Numerical Analysis

- Hand in HW
- problem w/ emailed HW
  Dropbox ?
  Moodle ?

- Take a few min and discuss ... How did these assignments go?
  Questions ? LaTex ? FreeMat ?

<u>Today:</u> We start ROOTFINDING.

Brainstorm ... How could we write computer code to find $f(x) = 0$?

- might think ... lets just go through the function in a loop ...

```
for i = 1:N
    x = x + Δx
    value = f(x)
```

A. ⟶ if $f(x)$ goes from pos to neg then root is approx between x and $x - \Delta x$

OR

B. ⟶ if $f(x)$ is within ϵ of zero root is at x.

```
end.
```

Problems with these ... plots ... single precision.

.m file on Moodle.

- On your laptop ... plot $y = (x-1)^3$ for $0 \le x \le 2$
- "grid on" puts a grid behind your function.

NOISE FROM ROUNDING – ESPECIALLY WHEN SINGLE PRECISION
LOTS OF POINTS CLOSE TO ZERO
ASYMPTOTIC FUNCTIONS that approach zero.

So we need to be more sneaky !!

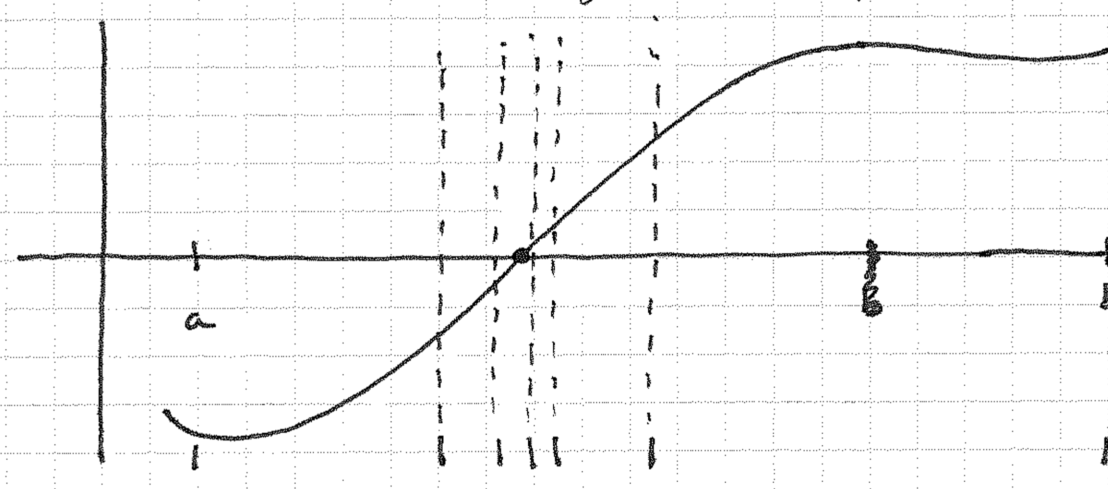
3.1 The BISECTION METHOD.

Assume $f(x)$ is a function that is real-valued and x is a real variable. Suppose $f(x)$ is continuous on an interval $a \le x \le b$ and that

$$f(a)f(b) < 0$$

Then $f(x)$ changes sign on $[a, b]$ and $f(x) = 0$ has at least one root on the interval.

The Bisection method repeatedly halves the interval $[a, b]$ and focuses on the half where $f(x)$ goes from pos to neg.



- Consider ½ distance between a and b
  $$c = (a+b)/2$$
- Check that $b - c \le \epsilon$ if it is then our distance is less than error tolerance and we accept c as the root.
- Choose which side still goes fro + to −
  if $s(f(b)) \cdot s(f(c)) \le 0$ then $a = c$
  otherwise $b = c$ and go again.

here test the sign to avoid underflow.

OVERFLOW ERRORS ... #'s to large for the computation → Fatal NaN.

UNDERFLOW ERRORS ... # to small rounded to zero → More Sneaky.

Be careful when multiplying small #'s.

In Bisection $f(b) \cdot f(c) \leq 0$
when we are very close two positive #'s could be rounded to zero!

<u>Ex</u> Use the Bisection Method and a regular calculator to find the largest root of

$$f(x) = x^6 - x - 1 = 0$$

accurate to within $\epsilon = 0.0001$.

#1 we need to choose $[a, b]$ ...

USE FREEMAT

PLOT $f(x)$ and find where the largest root is:

$$a < \alpha < b$$

#2 Check that $f(a) \cdot f(b) \leq 0$

#3 Start the Bisection Method ...
Do this by hand and write in words what steps you have to take..
This will be your PSUEDO CODE.

how do we check error?

ERROR BOUNDS ...

let a_n, b_n and c_n be the n^{th} computed values of a, b, and c.

then $$a_{n+1} - b_{n+1} = \frac{1}{2}(b_n - a_n) \qquad n \geq 1$$

and

$$b_n - a_n = \frac{1}{2^{n-1}}(b-a) \qquad n \geq 1$$

since the root α is either in $[a_n, c_n]$ or $[c_n, b_n]$

we know that the error

$$|\alpha - c_n| \leq c_n - a_n = b_n - c_n = \frac{1}{2}(b_n - a_n)$$

So we can write the error bound

$$|\alpha - c_n| \leq \frac{1}{2^n}(b-a).$$

Q. What happens as $n \to \infty$?

Q. What does this mean about the Bisection Method and Convergence?

Use this idea to see how many iterations will be necessary ...

we want $|\alpha - c_n| \leq \epsilon$

$$\frac{1}{2^n}(b-a) \leq \epsilon$$

solve for n...

$$n \geq \frac{\log\left(\frac{b-a}{\epsilon}\right)}{\log(2)}$$

HOW MANY ITERATIONS DID WE NEED FOR OUR EXAMPLE?