Numerical Differentiation

Caution: It is basically ill-conditioned process (for small h)

Forward difference quotient: $f'(x) \approx \dfrac{f(x+h)-f(x)}{h} = D_h^+ f(x)$, error: $\dfrac{f''(\xi)}{2}h = O(h)$

proportional to h

1st order approximation

$h>0$

error: by Taylor, $f(x+h) = f(x) + f'(x)\cdot h + \dfrac{f''(\xi)}{2}h^2$

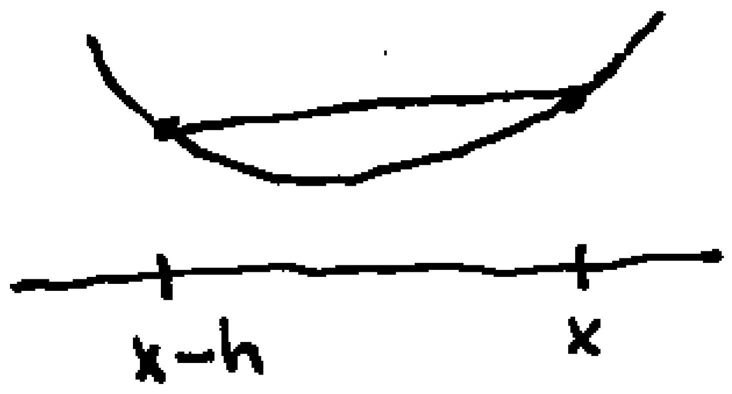
for some ξ between x, $x+h$

assuming $f \in C^2$

$$\Rightarrow \frac{f(x+h)-f(x)}{h} = f'(x) + \frac{f''(\xi)}{2}\cdot h$$

$\therefore$ error $= \dfrac{f''(\xi)}{2}\cdot h = O(h)$ as $h \to 0$, 1st order

Backward difference quotient: $f'(x) \approx \dfrac{f(x)-f(x-h)}{h} = D_h^- f(x)$ with error $\dfrac{f''(\xi)}{2}\cdot h = O(h)$



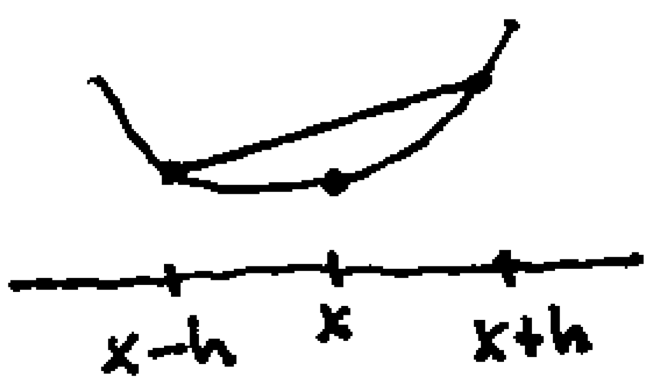
error: $f(x-h) = f(x) - f'(x)\cdot h + \dfrac{f''(\xi)}{2}\cdot h^2$

$$\Rightarrow \frac{f(x)-f(x-h)}{h} = f'(x) - \frac{f''(\xi)}{2}\cdot h$$

$\therefore$ error $= +\dfrac{f''(\xi)}{2}\cdot h = O(h)$ as $h \to 0$, 1st order approximation

Centered difference quotient: $f'(x) \approx \dfrac{f(x+h)-f(x-h)}{2h} = D_h f(x)$, error $-\dfrac{f'''(\xi)}{6}\cdot h^2 = O(h^2)$

2nd order



$$\text{error} = f'(x) - D_h f(x) =$$

$$= f'(x) - \frac{1}{2h}\Big[f(x) + f'(x)\cdot h + \frac{f''(x)}{2}h^2 + \frac{f'''(\xi_1)}{6}h^3$$
$$- f(x) + f'(x)h - \frac{f''(x)}{2}h^2 + \frac{f'''(\xi_2)}{6}h^3\Big]$$

$$= f'(x) - \frac{1}{\cancel{2h}}\left[\cancel{2h}\, f'(x)\right] - \frac{1}{6}\,\frac{f'''(\xi_1)+f'''(\xi_2)}{2}\cdot h^2$$

$$= -\frac{1}{6}f'''(\xi)\cdot h^2 \quad \text{for } \xi \in (x-h, x+h), \text{ assuming } f \in C^3$$

$= O(h^2)$, 2nd order, much better!

Much preferable whenever 2-sided values of f are known.

