Numerical Optimization

Chapter 7: Numerical Differentiation

Oscar Dalmau

Centro de Investigación en Matemáticas
CIMAT

April 27, 2021

Outline

Some reasons for finite difference approximations

There are several reasons to approximate derivatives:

- In some cases, **there exists an underlying function** that we need to differentiate, but we might know its values only at a sampled data set **without knowing the function itself**.
- There are some cases where **it is not obvious that an underlying function exists** and we only have a discrete data set. In this case, we may still be interested in studying changes in the data (which are related to derivatives).

Some reasons for finite difference approximations

- In some cases **we have the exact formula**, but it is very complicated to compute the derivative.
- For example:

$$
\begin{aligned}
f(x;\boldsymbol{a},\boldsymbol{b}) &= \sigma(a_1\sigma(a_2\sigma(a_3x+b_3))+b_2)+b_1) \\
\sigma(x) &= \frac{1}{1+e^{-x}}
\end{aligned}
$$

  and we interested in computing $\frac{\partial F_x}{\partial a_i}$ where

$$
F_x(\boldsymbol{a},\boldsymbol{b}) := f(x;\boldsymbol{a},\boldsymbol{b})
$$

- It might be simpler to **approximate the derivative** instead of computing the exact value or use an **automatic technique**.

Some reasons for finite difference approximations

- When we find approximate solutions to **variational problem and ordinary (or partial) differential equations**, we typically represent the solution as a discrete approximation that is defined on a grid.
- Since we have to evaluate derivatives at the grid points, we need methods for approximating the derivatives at these points
- The underlying function (in this case the solution of the equation) is unknown.

Some reasons for finite difference approximations

- **Finite Differencing**. This technique has is based on the Taylor's theorem
- **Automatic Differentiation.** This technique takes the view that the computer code for evaluating the function can be broken down into a composition of elementary arithmetic operations (build a Computational graph), to which the chain rule can be applied, for example backpropagation (reverse mode) in machine learning, or forward mode.
- **Symbolic Differentiation.** In this technique, the algebraic specification for the function f is manipulated by symbolic manipulation tools to produce new algebraic expressions for each component of the gradient, for example Mathematica or Maple

Forward difference

(See: http://web.media.mit.edu/ crtaylor/calculator.html): Let $f : \mathbb{R} \to \mathbb{R}$. Using Taylor

$$f(x+h) \quad = \quad f(x) + f'(x)h + O(h^2)$$

then

$$f'(x) \quad = \quad \frac{f(x+h) - f(x)}{h} + O(h)$$

Forward difference with order of approximation error 1 or is $O(h)$ (or first order method)

$$\delta_h^+ f(x) \quad = \quad \frac{f(x+h) - f(x)}{h}$$

Backward difference

Let $f : \mathbb{R} \to \mathbb{R}$. Using Taylor

$$f(x-h) \quad = \quad f(x) - f'(x)h + O(h^2)$$

Then

$$f'(x) \quad = \quad \frac{f(x) - f(x-h)}{h} + O(h)$$

Backward difference of approximation with error of order 1 or is $O(h)$

$$\delta_h^- f(x) \quad = \quad \frac{f(x) - f(x-h)}{h}$$

Centered differences

Using Taylor for $x \pm h$

$$
\begin{aligned}
f(x+h) &= \cancel{f(x)} + f'(x)h + \cancel{\frac{1}{2}f''(x)h^2} + \frac{1}{6}f'''(x)h^3 + \ldots \\
f(x-h) &= \cancel{f(x)} - f'(x)h + \cancel{\frac{1}{2}f''(x)h^2} - \frac{1}{6}f'''(x)h^3 + \ldots
\end{aligned}
$$

then

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

Centered differences of approximation error of order 2 or is $O(h^2)$

$$
\delta_h f(x) = \frac{f(x+h) - f(x-h)}{2h}
$$

Centered differences

We note that **centered differences**

$$\delta_h f(x) \quad = \quad \frac{f(x+h) - f(x-h)}{2h}$$

can be written as follows

$$\delta_h f(x) \quad = \quad \frac{1}{2}\big(\delta_h^+ f(x) + \delta_h^- f(x)\big) = \frac{f(x+h) - f(x-h)}{2h}$$

