n. If positive constants λ and α exist with

$$\lim_{n\to\infty} \frac{|p_{n+1} - p|}{|p_n - p|^{\alpha}} = \lambda,$$

then $\{p_n\}_{n=0}^{\infty}$ **converges to p of order α, with asymptotic error constant λ.** ■

An iterative technique of the form $p_n = g(p_{n-1})$ is said to be of *order* α if the sequence $\{p_n\}_{n=0}^{\infty}$ converges to the solution $p = g(p)$ of order α.

In general, a sequence with a high order of convergence converges more rapidly than a sequence with a lower order. The asymptotic constant affects the speed of convergence but not to the extent of the order. Two cases of order are given special attention.

(i) If $\alpha = 1$ (and $\lambda < 1$), the sequence is **linearly convergent**. BIS ($\lambda =$)

(ii) If $\alpha = 2$, the sequence is **quadratically convergent**. NM

→ basically, $\dfrac{|\text{error}_{n+1}|}{|\text{error}_n|^{\alpha}} \to$ constant

SFU
muraki

- BiS summary
  - ACCURACY: set by (< 10^{-14} rel tol gen achievable)
  - EFFICIENCY: roughly . iterations (funct.) per decimal place of accuracy
  - ROBUSTNESS: sign-change interval + continuity root.

MATHEMATICS of CONVERGENCE for NEWTON'S METHOD .

Theorem 2.6 Let $f \in C^2[a,b]$. If $p \in (a,b)$ is such that $f(p) = 0$ and $f'(p) \neq 0$, then there exists a $\delta > 0$ such that Newton's method generates a sequence $\{p_n\}_{n=1}^{\infty}$ converging to p for any initial approximation $p_0 \in [p-\delta, p+\delta]$. ■

↳ there is a range of · near x_e

$x_e - \leq x_0 \leq x_e +$

gives a NM.

- convergence rate using (NOT in BF)

$$x_{n+1} = x_n - f(x_n) \big/ f(x_{n-1})$$

- exact $f(\ ) = 0$ and $E_n = \ -$

muraki

· convert to update

$$+ \quad = \quad + \quad - f(x_e + \quad) \big/ f'(x_e + \quad)$$

· now use Taylor approx for small

$$f(x_e+\epsilon_n) \simeq f(x_e) + f(\quad)\epsilon_n + -f(\quad)\epsilon_n^2$$

$$f'(x_e+\epsilon_n) \simeq \quad f(\quad) + f(\quad)\epsilon_n$$

$$\epsilon_{n+1} \simeq \epsilon_n - \epsilon_n \cdot \frac{+\ -}{+\quad '}$$

common denom.

$$= \frac{\frac{1}{2} f_e'' \epsilon_n^2}{f_e'\left(1 + f_e''/f_e' \cdot \epsilon_n\right)}$$

$$\epsilon_{n+1} \simeq \frac{}{} \cdot \epsilon_n$$

conv!

$$\left(\quad \epsilon_{n+1} \right) \simeq \left(\quad \epsilon_n \right)$$

when <

SFU
muraki

- note: $\lambda_e = \frac{f_e''}{2f_e'} = \frac{f''(x_e)}{2f'(x_e)}$ depends only on function at

- error in sign when λ_e
- of E_n depends on being small
  - extra challenge for f_e OR f_e

- NM summary
  - ACCURACY: set by ($< 10^{-14}$ rel accur achievable)
  - EFFICIENCY: convergence (doubling) BUT funct. per iteration.
  - ROBUSTNESS: convergence to

CONVERGENCE for SECANT METHOD

- can obtain error update as done for NM.

$$(\lambda_e E_{n+1}) \simeq (\lambda_e \quad)(\lambda_e \quad)$$

because SM is - method

SFU
muraki

$$\log|\lambda_e E_{n+1}| = \log|\lambda_e E_n| + \log|\lambda_e E_{n-1}|$$

λ

- Fibonacci limit

$$\lim_{n\to\infty} \frac{\log|\lambda_e E_{n+1}|}{\log|\lambda_e E_n|} = \qquad = \underline{\qquad}$$

$$\log|\lambda_e E_{n+1}| \simeq \log|\lambda_e E_n|$$

better than !

OVERALL SUMMARY

	ACC / EFF	ROB	
BiS	Conv eval/iter	preserve	
NM	Conv eval/iter	need / $\lvert<1$	need
SM	Conv eval/iter	need / $\lvert<1$	

SFU
muraki

Fzero

Matlab's built-in command is `fzero(fun,x0,options)`.

- Finds a root of a function `fun` near or within `x0`.
- `x0` can be a point or an interval.
- `options` allows one to set various options, e.g. tolerances

Remarks:

1. Based on a hybrid method combining **bisection**, **secant** and **inverse quadratic interpolation**.
2. Cannot locate double roots – requires a sign change.
3. Proceeds roughly as follows:
   (i) If `x0` is a number, find an interval `[x0,x1]` with a sign change.
   (ii) Run secant/inverse quadratic interpolation.
   (iii) If diverging or not converging quickly, switch to a few steps of bisection, and then re-run secant/inverse quadratic interpolation.

Example

```
>> f = @(x) tan(x) - 1./x;
>> opts = optimset('TolX',1e-12);
>> [p,fval,exitflag,output] = fzero(f,1,opts)

p =

    0.8603


fval =

  -2.6579e-13


exitflag =

     1


output =

  struct with fields:

    intervaliterations: 6
            iterations: 5
             funcCount: 17
             algorithm: 'bisection, interpolation'
               message: 'Zero found in the interval [0.84, 1.11314]'
```

Example

```
>> opts = optimset('TolX',1e-12,'Display','iter');
>> fzero(f,1,opts)

Search for an interval around 1 containing a sign change:
 Func-count    a          f(a)             b          f(b)        Procedure
    1               1       0.557408            1       0.557408        initial interval
    3        0.971716       0.435476      1.02828       0.686295        search
    5            0.96       0.386691         1.04       0.742076        search
    7        0.943431       0.319185      1.05657       0.823694        search
    9            0.92       0.226307         1.08       0.945291        search
   11        0.886863      0.0991413      1.11314        1.13194        search
   12            0.84     -0.0748438      1.11314        1.13194        search

Search for a zero in the interval [0.84, 1.11314]:
 Func-count    x          f(x)             Procedure
   12            0.84     -0.0748438        initial
   13         0.85694     -0.0125512        interpolation
   14        0.860315   -6.72446e-05        interpolation
   15        0.860334   -5.79774e-08        interpolation
   16        0.860334   -2.65787e-13        interpolation
   17        0.860334   -2.65787e-13        interpolation

Zero found in the interval [0.84, 1.11314]

ans =

    0.8603
```

Remarks: In numerical computing, it is often useful to use the full output mode of a built-in command. This allows one to understand failures, check efficiency, etc.