Announcements

Homework 11

Sign-up for exam 3

Goals

- Wacky methods in 1D
- nD
  - ↳ Newton
  - ↳ Secant ?

Review

- FPI $\quad x_{k+1} = g(x_k)$

  $|g'(x^*)| < 1$

  $g'(x^*) = 0 \Rightarrow$ quadratic conv.

- Newton $\quad x_{k+1} = \underbrace{x_k - \frac{f(x_k)}{f'(x_k)}}_{g(x_k)}$

Improving on Newton?

How would we do "Newton + 1" (i.e. even faster, even better)?

Improving on Newton?

How would we do "Newton + 1" (i.e. even faster, even better)?

Easy:

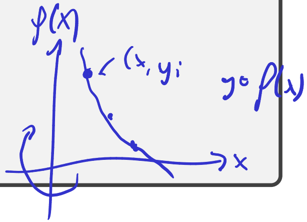
- Use second order Taylor approximation, solve resulting quadratic
- Get cubic convergence!
- Get a method that's *extremely* fast and *extremely* brittle
- Need second derivatives
- What if the quadratic has no solution?

Root Finding with Interpolants

Secant method uses a linear interpolant based on points $f(x_k)$, $f(x_{k-1})$, could use more points and higher-order interpolant:

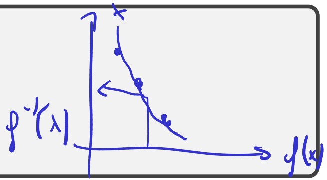
Muller's method

↳ find complex root



What about existence of roots in that case?

- Fit a quadratic to f^{-1}
- Find $f^{-1}(0)$
- Inverse quadratic interpolation



Finding all roots?

p_n poly of degree n

$$p_{n-1}(x) = \frac{p_n(x)}{(x - x^*)}$$

x^*: approximate root

"Deflation"

Achieving Global Convergence

The linear approximations in Newton and Secant are only good locally. How could we use that?

- "Hybrid" method: bisection first, switch to Newton when "close".
- "trust region" methods: e.g. use bracket from bis

Achieving Global Convergence

The linear approximations in Newton and Secant are only good locally.
How could we use that?

- Hybrid methods: bisection + Newton
  - Stop if Newton leaves bracket
- Fix a region where they're 'trustworthy' (trust region methods)
- Limit step size
- Sufficient conditions for convergence of Newton (under *strong* assumptions) are available.

Fixed Point Iteration

$$\vec{g}(\vec{x}+\vec{h}) = \vec{g}(\vec{x}) + J_{\vec{g}}(\vec{x})\vec{h} + O(\|\vec{h}\|^2)$$

$$\begin{aligned} \boldsymbol{x}_0 &= \langle\text{starting guess}\rangle \\ \boldsymbol{x}_{k+1} &= \boldsymbol{g}(\boldsymbol{x}_k) \end{aligned}$$

$e_{k+1} = x_{k+1} - x^* = g(x_k) - g(x^*)$

$= g'(x^*)(x_k - x^*) + O(|x_k - x^*|^2)$

When does this converge?

Criterion for convergence $\|J_g(x^*)\| < 1$.

$|g'(x^*)| < 1$

$$J_g(x) = \begin{bmatrix} \partial_1 g_1 & \cdots & \partial_n g_1 \\ \vdots & & \\ \partial_1 g_n & & \partial_n g_n \end{bmatrix} \qquad \vec{g} = \begin{pmatrix} g_1 \\ \vdots \\ g_n \end{pmatrix}$$

For a given matrix A, there exists a norm $\|\cdot\|_A$ so that $\rho(A) \le \|A\|_A \le \rho(A) + \varepsilon$

A sharper criterion: $\rho(A) < 1$.