Centered differences

In general, if $a \neq 0$

$$
\begin{aligned}
\delta_{ah}^{+} f(x) &= \frac{f(x+ah) - f(x)}{ah} \\
\delta_{ah}^{-} f(x) &= \frac{f(x) - f(x-ah)}{ah} \\
\delta_{ah} f(x) &= \frac{f(x+ah) - f(x-ah)}{2ah} = \frac{1}{2}\big(\delta_{ah}^{+} f(x) + \delta_{ah}^{-} f(x)\big)
\end{aligned}
$$

For example, if $a = \frac{1}{2}$ then

$$
\delta_{\frac{h}{2}} f(x) = \frac{f(x+\frac{h}{2}) - f(x-\frac{h}{2})}{h} = \frac{1}{2}\big(\delta_{\frac{h}{2}}^{+} f(x) + \delta_{\frac{h}{2}}^{-} f(x)\big)
$$

Example

Consider the function $f(x) = xe^x$ whose first derivative is $f'(x) = (x+1)e^x$ at point $x_0 = 2$ is $f'(x_0) = 22.1672$. Comment: the error is the truncation error

Table: Finite difference approximations at $x_0 = 2$.

h	$\delta_h^+(x_0)$	$\delta_h^-(x_0)$	$\delta_h(x_0)$	$\delta_{h/2}(x_0)$	$\delta_{h/4}(x_0)$
0.1	23.7084	20.7491	22.2288	22.1826	22.1710
|error|	1.5413	1.4180	0.0616	0.0154	0.0038
0.01	22.3156	22.0200	22.1678	22.1673	22.1672
|error|	0.1484	0.1472	0.0006	0.0002	0.0000
0.001	22.1820	22.1524	22.1672	22.1672	22.1672
|error|	0.0148	0.0148	0.0000	0.0000	0.0000

Outline

Second-order approximation of the second derivative

How to approximate $f''(x)$? Using Taylor

$$
\begin{aligned}
f(x+h) &= f(x) + \cancel{f'(x)h} + \frac{1}{2}f''(x)h^2 + \cancel{\frac{1}{3!}f'''(x)h^3} + \ldots \\
f(x-h) &= f(x) - \cancel{f'(x)h} + \frac{1}{2}f''(x)h^2 - \cancel{\frac{1}{3!}f'''(x)h^3} + \ldots
\end{aligned}
$$

then

$$
f(x+h) + f(x-h) = 2f(x) + f''(x)h^2 + O(h^4)
$$

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

Second-order approximation of the second derivative

From

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

we obtain the **second-order central difference** of error approximation of order 2

$$\delta_h^2 f(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$

Note: u_h is said to be n**th-order accurate** if the error, $E(h) := \|u - u_h\|$ is proportional to the step-size h to the nth power

$$E(h) = \|u - u_h\| \leq Ch^n$$

ie, $E(h) = \|u - u_h\| = O(h^n)$

Second-order approximation of the second derivative

Second order forward difference of approximation order 1

$$\begin{aligned}\delta^{+}{}_{h}^{2} f(x) &= \frac{\delta_h^+ f(x+h) - \delta_h^+ f(x)}{h} = \frac{\frac{f(x+2h)-f(x+h)}{h} - \frac{f(x+h)-f(x)}{h}}{h} \\ &= \frac{f(x+2h) - 2f(x+h) + f(x)}{h^2}\end{aligned}$$

Second-order backward difference of approximation order 1

$$\begin{aligned}\delta^{-}{}_{h}^{2} f(x) &= \frac{\delta_h^- f(x) - \delta_h^- f(x-h)}{h} = \frac{\frac{f(x)-f(x-h)}{h} - \frac{f(x-h)-f(x-2h)}{h}}{h} \\ &= \frac{f(x) - 2f(x-h) + f(x-2h)}{h^2}\end{aligned}$$

Homework: Compute the approximation order using the Taylor series approach, compute the *truncation error* $f''(x) - \delta^{+}{}_{h}^{2} f(x)$

Approximation of the third derivative

How to approximate $f'''(x)$? Using Taylor (trial and error!)

