Grading:

- Will drop lowest exam
  lowest homework
  two lowest quizzes.

$$\lim_{k\to\infty} \frac{\|e_{k+1}\|}{\|e_k\|^r} = C \begin{matrix} <\infty \\ >0 \end{matrix}$$

x_0

$x_{k+1} = g(x_k)$

Let x^* be the fixed point.

sufficient conditions:

- $|g'(x^*)| < 1$
  suff. for locally linear conv.
- $g'(x^*) = 0$
  suff. for locally quadratic convergence

Newton's Method

Derive Newton's method.

$$\tilde{f}_k(x_k + h) = f(x_k) + f'(x_k)h = 0$$

For which h is $\tilde{f}_k(x_k + h) = 0$? $\Leftarrow$ $h = -\frac{f(x_k)}{f'(x_k)}$

$x_0 =$ (starting guess)

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

Demo: Newton's method [cleared]

Convergence and Properties of Newton

What's the rate of convergence of Newton's method? (assume root of mult. 1 for now)

$$g'(x) = \frac{f(x) f''(x)}{f'(x)^2}$$

at fixed point x^*, i.e. root of the function, $g'(x) = 0$

$\Rightarrow$ locally quadratic convergence.

Drawbacks of Newton?

- local convergence
- need derivative

Demo: Convergence of Newton's Method [cleared]

Secant Method

What would Newton without the use of the derivative look like?

$$f'(x_n) \approx \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}$$

$x_{k+1} = g(x_k)$

So

$x_0 = $ (starting guess)

$x_1 = $ (another starting guess)

$$x_{k+1} = x_k - \frac{f(x_k)}{\frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}}$$

Convergence of Properties of Secant

Rate of convergence is $\left(1+\sqrt{5}\right)/2 \approx 1.618$. (proof)

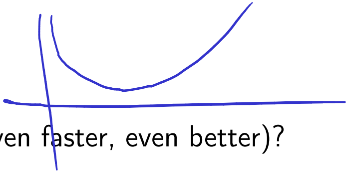
Drawbacks of Secant?

- Slower than Newton
- Local convergence
- second starting guess

Demo: Secant Method [cleared]
Demo: Convergence of the Secant Method [cleared]

Secant (and similar methods) are called Quasi-Newton Methods.

Improving on Newton?



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

- Use quadratic approximation instead of linear
- needs two derivatives
- cubic convergence, but: even better starting guess needed
- complex iterates?

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:

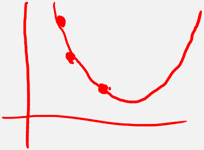
(or a subset)

- Can fit a polynomial to $(x_0, f(x_0)) \ldots (x_k, f(x_k))$
- Solve for a root of that → that's the next iterate.
- Fit a quadratic to the last three: Muller's method
  - somewhat useful for root finding

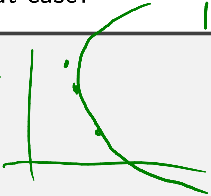
What about existence of roots in that case?

Inverse quadratic interpolation

NOT: intp. f



INSTEAD: intp. f^{-1}



next guess: $\tilde{f}^{-1}(0)$

Achieving Global Convergence

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

- Hybrid method, e.g. bisection + Newton
  maybe: stop if Newton leaves the bracket
- Fix a region where you "trust" the quadratically conv. method

"trust region methods"

- Sufficient cond. for Newton conv. exist.

Fixed Point Iteration

$0 < \quad < 1$

$\|\vec{g}(\vec{x}) - \vec{g}(\vec{y})\| \le \lambda \|\vec{x} - \vec{y}\|$

$\approx J_{\vec{g}}(\vec{\theta}) \cdot (\vec{x} - \vec{y})$

$$\begin{aligned} \boldsymbol{x}_0 &= \langle \text{starting guess} \rangle \\ \boldsymbol{x}_{k+1} &= \boldsymbol{g}(\boldsymbol{x}_k) = [g_1 \ \dots \ g_n] \end{aligned}$$

When does this converge?

$$J_{\vec{g}}(x^*) = \begin{bmatrix} \frac{\partial}{\partial x_1} g_1 & & \frac{\partial}{\partial x_n} g_1 \\ \vdots & & \vdots \\ \frac{\partial}{\partial x_1} g_n & \dots & \end{bmatrix}$$

$\|J_{\vec{g}}(x^*)\| < 1$ is sufficient.

Thm: For any matrix A, and any $\varepsilon > 0$, there exists a norm $\|\cdot\|_A$ such that

$\rho(A) \le \|A\|_A \le \rho(A) + \varepsilon$.

spectral radius: biggest eigenvalue by mag.

Actual criterion:

$$\rho(\partial g(x^*)) < 1.$$

Newton's Method

$h: \mathbb{R}^n \to \mathbb{R}$

$f: \mathbb{R}^n \to \mathbb{R}^n$

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

Look for an s so that $\hat{f}_k(\vec{x}_k + \vec{s}) = 0$

$$\tilde{\vec{f}}_k(\vec{x}_k + \vec{s}) = \vec{f}(\vec{x}_k) + J_f(x_k) \cdot \vec{s}$$

first two terms of Taylor, remainder terms of second order in $\|s\|$.

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

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

$\vec{x}_0 = $ (starting guess)

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

Downsides of n-dim. Newton?

- locally conv.
- need deriv.

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

Secant in n dimensions?

What would the secant method look like in n dimensions?