Cubic Interpolation & Derivatives in d Dimensions

Oliver K. Ernst
Physics Department, UC San Diego

September 20, 2018

Contents

1 In one dimension

1.1 Interpolation

Consider a 1D grid of points of length n, with abscissas $(s_0, \dots, s_{n-1})$.

Let values be defined on each point, given by $(t_0, \dots, t_{n-1})$.

Given a point $\tilde{x}$, define the index $i \in \{0, 1, \dots, n-2\}$ such that $\tilde{x}$ that falls between s_i, s_{i+1}.

Consider two scenarios:

1. **Case #1:** the point is not near the boundary, i.e. $i \neq 0 \cap i \neq i-2$.

   We can then define the local neighborhood of 4 values:

   $$\begin{aligned} p_0 &= t_{i-1} \\ p_1 &= t_i \\ p_2 &= t_{i+1} \\ p_3 &= t_{i+2} \end{aligned}$$

   or more compactly

   $$p_j = t_{i-1+j}$$

   for $j = 0, 1, 2, 3$.

   When does such a scenario occur?

If the index i falls on the lower boundary, it is $\delta_{i,0}$. A further necessary condition is that the point being queried is at that boundary, i.e. if $i = 0$, then only $j = 0$ is approximated; if $i = n-2$, then only $j = 3$ is approximated. The condition then is $\delta_{i,0}\delta_{j,0}$.

At the other boundary, the condition is $\delta_{i,n-2}\delta_{j,3}$.

If the first condition is not met, the original t_{i-1+j} holds and it is $1 - \delta_{i,0}\delta_{j,0}$ and $1 - \delta_{i,n-2}\delta_{j,3}$.

If both are not met, it is $(1 - \delta_{i,0}\delta_{j,0})(1 - \delta_{i,n-2}\delta_{j,3})$.

2. **Case #2:** the point is near the boundary, i.e. $(i = 0 \cap j = 0) \cup (i = n-2 \cap j = 3)$.

   In this case, we must approximate the values at the boundary. The best approximation is a linear one:

$$\begin{aligned} & p_0 \approx 2p_1 - p_2 = 2t_i - t_{i+1} \\ \text{or} \qquad & p_3 \approx 2p_2 - p_1 = 2t_{i+1} - t_i \end{aligned} \tag{1}$$

The general formula for p is then:

$$p_j = (1 - \delta_{i,0}\delta_{j,0})(1 - \delta_{i,n-2}\delta_{j,3})t_{i-1+j} + \delta_{i,0}\delta_{j,0}(2t_i - t_{i+1}) + \delta_{i,n-2}\delta_{j,3}(2t_{i+1} - t_i) \tag{2}$$

for $j = 0, 1, 2, 3$.

Finally, let x be the **fraction** that $\tilde{x}$ is between the two neighboring points, i.e. $x = (\tilde{x} - s_i)/(s_{i+1} - s_i)$.

The cubic interpolation formula is:

$$f(x; p_0, p_1, p_2, p_3) = \left(-\frac{1}{2}p_0 + \frac{3}{2}p_1 - \frac{3}{2}p_2 + \frac{1}{2}p_3\right)x^3 + \left(p_0 - \frac{5}{2}p_1 + 2p_2 - \frac{1}{2}p_3\right)x^2 + \left(-\frac{1}{2}p_0 + \frac{1}{2}p_2\right)x + p_1 \tag{3}$$

1.2 The derivative with respect to a point value

Consider the derivative

$$\frac{df}{dp_k} \tag{4}$$

Assume that p_k corresponds to a real point and is not approximated by other values, i.e.

$$p_k = t_{i-1+k} \tag{5}$$

then:

$$\frac{df}{dp_k} = \sum_{j=0}^{3} \frac{\partial f}{\partial p_j}\frac{\partial p_j}{\partial p_k} \tag{6}$$

To evaluate the second term:

$$\begin{aligned} \frac{\partial p_j}{\partial p_k} =& \frac{\partial p_j}{\partial t_{i-1+k}} \\ =& (1 - \delta_{i,0}\delta_{j,0})(1 - \delta_{i,n-2}\delta_{j,3})\frac{\partial t_{i-1+j}}{\partial t_{i-1+k}} + \delta_{i,0}\delta_{j,0}(2\frac{\partial t_i}{\partial t_{i-1+k}} - \frac{\partial t_{i+1}}{\partial t_{i-1+k}}) + \delta_{i,n-2}\delta_{j,3}(2\frac{\partial t_{i+1}}{\partial t_{i-1+k}} - \frac{\partial t_i}{\partial t_{i-1+k}}) \\ =& (1 - \delta_{i,0}\delta_{j,0})(1 - \delta_{i,n-2}\delta_{j,3})\delta_{k,j} + \delta_{i,0}\delta_{j,0}(2\delta_{k,1} - \delta_{k,0}) + \delta_{i,n-2}\delta_{j,3}(2\delta_{k,2} - \delta_{k,1}) \end{aligned} \tag{7}$$