$$f(x+h) = \cancel{f(x)} + f'(x)h + \cancel{\frac{1}{2}f''(x)h^2} + \frac{1}{3!}f'''(x)h^3 + \cancel{\frac{1}{4!}f^{(iv)}(x)h^4} + \ldots$$

$$f(x-h) = \cancel{f(x)} - f'(x)h + \cancel{\frac{1}{2}f''(x)h^2} - \frac{1}{3!}f'''(x)h^3 + \cancel{\frac{1}{4!}f^{(iv)}(x)h^4} + \ldots$$

then

$$f(x+h) - f(x-h) = 2f'(x)h + \frac{2}{3!}f'''(x)h^3 + \frac{2}{5!}f^{(v)}(x)h^5 + \ldots$$

evaluating the previous relation in $2h$ (similar to Richardson extrapolation, see next Section)

$$f(x+2h) - f(x-2h) = 4f'(x)h + \frac{16}{3!}f'''(x)h^3 + \frac{64}{5!}f^{(v)}(x)h^5 + \ldots$$

Approximation of the third derivative

$$f(x+h) - f(x-h) = 2f'(x)h + \frac{2}{3!}f'''(x)h^3 + \frac{2}{5!}f^{(v)}(x)h^5 + ...$$

$$f(x+2h) - f(x-2h) = 4f'(x)h + \frac{16}{3!}f'''(x)h^3 + \frac{64}{5!}f^{(v)}(x)h^5 + ...$$

multiplying the first relation by 2 and substracting side by side

$$2f(x+h) - 2f(x-h) = \cancel{4f'(x)h} + \frac{4}{3!}f'''(x)h^3 + \frac{4}{5!}f^{(v)}(x)h^5 + ...$$

$$f(x+2h) - f(x-2h) = \cancel{4f'(x)h} + \frac{16}{3!}f'''(x)h^3 + \frac{64}{5!}f^{(v)}(x)h^5 + ...$$

we get

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

Undetermined Coefficients Method

- Then **Undetermined Coefficients Method** is a practical way for generating approximations of derivatives (or integration), ie, better than the previous **trial and error method**
- For example, suppose that we are interested in finding an approximation of the second derivative $f''(x)$ based on the values of the function at three equally spaced points, $f(x-h), f(x), f(x+h)$, i.e.,

$$f''(x) \quad \approx \quad af(x+h) + bf(x) + cf(x-h)$$

the problem is to find the coefficients a, b, c

Undetermined Coefficients Method

In order to find a, b, c we can use Taylor Series for $f(x-h), f(x+h)$, substitute both expansions in

$$f''(x) \quad \approx \quad af(x+h) + bf(x) + cf(x-h)$$

and solve a 3×3 linear system for a, b, c

Undetermined Coefficients Method

$$\begin{aligned} f(x+h) &= f(x) + f'(x)h + \frac{1}{2}f''(x)h^2 + \frac{1}{3!}f'''(x)h^3 + \ldots \\ f(x-h) &= f(x) - f'(x)h + \frac{1}{2}f''(x)h^2 - \frac{1}{3!}f'''(x)h^3 + \ldots \end{aligned}$$

$$\begin{aligned} f''(x) &\approx af(x) + af'(x)h + \frac{a}{2}f''(x)h^2 + \frac{a}{3!}f'''(x)h^3 + \ldots \\ &\quad +bf(x) \\ &\quad +cf(x) - cf'(x)h + \frac{c}{2}f''(x)h^2 - \frac{c}{3!}f'''(x)h^3 + \ldots \\ &= (a+b+c)f(x) + (a-c)f'(x)h + \frac{a+c}{2}f''(x)h^2 + \ldots \end{aligned}$$

Undetermined Coefficients Method

Then

$$
\begin{aligned}
a + b + c &= 0 \\
a - c &= 0 \\
\frac{a+c}{2} h^2 &= 1
\end{aligned}
$$

Solving for a, b, c we obtain $a = c = \frac{1}{h^2}$ and $b = -\frac{2}{h^2}$ then

$$
f''(x) \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}
$$

as $\frac{a-c}{3!} f'''(x) h^3 = 0$ and $\frac{a+c}{4!} f''''(x) h^4 \neq 0$ the approximation order is 2, ie, $\frac{O(h^4)}{h^2} = O(h^2)$, or we can compute the error $f''(x) - \frac{f(x+h)-2f(x)+f(x-h)}{h^2}$ using Taylor

