Numerical derivatives

Given a univariate function

$$f : \mathbb{R} \to \mathbb{R} \,,$$

compute

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

Only consider *finite differences*.

Numerical derivatives

Given a univariate function

$$f : \mathbb{R} \to \mathbb{R} \ ,$$

compute

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

Only consider *finite differences*.

- Solving differential equations
- Numerical optimization

Numerical derivatives

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

$$\mathbf{g}_k = \frac{\partial f}{\partial x_k}$$

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

$$\mathbf{H}_{jk} = \frac{\partial^2 f}{\partial x_j \partial x_k}$$

Jacobian of $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$

$$\mathbf{J}_{jk} = \frac{\partial f_j}{\partial x_k}$$

Numerical derivatives

- Can compute $f(x)$, a black box.
- Need to construct an approximation $\mathcal{D}f(x)$ for $f'(x)$, balancing
  - Accuracy of the estimate (here: $|\mathcal{D}f$ - $f'| \to \min$)
  - Computational complexity (here: the number of evaluations of f)

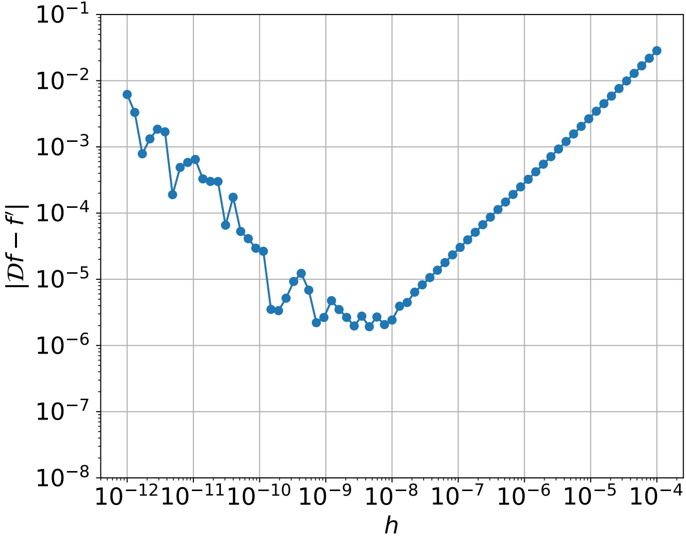
Numerical derivatives: the two-point forward difference

The two-point forward difference scheme

$$\mathcal{D}f \equiv \frac{f(x+h) - f(x)}{h} \approx f'(x)$$

And the only thing left is to choose h?

Numerical derivatives: the two-point forward difference



$f(x) = x^{11}$ $\qquad$ $x = 1.2$

Numerical derivatives: truncation error

Assuming that $f(x)$ is sufficiently smooth in the neighborhood of x, expand it into the Taylor series around x:

$$f(x+h) = f(x) + f'(x)\,h + f''(x)\,\frac{h^2}{2} + f'''(x)\frac{h^3}{6} + O(h^4)$$

$$\mathcal{D}f \equiv \frac{f(x+h) - f(x)}{h} = f'(x) + \frac{f''(x)}{2}h + \cdots$$

Numerical derivatives: two-point central scheme

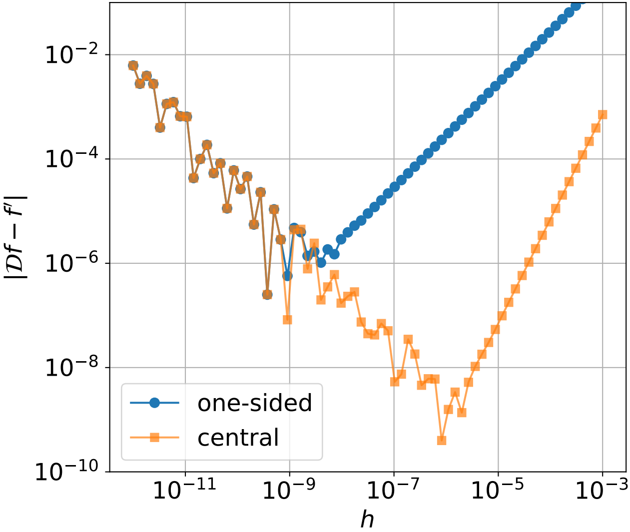
Consider a *central* finite difference scheme

$$\mathcal{D}_c f \equiv \frac{f(x+h) - f(x-h)}{2h}$$

Expand $f(x \pm h)$ into the Taylor series around x,

$$\mathcal{D}_c f = f'(x) + \frac{f'''(x)}{6} h^2 + \cdots$$

Numerical derivatives: the two-point forward difference



$f(x) = x^{11}$ $\qquad$ $x = 1.2$

What is the step of the finite difference scheme?