Then

$$\frac{df}{dp_k} = (1 - \delta_{i,0}\delta_{k,0})(1 - \delta_{i,n-2}\delta_{k,3})\frac{\partial f}{\partial p_k} + \delta_{i,0}(2\delta_{k,1} - \delta_{k,0})\frac{\partial f}{\partial p_0} + \delta_{i,n-2}(2\delta_{k,2} - \delta_{k,1})\frac{\partial f}{\partial p_3} \tag{8}$$

The first term is the standard derivative. The delta terms in front of it ensures: (1) it exists when we are not at the boundary, (2) it vanishes when we are at the $i = 0$ boundary and trying to take a derivative with respect to $k = 0$ (p_0 is not a real point!), and (3) similar for $i = n - 2$ & $k = 3$.

The remaining derivatives that appear are:

$$\begin{aligned}
\frac{\partial f}{\partial p_0} &= -\frac{1}{2}x^3 + x^2 - \frac{1}{2}x \\
\frac{\partial f}{\partial p_1} &= \frac{3}{2}x^3 - \frac{5}{2}x^2 + 1 \\
\frac{\partial f}{\partial p_2} &= -\frac{3}{2}x^3 + 2x^2 + \frac{1}{2}x \\
\frac{\partial f}{\partial p_3} &= \frac{1}{2}x^3 - \frac{1}{2}x^2
\end{aligned} \tag{9}$$

1.3 The derivative with respect to x

The derivative with respect to the point $\tilde{x}$ is:

$$\begin{aligned}
\frac{\partial f}{\partial x} &= 3\left(-\frac{1}{2}p_0 + \frac{3}{2}p_1 - \frac{3}{2}p_2 + \frac{1}{2}p_3\right)x^2 + 2\left(p_0 - \frac{5}{2}p_1 + 2p_2 - \frac{1}{2}p_3\right)x - \frac{1}{2}p_0 + \frac{1}{2}p_2 \\
\frac{\partial f}{\partial \tilde{x}} &= \frac{\partial f}{\partial x}\frac{\partial x}{\partial \tilde{x}} = \frac{\partial f}{\partial x}\left(s_{i+1} - s_i\right)^{-1}
\end{aligned} \tag{10}$$

If the point falls near the boundary, the result is unchanged as if we make the substitutions $p_0 \approx 2p_1 - p_2$ or $p_3 \approx 2p_2 - p_1$ in (10).

2 In d-dimensions

2.1 Interpolation

In d dimensions, let the grid abscissas in each dimension $\delta = 0, \dots, d-1$ be $(a_0^{\langle\delta\rangle}, \dots, a_{n_d-1}^{\langle\delta\rangle})$ of length $n^{\langle\delta\rangle}$.

The vector of length d describing the location of a grid point is then:

$$\boldsymbol{s}^{[d]:\langle j_0,\dots,j_{d-1}\rangle} = \left(a_{j_0}^{\langle 0\rangle}, \dots, a_{j_{d-1}}^{\langle d-1\rangle}\right) \tag{11}$$

Here, we use the notation $\boldsymbol{s}^{[d]:\langle j_0,\dots,j_{d-1}\rangle}$ that there is one of such vector for every set of $[d]$ indexes $\langle j_0, \dots, j_{d-1}\rangle$.

Let the values associated with each grid point be:

$$t^{[d]:\langle j_0,\dots,j_{d-1}\rangle} \tag{12}$$

To further reduce the burden of indexing, define:

$$[d] : \langle 0 : d-1\rangle = [d] : \langle j_0, \dots, j_{d-1}\rangle \tag{13}$$

when specific indexes are not needed, then the abscissas and associated points are:

$$\begin{aligned}
\text{abscissa:} \quad & \boldsymbol{s}^{[d]:\langle 0:d-1\rangle} \\
\text{ordinate:} \quad & t^{[d]:\langle 0:d-1\rangle}
\end{aligned}$$

Given a point $\tilde{\boldsymbol{x}} = (\tilde{x}_0, \dots, \tilde{x}_{d-1})$, define the indexes $i_\delta \in \{0, 1, \dots, n^{\langle\delta\rangle} - 2\}$ for $\delta = 0, \dots, d-1$ such that $\tilde{x}_\delta$ falls between $a_{i_\delta}^{\langle\delta\rangle}, a_{i_{\delta+1}}^{\langle\delta\rangle}$.