Undetermined Coefficients Method

- The previous strategy can be generalized for any number of points $x + ih$, $i \in I = \{a_1, a_2, \cdots, a_{|I|}\}$ and any derivative order N such that $N < |I|$.
- The problem becomes a $|I| \times |I|$ linear system which should be solved for the unknown coefficients $a_1, a_2, \cdots, a_{|I|}$.
- We can use the Taylor approach to compute the approximation order, ie, by computing the error
- $s = \{a_1, a_2, \cdots, a_{|I|}\}$ is called $|I|$-point stencil. For example, $s = \{-2, -1, 0, 1, 2\}$ is a five-point stencil.
- Examples: In the forward difference derivative $s = \{0, 1\}$, in the backward difference derivative $s = \{-1, 0\}$ and in the second-order central difference $s = \{-1, 0, 1\}$

Richardson's Extrapolation

- Richardson's extrapolation is a general procedure for improving the accuracy of approximations when the structure of the error is known.
- Here we use it in the context of numerical differentiation, however, it can also be used for numerical integration.
- The general idea is to **use two different approximations** to some quantity (derivative/integral) to **form a third more accurate approximation**.

Richardson's Extrapolation

Example 1: Suppose $a_1 a_2 \neq 0$ and the quantity A satisfies

$$A = A(h) + a_1 h + a_2 h^2$$

where

- A is the exact value,
- $A(h)$ is an approximation of A using the step size h and
- 1 is the known order of accuracy (error approximation order $O(h)$) of the method/approximation in this example.

Richardson's Extrapolation

From

$$A = A(h) + a_1 h + a_2 h^2$$

we can use the half of the step size and then we multiply by 2

$$\begin{aligned} A &= A\left(\frac{h}{2}\right) + a_1 \frac{h}{2} + a_2 \frac{h^2}{4} \\ 2A &= 2A\left(\frac{h}{2}\right) + a_1 h + a_2 \frac{h^2}{2} \end{aligned}$$

and substracting the first relation

$$A = 2A\left(\frac{h}{2}\right) - A(h) - a_2 \frac{h^2}{2} = 2A\left(\frac{h}{2}\right) - A(h) + O(h^2)$$

finally, $2A\left(\frac{h}{2}\right) - A(h)$ has an approximation order 2 while $A(h)$ is 1st order accurate.

Richardson's Extrapolation

In general, if $a_1 \neq 0$, $k_1 < k_2$ and

$$A = A(h) + a_1 h^{k_1} + O(h^{k_2})$$

$A(h)$ has accuracy order k_1. Using half of the step size and multiplying by 2^{k_1}

$$2^{k_1} A = 2^{k_1} A\left(\frac{h}{2}\right) + a_1 h^{k_1} + O(h^{k_2})$$

and substracting the first relation

$$A = \frac{2^{k_1} A\left(\frac{h}{2}\right) - A(h)}{2^{k_1} - 1} + O(h^{k_2})$$

and the new approximation has accuracy order k_2

Richardson's Extrapolation

Example 2: For derivative approximation

$$f'(x) \quad = \quad \frac{f(x+h) - f(x-h)}{2h} + a_1 h^2 + a_2 h^4 + \cdots$$

in this case

- $D = f'(x)$ is the exact value,
- $D(h) = \frac{f(x+h) - f(x-h)}{2h}$ is an approximation of D using the step size h and
- 2 is the known order of accuracy (approximation order 2) of the method/approximation in this example.

Richardson's Extrapolation

Following the Example 1,

$$D = \frac{4D(h) - D(2h)}{3} + O(h^4)$$

ie,

$$\begin{aligned} f'(x) &\approx \frac{4D(h) - D(2h)}{3} \\ &= \frac{-f(x+2h) + 8f(x+h) - 8f(x-h) + f(x-2h)}{12h} \end{aligned}$$

has an approximation order 4 while $D(h)$ has 2

Boundary value problem

Second order linear two-point boundary value problem: Find the function $u(x)$ such that

$$\begin{aligned} -u''(x) + \kappa u(x) &= f(x);\ x \in I = [0,1] \\ u(0) = a; &\qquad u(1) = b \end{aligned}$$

Question: Can we use Euler-Lagrange equation to obtain the previous problem, ie, which is the corresponding variational optimization problem?