With exact arithmetics,

$$\frac{(x+h)-x}{h} = 1$$

With floating-point arithmetics, $(x+h)$ is truncated to the nearest exactly representable number

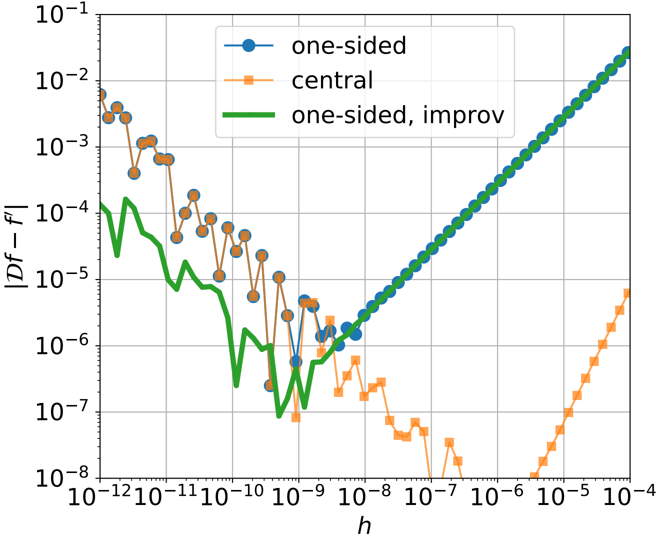
```
>>> x = 1.0
>>> h = np.array([1e-15, 1e-14, 1e-10])
>>> ((x + h) - x)/h - 1.0
array([  1.1e-01,  -8.0e-04,   8.3e-08])
```

What is the step of the finite difference scheme?

To keep the numerator and denominator of $\mathcal{D}f$ consistent, make sure that $x + h$ and x differ by an exactly representable number:

```
>>> def deriv(f, x0, h):
...     x1 = x0 + h
...     dx = x1 - x0
...     df = f(x1) - f(x0)
...     return df / dx
```

Numerical derivatives: the two-point forward difference



$f(x) = x^{11}$ $\qquad\qquad$ $x = 1.2$

Numerical derivatives: roundoff errors

Consider a one-sided two-point scheme

$$\mathcal{D}f = \frac{f(x+h) - f(x)}{h} \approx f'(x)$$

and assume that the values of $f(x)$ are computed to the relative accuracy ϵ_f.

Then the roundoff error of $\mathcal{D}f$ is

$$\epsilon_r \sim \epsilon_f \left| \frac{f(x)}{h} \right|$$

Numerical derivatives: roundoff errors

The total error is the combination of roundoff and linearization errors:

$$\epsilon_r + \epsilon_t \sim \epsilon_f \frac{f}{h} + f'' h \ .$$

This way, the total error has a minimum at

$$h_{\mathrm{opt}} \sim \epsilon_f^{1/2} \sqrt{f/f''} \ .$$

and the minimum value of the total error is

$$\epsilon_{\mathrm{min}} \propto \epsilon_f^{1/2}$$

Numerical derivatives: roundoff errors

For a central finite difference scheme,

$$\mathcal{D}_c f \equiv \frac{f(x+h) - f(x-h)}{2h} \,,$$

truncation error scales $\propto h^2$, thus

$$h_{\text{opt}} \propto \epsilon_f^{1/3}$$
$$\epsilon_{\text{min}} \propto \epsilon_f^{2/3}$$

Numerical derivatives

- The step and the denominator must be consistent
- For a scheme with the truncation error of $O(h^d)$, the optimal step of the finite difference

$$h_{\text{opt}} = \epsilon_f^{1/(d+1)} x_c$$

  The simplest heuristics for x_c is $x_c = \max(1, x)$.
- The best total error scales as

$$\epsilon_{\min} \propto \epsilon_f^{d/(d+1)}$$

Finite differences: higher order schemes

Higher order finite differences

The main idea: form linear combinations of $f(x \pm h)$, $f(x \pm 2h)$ etc.

Example: construct a one-sided scheme of the order $O(h^2)$ for the first derivative.

$$f(x+h) = f + f'\,h + \frac{1}{2}f''\,h^2 + \frac{1}{6}f'''\,h^3 + O(h^4)$$

Need three terms to cancel $f(x)$ and $f''(x)$.

Higher order finite differences

Use the method of indeterminate coefficients: write

$$f'(x) = \frac{A\,f(x) + B\,f(x+h) + C\,f(x+2h)}{h} + O(h^2)$$

and adjust A, B and C.

$$A\,f(x) + B\,f(x+h) + C\,f(x+2h) =$$

$$\begin{aligned} &A\,f \\ &+B\left(f + f'\,h + \frac{1}{2}f''\,h^2 + \frac{1}{6}f'''\,h^3 + O(h^4)\right) \\ &+C\left(f + f'\,2h + \frac{1}{2}f''\,4h^2 + \frac{1}{6}f'''\,8h^3 + O(h^4)\right) \end{aligned}$$

Higher order finite differences

Solving the linear system for the coefficients in front of $f(x)$, $f'(x)$ and $f''(x)$, we find

$$f'(x) = \frac{-\frac{3}{2}f(x) + 2f(x+h) - \frac{1}{2}f(x+2h)}{h} + O(h^2) \ .$$

Likewise,

$$f''(x) = \frac{f(x) - 2f(x+h) + f(x+2h)}{h^2} + O(h) \ .$$

For a better algorithm of generating such schemes see:
B. Fornberg, *Generation of finite difference formulas on arbitrary spaced grids*, Mathematics of Computation **51**, 669 (1988).