We want to define the values p as in the 1D case. We must consider two scenarios as before:

1. **Case #1**: all points are in the interior, i.e. $i_\delta \neq 0 \cap i_\delta \neq n^{\langle\delta\rangle} - 2$ for all δ.

   Define the values $p^{[d]:\langle 0:d-1\rangle}$:

$$
\begin{aligned}
p^{[d]:\langle 0,\ldots,0,0\rangle} &= t^{[d]:\langle i_0-1,\ldots,i_{d-2}-1,i_{d-1}-1\rangle} \\
p^{[d]:\langle 0,\ldots,0,1\rangle} &= t^{[d]:\langle i_0-1,\ldots,i_{d-2}-1,i_{d-1}\rangle} \\
p^{[d]:\langle 0,\ldots,0,2\rangle} &= t^{[d]:\langle i_0-1,\ldots,i_{d-2}-1,i_{d-1}+1\rangle} \\
p^{[d]:\langle 0,\ldots,0,3\rangle} &= t^{[d]:\langle i_0-1,\ldots,i_{d-2}-1,i_{d-1}+2\rangle} \\
p^{[d]:\langle 0,\ldots,1,0\rangle} &= t^{[d]:\langle i_0-1,\ldots,i_{d-2},i_{d-1}-1\rangle} \\
p^{[d]:\langle 0,\ldots,2,0\rangle} &= t^{[d]:\langle i_0-1,\ldots,i_{d-2}+1,i_{d-1}-1\rangle} \\
p^{[d]:\langle 0,\ldots,3,0\rangle} &= t^{[d]:\langle i_0-1,\ldots,i_{d-2}+2,i_{d-1}-1\rangle}
\end{aligned}
$$

   etc., or more generally

$$
p^{[d]:\langle j_0,\ldots,j_{d-1}\rangle} = t^{[d]:\langle i_0-1+j_0,\ldots,i_{d-1}-1+j_{d-1}\rangle}
$$

   where $j = 0,1,2,3$. There are 4^d of such points $p^{[d]:\langle 0:d-1\rangle}$ in total.

   What is the condition for this event to occur?

   If all points fall near the lower boundary, it is $\prod_{\alpha=0}^{d-1} \delta_{i_\delta,0}\delta_{j_\delta,0}$.

   If no points fall near the lower boundary, it is $\prod_{\alpha=0}^{d-1}(1-\delta_{i_\delta,0}\delta_{j_\delta,0})$.

   Combined with the upper boundary, the full condition for no points to fall near the boundary is:

$$
\prod_{\alpha=0}^{d-1}(1-\delta_{i_\delta,0}\delta_{j_\delta,0})(1-\delta_{i_\delta,n^{\langle\delta\rangle}-2}\delta_{j_\delta,3})
$$

2. **Case #2**: at least one point falls near the boundary, i.e. $(i_\delta = 0 \cap j_\delta = 0) \cup (i_\delta = n^{\langle\delta\rangle} - 2 \cap j_\delta = 3)$ for at least one δ.

   When this event occurs, some points p must be approximated by others. Define the mappings acting on indexes i_δ:

$$
\begin{aligned}
\mathcal{M} &: \begin{cases} i_\delta \to i_\delta + 1 & \text{if } i_\delta = -1 \\ i_\delta \to i_\delta - 1 & \text{if } i_\delta = n^{\langle\delta\rangle} \\ i_\delta \to i_\delta & \text{otherwise} \end{cases} \\
\mathcal{P} &: \begin{cases} i_\delta \to i_\delta + 2 & \text{if } i_\delta = -1 \\ i_\delta \to i_\delta - 2 & \text{if } i_\delta = n^{\langle\delta\rangle} \\ i_\delta \to i_\delta & \text{otherwise} \end{cases}
\end{aligned}
\tag{14}
$$

   that is, $\mathcal{M}$ takes one step back into the lattice for any indexes that are outside, and $\mathcal{P}$ takes two steps back in.

   We let the notation $[d] : \mathcal{M}\langle j_0,\ldots,j_{d-1}\rangle$ denote that $\mathcal{M}$ is applied to all indexes, and similarly for $\mathcal{P}$.