Boundary value problem

Second order linear two-point boundary value problem: Find the function $u(x)$ such that

$$\begin{aligned} -u''(x) + \kappa u(x) &= f(x);\ x \in I = [0,1] \\ u(0) = a; &\qquad u(1) = b \end{aligned}$$

First, we define a discretization $x_i = x_0 + ih$; $i = 0, 1, \cdots, n+1$, $x_0 = 0$ and $x_{n+1} = 1$. Due to $x_i = ih$ and $x_{n+1} = 1$ then

$$\begin{aligned} x_{n+1} &= (n+1)h \\ h &= \frac{1}{n+1} \\ x_i &= \frac{i}{n+1} \end{aligned}$$

Boundary value problem

Using an approximation of $u''(x)$, for example, central difference

$$
\begin{aligned}
-\frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + \kappa u_i &= f_i;\; i = 1, 2, \cdots, n \\
u_0 = a; &\qquad u_{n+1} = b
\end{aligned}
$$

with

$$
\begin{aligned}
u_i &\stackrel{def}{=} u(x_i) = u(\frac{i}{n+1}) \\
f_i &\stackrel{def}{=} f(x_i) = f(\frac{i}{n+1})
\end{aligned}
$$

for $i = 0, 1, 2, \cdots, n+1$.

Boundary value problem

Then we need to solve (use for example, Conjugate Gradient)

$$Au = b$$

(it corresponds to a 'discrete' quadratic optimization problem) where $u = [u_1, u_2, \cdots, u_n]^T$, $u_0 = a$, $u_{n+1} = b$ and

$$A = \begin{bmatrix} 2+h^2\kappa & -1 & 0 & \cdots & 0 & 0 \\ -1 & 2+h^2\kappa & -1 & \cdots & 0 & 0 \\ \dots & \dots & \dots & \ddots & \dots & \dots \\ 0 & 0 & 0 & \cdots & 2+h^2\kappa & -1 \\ 0 & 0 & 0 & \cdots & -1 & 2+h^2\kappa \end{bmatrix}$$

$$\begin{aligned} b_i &= h^2 f_i;\ i = 2, 3, \cdots, n-1 \\ b_1 &= h^2 f_1 + u_0 = h^2 f_1 + a \\ b_n &= h^2 f_n + u_{n+1} = h^2 f_n + b \end{aligned}$$

Approximating the gradient

- An approximation to the gradient vector $\nabla f(\boldsymbol{x})$ can be obtained by evaluating the function f at different points, where $f : \mathbb{R}^n \to \mathbb{R}$.
- We can approximate the partial derivative $\frac{\partial f}{\partial x_i}$ using the forward difference, backward difference or central difference, as follows

$$\begin{aligned}
\frac{\partial f(\boldsymbol{x})}{\partial x_i} &\approx \frac{f(\boldsymbol{x} + h\boldsymbol{e}_i) - f(\boldsymbol{x})}{h} \\
\frac{\partial f(\boldsymbol{x})}{\partial x_i} &\approx \frac{f(\boldsymbol{x}) - f(\boldsymbol{x} - h\boldsymbol{e}_i)}{h} \\
\frac{\partial f(\boldsymbol{x})}{\partial x_i} &\approx \frac{f(\boldsymbol{x} + h\boldsymbol{e}_i) - f(\boldsymbol{x} - h\boldsymbol{e}_i)}{2h}
\end{aligned}$$

$i = 1, 2, \cdots, n$

Approximating the gradient

The difference approximation, of the partial derivatives $\frac{\partial f(\boldsymbol{x})}{\partial x_i}$, can also be obtained from Taylor's theorem where $\xi \in (0, 1)$

$$
\begin{aligned}
f(\boldsymbol{x} \pm \boldsymbol{d}) &= f(\boldsymbol{x}) \pm \nabla f(\boldsymbol{x})^T \boldsymbol{d} + \frac{1}{2} \boldsymbol{d}^T \nabla^2 f(\boldsymbol{x} + \xi \boldsymbol{d}) \boldsymbol{d} \\
f(\boldsymbol{x} \pm \boldsymbol{d}) &= f(\boldsymbol{x}) \pm \nabla f(\boldsymbol{x})^T \boldsymbol{d} + O(\|\boldsymbol{d}\|^2)
\end{aligned}
$$

by taking $\boldsymbol{d} = h \boldsymbol{e}_i$ with $\boldsymbol{e}_i$ a vector of the canonical basis.