Landau Big O symbol:

$f(x) = \mathcal{O}(g(x))$ as $x \to x_0$ means $\dfrac{|f(x)|}{|g(x)|} \le C$ as $x \to x_0$ (near x_0)

$|f(x)| \le C \cdot |g(x)|$ for x near x_0

$|f(x)|$ proportional to $|g(x)|$ near x_0

Centered difference quotient for 2^{nd} derivative

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + \mathcal{O}(h^2) \text{ as } h \to 0$$

of 2^{nd} order

Basic methods for approximating derivatives:

Given values $f_1, f_2, \dots, f_n$ of $f(x)$ at nodes $x_1, x_2, \dots, x_n$ find $f'(x_i)$

1. Finite difference quotients (piecewise linear interpolation)

   e.g. $f'(x_i) \approx \dfrac{f(x_{i+1}) - f(x_{i-1})}{x_{i+1} - x_{i-1}}$ of order h^2

2. If values are reliable, approximate f by cubic spline interpolate near x_i and find its derivative at x_i

3. If values are unreliable, use Least Squares fit to a polynomial, and find its derivative at x_i

<u>Cannot</u> expect great accuracy, interpolant may have quite different slope than f!

Roundoff error in approximating derivatives

The previous approximations assume values are known at infinite precision!

In reality, values are contaminated by errors (measurements, roundoff, ...)

so what we actually know are approximate values of $f(x)$:

$$\tilde{f}(x) = f(x) + \varepsilon(x)$$

What we can actually compute is, e.g.

$$D_h\tilde{f}(x) = \frac{\tilde{f}(x+h)-\tilde{f}(x-h)}{2h} = \frac{f(x+h)-f(x-h)}{2h} + \frac{\varepsilon(x+h)-\varepsilon(x-h)}{2h}$$

$$= D_h f(x) + \underline{\qquad\qquad}$$

so error due to "roundoff" is

$$|D_h\tilde{f}(x) - D_h f(x)| \le \frac{1}{h}\cdot\varepsilon \quad , \quad \varepsilon = \max_x\{\varepsilon(x+h), \varepsilon(x-h)\}$$

$\therefore$ the roundoff magnification factor is $\frac{1}{h}$, which $\nearrow \infty$ as $h \searrow 0$!

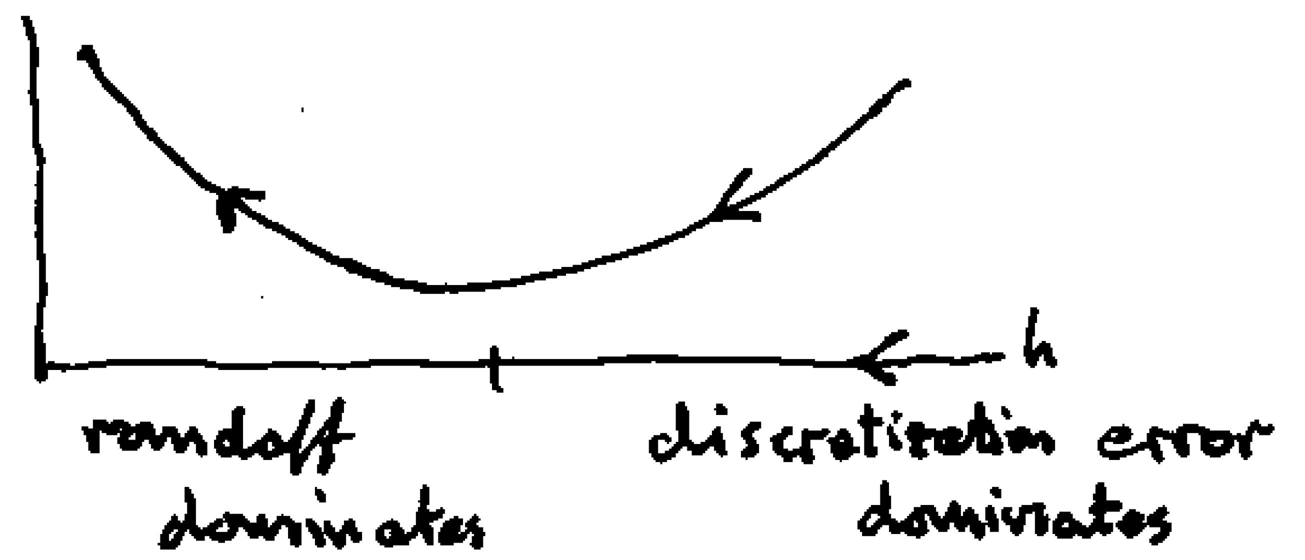
So the process is <u>ill-conditioned</u>: small error ε in values of $f(x)$ may result in big error for $D_h f$ (for small h)

<u>Total error</u> $= f'(x) - D_h\tilde{f}(x)$ = exact − computed has 2 components

$$= \underbrace{f'(x) - D_h f(x)}_{\substack{\text{discretization error}\\ \text{of method at } \infty \text{ precision}}} + \underbrace{D_h f(x) - D_h\tilde{f}(x)}_{\substack{\text{roundoff error}\\ \text{of method due to data errors (roundoff)}}}$$

for centered $\quad = \underbrace{-\frac{f'''(\xi)}{6}h^2}_{\searrow 0} + \underbrace{\frac{\varepsilon}{h}}_{\nearrow\infty} \quad$ as $h \searrow 0$

General fact of computation: As we reduce h in order to reduce discretization error:



1. the approximation improves for a while, then gets worse! There is an optimal h but cannot be found a priori
2. There is a min error that we cannot reduce further! It is the best the method can do. If not good enough, use another method!