Bisection method: main idea

Monday, August 31, 2020 10:21 PM

Bisection method (aka binary search)

Main idea in pictures*

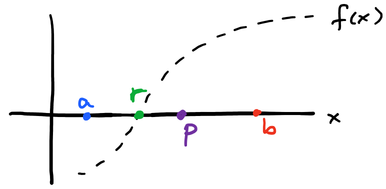
* I'm purposefully not giving pseudocode. Better to understand the idea.

- Suppose we have a, b such that $\text{sign}(f(a)) = -\text{sign}(f(b))$

  "opposite signs" i.e., $f(a) < 0$, $f(b) > 0$ or vice-versa.

  A slick way to encode this is to say $f(a) \cdot f(b) < 0$

  then by the I.V.T. (recall we assume f is continuous in this chapter), we know there is a root $r \in (a, b)$



- If we guess the midpoint $p = \frac{a+b}{2}$,

  then our error $|r - p|$ is at most half the width of the interval, $\frac{b-a}{2}$

- (*Main Idea) Recurse: $a_1 = a$, $b_1 = b$

  let's find a smaller interval $[a_2, b_2]$

  Evaluate f at the midpoint p

  If $f(a) \cdot f(p) < 0$ (i.e. have opposite sign)

  $a_2 = a_1$
  $b_2 = p$ $\quad$ p becomes the new right endpoint

  Else if $f(b) \cdot f(p) < 0$

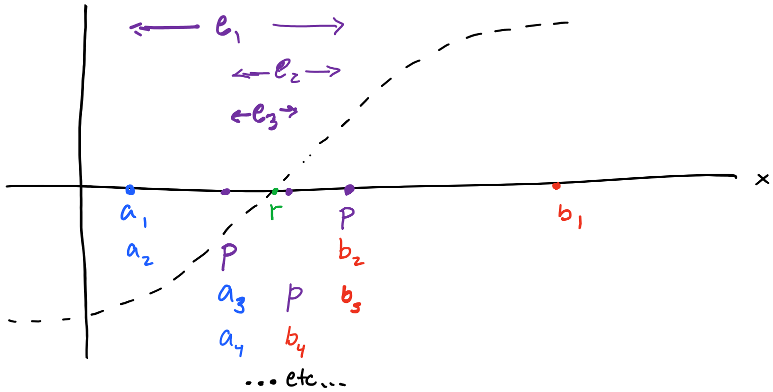
  $a_2 = p$
  $b_2 = b_1$ $\quad$ p becomes the new left endpoint

  Else

  ... this can't happen ...

Once we recurse, and take a new midpoint, our error is at most $\frac{1}{2}$ the width of our new interval: $\frac{b_2 - a_2}{2}$

... and observe this is half the error bound from the previous step.



Limitations of the method:

- you need initial knowledge of $[a, b]$
- doesn't converge as quickly as other methods (like Newton's Method)
- doesn't easily extend to higher dimensions (whereas Newton's method does)
- requiring $f(a)$, $f(b)$ to have opposite signs guarantees a root, but it's not necessary to have a root. So bisection doesn't always apply

Ex: $f(x) = x^2$ has a root at $x=0$ but never changes sign.

biggest drawbacks

Benefits:

- doesn't need user to supply derivative f' (in fact, doesn't even require f to be differentiable)
- Converges "fast enough" usually

Next lecture... rate of convergence of the bisection method.