Approximating the gradient

If $\nabla^2 f(\boldsymbol{x})$ is bounded, ie $\|\nabla^2 f(\boldsymbol{x})\| \leq M$, then

$$|f(\boldsymbol{x}+\boldsymbol{d}) - f(\boldsymbol{x}) - \nabla f(\boldsymbol{x})^T \boldsymbol{d}| \quad \leq \quad \frac{M}{2}\|\boldsymbol{d}\|^2$$

for example, if $\boldsymbol{d} = h\boldsymbol{e}_i$ then

$$|f(\boldsymbol{x}+h\boldsymbol{e}_i) - f(\boldsymbol{x}) - h\frac{\partial f(\boldsymbol{x})}{\partial x_i}| \quad \leq \quad \frac{M}{2}h^2$$

$$\begin{aligned} h\frac{\partial f(\boldsymbol{x})}{\partial x_i} &= f(\boldsymbol{x}+h\boldsymbol{e}_i) - f(\boldsymbol{x}) + O(h^2) \\ \frac{\partial f(\boldsymbol{x})}{\partial x_i} &= \frac{f(\boldsymbol{x}+h\boldsymbol{e}_i) - f(\boldsymbol{x})}{h} + O(h) \end{aligned}$$

Approximating the Jacobian

If $\boldsymbol{r} : \mathbb{R}^n \to \mathbb{R}^m$ then

$$\boldsymbol{r}(\boldsymbol{x}+\boldsymbol{d}) \quad = \quad \boldsymbol{r}(\boldsymbol{x}) + J(\boldsymbol{x})\boldsymbol{d} + O(\|\boldsymbol{d}\|^2)$$

where the Jacobian $J(\boldsymbol{x})$ is computed as follows

$$J(\boldsymbol{x}) \quad = \quad \left[\frac{\partial r_j}{\partial x_i}\right]_{\substack{j=1,\ldots,m \\ i=1,\ldots,n}} = \begin{bmatrix} \nabla r_1(\boldsymbol{x})^T \\ \nabla r_2(\boldsymbol{x})^T \\ \cdots \\ \nabla r_m(\boldsymbol{x})^T \end{bmatrix}$$

Approximating the Jacobian

We could be interested in approximating

- One entry of the Jacobian, ie, $\frac{\partial r_j}{\partial x_i}$
- One column of the Jacobian, ie, $\frac{\partial \boldsymbol{r}}{\partial x_i}$
- One row of the Jacobian, ie, $\nabla_{\boldsymbol{x}} r_j$
- Or directly the product $J(\boldsymbol{x})\boldsymbol{d}$

Approximating the Jacobian

When $\boldsymbol{r}$ is twice continuously differentiable with bounded second order derivative, then

$$|\boldsymbol{r}(\boldsymbol{x}+\boldsymbol{d}) - \boldsymbol{r}(\boldsymbol{x}) - J(\boldsymbol{x})\boldsymbol{d}| \leq \frac{M}{2}\|\boldsymbol{d}\|^2$$

We can directly approximate $J(\boldsymbol{x})\boldsymbol{d}$ as

$$\begin{aligned} J(\boldsymbol{x})\boldsymbol{d} &= \frac{\boldsymbol{r}(\boldsymbol{x}+h\boldsymbol{d}) - \boldsymbol{r}(\boldsymbol{x})}{h} + O(h) \\ J(\boldsymbol{x})\boldsymbol{d} &\approx \frac{\boldsymbol{r}(\boldsymbol{x}+h\boldsymbol{d}) - \boldsymbol{r}(\boldsymbol{x})}{h} \end{aligned}$$

with accuracy order 1

Approximating the Jacobian

In one requires to **approximate the Jacobian**, we can do this **by column**

$$\frac{\partial \boldsymbol{r}}{\partial x_i} \approx \frac{\boldsymbol{r}(\boldsymbol{x} + h\boldsymbol{e}_i) - \boldsymbol{r}(\boldsymbol{x})}{h}$$