   If a point $t^{[d]:\langle i_0-1+j_0,\ldots,i_{d-1}-1+j_{d-1}\rangle}$ is outside the lattice, then generalizing the linear approximation from the 1D case, it can be approximated as:

$$
t^{[d]:\langle i_0-1+j_0,\ldots,i_{d-1}-1+j_{d-1}\rangle} \approx 2 \times t^{[d]:\mathcal{M}\langle i_0-1+j_0,\ldots,i_{d-1}-1+j_{d-1}\rangle} - t^{[d]:\mathcal{P}\langle i_0-1+j_0,\ldots,i_{d-1}-1+j_{d-1}\rangle}
$$

   where the mappings applied to the indexes have now referenced a valid point.

The general formula then for the points p is:

$$p^{[d]:\langle j_0,\ldots,j_{d-1}\rangle} = \left(\prod_{\alpha=0}^{d-1}(1-\delta_{i_\delta,0}\delta_{j_\delta,0})(1-\delta_{i_\delta,n^{\langle\delta\rangle}-2}\delta_{j_\delta,3})\right) t^{[d]:\langle i_0-1+j_0,\ldots,i_{d-1}-1+j_{d-1}\rangle} + \left(1-\prod_{\alpha=0}^{d-1}(1-\delta_{i_\delta,0}\delta_{j_\delta,0})(1-\delta_{i_\delta,n^{\langle\delta\rangle}-2}\delta_{j_\delta,3})\right) \times \left(2\times t^{[d]:\mathcal{M}\langle i_0-1+j_0,\ldots,i_{d-1}-1+j_{d-1}\rangle} - t^{[d]:\mathcal{P}\langle i_0-1+j_0,\ldots,i_{d-1}-1+j_{d-1}\rangle}\right) \tag{15}$$

where $j=0,1,2,3$.

Finally, let $\boldsymbol{x}$ be the fraction with components:

$$x_\delta = (\tilde{x}_\delta - a_{i_\delta}^{\langle\delta\rangle})/(a_{i_\delta+1}^{\langle\delta\rangle} - a_{i_\delta}^{\langle\delta\rangle}) \tag{16}$$

for $\delta = 0,\ldots,d-1$.

The cubic interpolation now proceeds iteratively - define points $p^{[d-1]:\langle 0:d-2\rangle}$:

$$p^{[d-1]:\langle j_0,\ldots,j_{d-2}\rangle} = f\left(x_{d-1}; p^{[d]:\langle j_0,\ldots,j_{d-2},0\rangle}, p^{[d]:\langle j_0,\ldots,j_{d-2},1\rangle}, p^{[d]:\langle j_0,\ldots,j_{d-2},2\rangle}, p^{[d]:\langle j_0,\ldots,j_{d-2},3\rangle}\right) \tag{17}$$

(notice the indexes appearing on the left), for all $j=0,1,2,3$, or equivalently and more compactly:

$$p^{[d-1]:\langle 0:d-2\rangle} = f\left(x_{d-1}; p^{[d]:\langle 0:d-2,0\rangle}, p^{[d]:\langle 0:d-2,1\rangle}, p^{[d]:\langle 0:d-2,2\rangle}, p^{[d]:\langle 0:d-2,3\rangle}\right) \tag{18}$$

There are 4^{d-1} of such points $p^{[d-1]:\langle 0:d-2\rangle}$.

In general, the recursion relation to go from dimension $\delta+1$ to δ is:

$$p^{[\delta]:\langle 0:\delta-1\rangle} = f\left(x_\delta; p^{[\delta+1]:\langle 0:\delta-1,0\rangle}, p^{[\delta+1]:\langle 0:\delta-1,1\rangle}, p^{[\delta+1]:\langle 0:\delta-1,2\rangle}, p^{[\delta+1]:\langle 0:\delta-1,3\rangle}\right) \tag{19}$$

for $\delta=0,\ldots,d-1$. The last iteration with $\delta=0$ gives:

$$p^{[0]} = f\left(x_0; p^{[1]:\langle 0\rangle}, p^{[1]:\langle 1\rangle}, p^{[1]:\langle 2\rangle}, p^{[1]:\langle 3\rangle}\right) \tag{20}$$

is the desired interpolated value we seek.