Newton's Method

What does Newton's method look like in n dimensions?

$$\vec{f}(\vec{x}+\vec{s}) \approx \vec{f}(\vec{x}) + J_{\vec{f}}(\vec{x})\vec{s} = 0$$

$$J_f(\vec{x})\vec{s} = -f(x)$$

$$\vec{s} = -J_f^{-1}(\vec{x})\, f(\vec{x})$$

$$\vec{x}_{k+1} = \vec{x}_k - J_{\vec{f}}(\vec{x}_k)^{-1} f(\vec{x})$$

Downsides of n-dim. Newton?

- need J_f (a lot of data!)
- LU every step

Demo: Newton's method in n dimensions [cleared]

Secant in n dimensions?

What would the secant method look like in n dimensions?

$$f(\vec{x}_k), \quad f(\vec{x}_{k+1}) \rightsquigarrow J_f(\vec{x})\ ?$$

$f(\vec{x}_k) \in \mathbb{R}^n$, $f(\vec{x}_{k+1}) \in \mathbb{R}^n$, $J_f \in \mathbb{R}^{n\times n}$ $\to \tilde{J}$

Idea: maintain a guess of J_f, update it from $f(\vec{x}_k)$, $f(\vec{x}_{k+1})$, ...

require
$$\tilde{J}(\vec{x}_{k+1} - \vec{x}_k) = \vec{f}(\vec{x}_{k+1}) - \vec{f}(\vec{x}_k)$$

Broyden's method:
- update $\tilde{J}$ with a rank-1 update
- keep a guess of $\tilde{J}^{-1}$ instead
  ↪ update via Sherman-Morrison → 'good Broyden' / 'bad Broyden'

Numerically Testing Derivatives

Getting derivatives right is important. How can I test/debug them?

Pick $h>0$, $\vec{s} \in \mathbb{R}^n$ with $\|\vec{s}\|_2 = 1$.

$$\left\| \frac{f(x+h\vec{s}) - f(\vec{x})}{h} - J_f(\vec{x})\,\vec{s} \right\| = O(h)$$

Plug in many h, error should decrease $(h \to 0)$ linearly in h.

$$f(\vec{x}+h\vec{s}) = f(\vec{x}) + J_f(\vec{x})(h\vec{s}) + O(h^2)$$

I_h-class: non linear

Outline

Optimization: Problem Statement

Have: Objective function $f : \mathbb{R}^n \to \mathbb{R}$
Want: Minimizer $\boldsymbol{x}^* \in \mathbb{R}^n$ so that

$$f(\boldsymbol{x}^*) = \min_{\boldsymbol{x}} f(\boldsymbol{x}) \quad \text{subject to} \quad \boldsymbol{g}(\boldsymbol{x}) = 0 \quad \text{and} \quad \boldsymbol{h}(\boldsymbol{x}) \leq 0.$$

- $\boldsymbol{g}(\boldsymbol{x}) = 0$ and $\boldsymbol{h}(\boldsymbol{x}) \leq 0$ are called constraints.
  They define the set of feasible points $\boldsymbol{x} \in S \subseteq \mathbb{R}^n$.
- If $\boldsymbol{g}$ or $\boldsymbol{h}$ are present, this is constrained optimization.
  Otherwise unconstrained optimization.
- If $\boldsymbol{f}$, $\boldsymbol{g}$, $\boldsymbol{h}$ are *linear*, this is called linear programming.
  Otherwise nonlinear programming.

not here

Optimization: Observations

Q: What if we are looking for a *maximizer* not a minimizer? $-f$

Give some examples:

- fastest / cheapest
- Solve $\vec{z}(\vec{x}) = \vec{0}$ "as well as possible"
  - $\rightsquigarrow$ Minimize $\|\vec{z}(\vec{x})\|_2 = f(\vec{x})$

What about multiple objectives?

- Pareto optimality
- combine into one (that's us!)

Existence/Uniqueness

Terminology: global minimum / local minimum

Under what conditions on f can we say something about existence/uniqueness?

If $f : S \to \mathbb{R}$ is continuous on a closed and bounded set $S \subseteq \mathbb{R}^n$, then

$f : S \to \mathbb{R}$ is called *coercive* on $S \subseteq \mathbb{R}^n$ if

If f is coercive and continuous and S is closed, ...