$i = 1, 2, \cdots, n$

Approximating the Jacobian

entry by entry

$$\frac{\partial r_j}{\partial x_i} \approx \frac{r_j(\boldsymbol{x} + h\boldsymbol{e}_i) - r_j(\boldsymbol{x})}{h}$$

$i = 1, 2, \cdots, n$ (column) and $j = 1, 2, \cdots, m$ (row).

$$\nabla_{\boldsymbol{x}} r_j = \left[\frac{\partial r_j}{\partial x_i}\right]_i$$

Approximating the Jacobian: Example

Just to illustrate the idea, consider the following function (you can imagine more complicated cases)

$$\boldsymbol{r}(\boldsymbol{x}) = \begin{bmatrix} 2(x_2^3 - x_1^2) \\ 3(x_2^3 - x_1^2) + 2(x_3^3 - x_2^2) \\ 3(x_3^3 - x_2^2) + 2(x_4^3 - x_3^2) \\ 3(x_4^3 - x_3^2) + 2(x_5^3 - x_4^2) \\ 3(x_5^3 - x_4^2) \end{bmatrix}$$

Approximating the Jacobian: Example

Structure of the Jacobian

$$J(\boldsymbol{x}) = \begin{bmatrix} \times & \times & & & \\ \times & \times & \times & & \\ & \times & \times & \times & \\ & & \times & \times & \times \\ & & & \times & \times \end{bmatrix}$$

Then

$$J(\boldsymbol{x})\boldsymbol{d} = \begin{bmatrix} \times & \times & & & \\ \times & \times & \times & & \\ & \times & \times & \times & \\ & & \times & \times & \times \\ & & & \times & \times \end{bmatrix} \boldsymbol{d}$$

Approximating the Jacobian: Example

$$J(\boldsymbol{x})\boldsymbol{d} = \begin{bmatrix} \frac{\partial r_1(\boldsymbol{x})}{\partial x_1} d_1 + \frac{\partial r_1(\boldsymbol{x})}{\partial x_2} d_2 \\ \frac{\partial r_2(\boldsymbol{x})}{\partial x_1} d_1 + \frac{\partial r_2(\boldsymbol{x})}{\partial x_2} d_2 + \frac{\partial r_2(\boldsymbol{x})}{\partial x_3} d_3 \\ \frac{\partial r_3(\boldsymbol{x})}{\partial x_2} d_2 + \frac{\partial r_3(\boldsymbol{x})}{\partial x_3} d_3 + \frac{\partial r_3(\boldsymbol{x})}{\partial x_4} d_4 \\ \frac{\partial r_4(\boldsymbol{x})}{\partial x_3} d_3 + \frac{\partial r_4(\boldsymbol{x})}{\partial x_4} d_4 + \frac{\partial r_4(\boldsymbol{x})}{\partial x_5} d_5 \\ \frac{\partial r_5(\boldsymbol{x})}{\partial x_4} d_4 + \frac{\partial r_5(\boldsymbol{x})}{\partial x_5} d_5 \end{bmatrix}$$

In some algorithms, we do not require to compute the full Hessian but the product $\nabla^2 f_k \boldsymbol{d}_k$, like Newton-CG algorithms, then,

$$\nabla f(\boldsymbol{x} + h\boldsymbol{d}) = \nabla f(\boldsymbol{x}) + h\nabla^2 f(\boldsymbol{x})\boldsymbol{d} + O(h^2)$$

and $\nabla^2 f(\boldsymbol{x})\boldsymbol{d}$ can be approximated as

$$\begin{aligned} \nabla^2 f(\boldsymbol{x})\boldsymbol{d} &= \frac{\nabla f(\boldsymbol{x} + h\boldsymbol{d}) - \nabla f(\boldsymbol{x})}{h} + O(h) \\ \nabla^2 f(\boldsymbol{x})\boldsymbol{d} &\approx \frac{\nabla f(\boldsymbol{x} + h\boldsymbol{d}) - \nabla f(\boldsymbol{x})}{h} \end{aligned}$$

with approximation error h.

The **entries of the Hessian** can be approximated using Taylor, ie,

$$f(\boldsymbol{x}+\boldsymbol{d}) \quad = \quad f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^T \nabla^2 f(\boldsymbol{x}) \boldsymbol{d} + O(\|\boldsymbol{d}\|^3)$$

with $\boldsymbol{d} = h\boldsymbol{e}_i$, $\boldsymbol{d} = h\boldsymbol{e}_j$ and $\boldsymbol{d} = h(\boldsymbol{e}_i + \boldsymbol{e}_j)$

The entries of the Hessian can be approximated using Taylor, ie,

$$
\begin{aligned}
f(\boldsymbol{x}+h\boldsymbol{e}_i) &= f(\boldsymbol{x}) + h\frac{\partial f(\boldsymbol{x})}{\partial x_i} + \frac{1}{2}h^2\frac{\partial^2 f(\boldsymbol{x})}{\partial x_i^2} + O(h^3) \\
f(\boldsymbol{x}+h\boldsymbol{e}_j) &= f(\boldsymbol{x}) + h\frac{\partial f(\boldsymbol{x})}{\partial x_j} + \frac{1}{2}h^2\frac{\partial^2 f(\boldsymbol{x})}{\partial x_j^2} + O(h^3) \\
f(\boldsymbol{x}+h(\boldsymbol{e}_i+\boldsymbol{e}_j)) &= f(\boldsymbol{x}) + h\left(\frac{\partial f(\boldsymbol{x})}{\partial x_i} + \frac{\partial f(\boldsymbol{x})}{\partial x_j}\right) \\
&\quad + \frac{1}{2}h^2\left(\frac{\partial^2 f(\boldsymbol{x})}{\partial x_i^2} + 2\frac{\partial^2 f(\boldsymbol{x})}{\partial x_i \partial x_j} + \frac{\partial^2 f(\boldsymbol{x})}{\partial x_j^2}\right) + O(h^3)
\end{aligned}
$$

Then

$$f(\boldsymbol{x} + h(\boldsymbol{e}_i + \boldsymbol{e}_j)) - f(\boldsymbol{x} + h\boldsymbol{e}_i) - f(\boldsymbol{x} + h\boldsymbol{e}_j) = -f(\boldsymbol{x}) + h^2 \frac{\partial^2 f(\boldsymbol{x})}{\partial x_i \partial x_j} + O(h^3)$$

$$\frac{f(\boldsymbol{x} + h\boldsymbol{e}_i + h\boldsymbol{e}_j) - f(\boldsymbol{x} + h\boldsymbol{e}_i) - f(\boldsymbol{x} + h\boldsymbol{e}_j) + f(\boldsymbol{x})}{h^2} = \frac{\partial^2 f(\boldsymbol{x})}{\partial x_i \partial x_j} + O(h)$$

The entries of the Hessian can be approximated using for example the forward-differences

$$\begin{aligned}
\frac{\nabla^2 f(\boldsymbol{x})}{\partial x_i \partial x_j} &= \frac{\partial}{\partial x_i}\frac{\partial f(\boldsymbol{x})}{\partial x_j} \\
&\approx \frac{\frac{\partial f(\boldsymbol{x}+h\boldsymbol{e}_i)}{\partial x_j} - \frac{\partial f(\boldsymbol{x})}{\partial x_j}}{h} \\
&\approx \frac{\frac{f(\boldsymbol{x}+h\boldsymbol{e}_i+h\boldsymbol{e}_j)-f(\boldsymbol{x}+h\boldsymbol{e}_i)}{h} - \frac{f(\boldsymbol{x}+h\boldsymbol{e}_j)-f(\boldsymbol{x})}{h}}{h} \\
&= \frac{f(\boldsymbol{x}+h\boldsymbol{e}_i+h\boldsymbol{e}_j)-f(\boldsymbol{x}+h\boldsymbol{e}_i)-f(\boldsymbol{x}+h\boldsymbol{e}_j)+f(\boldsymbol{x})}{h^2}
\end{aligned}$$

with approximation error h.