2.1.1 Pseudocode

1. function **iterate**(δ, d, $\boldsymbol{x}$, $(j_1,\ldots,j_{d-1})$, $p^{[d]:\langle 0:d-1\rangle}$):
   // This calculates $p^{[\delta]:\langle 0:\delta-1\rangle}$ = left side of (19)
   (a) if $\delta == d-1$: // Arrived at (18) which we can do with the points given
      i. return $f\left(x_{d-1}; p^{[d]:\langle 0:d-2,0\rangle}, p^{[d]:\langle 0:d-2,1\rangle}, p^{[d]:\langle 0:d-2,2\rangle}, p^{[d]:\langle 0:d-2,3\rangle}\right)$ // $= p^{[d-1]}$
   (b) else: // Calculate the right side of (19)
      i. $p^{[\delta+1]:\langle 0:\delta-1,0\rangle}$ = **iterate**($\delta+1$, d, $\boldsymbol{x}$, $(j_1,\ldots,j_\delta=0,\ldots,j_{d-1})$, $p^{[d]:\langle 0:d-1\rangle}$)
      ii. $p^{[\delta+1]:\langle 0:\delta-1,1\rangle}$ = **iterate**($\delta+1$, d, $\boldsymbol{x}$, $(j_1,\ldots,j_\delta=1,\ldots,j_{d-1})$, $p^{[d]:\langle 0:d-1\rangle}$)
      iii. $p^{[\delta+1]:\langle 0:\delta-1,2\rangle}$ = **iterate**($\delta+1$, d, $\boldsymbol{x}$, $(j_1,\ldots,j_\delta=2,\ldots,j_{d-1})$, $p^{[d]:\langle 0:d-1\rangle}$)
      iv. $p^{[\delta+1]:\langle 0:\delta-1,3\rangle}$ = **iterate**($\delta+1$, d, $\boldsymbol{x}$, $(j_1,\ldots,j_\delta=3,\ldots,j_{d-1})$, $p^{[d]:\langle 0:d-1\rangle}$)
      v. return $f\left(x_\delta; p^{[\delta+1]:\langle 0:\delta-1,0\rangle}, p^{[\delta+1]:\langle 0:\delta-1,1\rangle}, p^{[\delta+1]:\langle 0:\delta-1,2\rangle}, p^{[\delta+1]:\langle 0:\delta-1,3\rangle}\right)$ // $= p^{[\delta]}$
2. To start: **iterate**(0,d,$\boldsymbol{x}$,$(j_1,\ldots,j_{d-1})$,$p^{[d]:\langle 0:d-1\rangle}$)

2.2 The derivative with respect to a point value

What is the derivative with respect to a point value?

Let the point to differentiate with respect to be:

$$p^{[d]:\langle k_0,\ldots,k_{d-1}\rangle} \tag{21}$$

where $k = 0, 1, 2, 3$. We assume that by definition, we are differentiating with respect to a real lattice point, not one which is outside the lattice. In terms of the indexes i_δ, this can then be written as:

$$p^{[d]:\langle k_0,\ldots,k_{d-1}\rangle} = t^{[d]:\langle i_0-1+k_0,\ldots,i_{d-1}-1+k_{d-1}\rangle} \tag{22}$$

Then we seek:

$$\frac{\partial p^{[0]}}{\partial p^{[d]:\langle k_0,\ldots,k_{d-1}\rangle}} \tag{23}$$

Differentiating the recursion (19) and using the chain rule gives:

$$\frac{\partial p^{[\delta]:\langle 0:\delta-1\rangle}}{\partial p^{[d]:\langle k_0,\ldots,k_{d-1}\rangle}} = \sum_{j_\delta=0}^{3} \frac{\partial f\left(x_\delta; p^{[\delta+1]:\langle 0:\delta-1,0\rangle}, p^{[\delta+1]:\langle 0:\delta-1,1\rangle}, p^{[\delta+1]:\langle 0:\delta-1,2\rangle}, p^{[\delta+1]:\langle 0:\delta-1,3\rangle}\right)}{\partial p^{[\delta+1]:\langle 0:\delta-1,j_\delta\rangle}} \frac{\partial p^{[\delta+1]:\langle 0:\delta-1,j_\delta\rangle}}{\partial p^{[d]:\langle k_0,\ldots,k_{d-1}\rangle}} \tag{24}$$

for $\delta = 0, \ldots, d-2$.

The first term can be evaluated using (9). We immediately notice an important property: the interpolation is linear in the point values, such that the first term does not depend on them. This greatly reduces the complexity - we use the notation from (9):

$$\frac{\partial f(x_\delta)}{\partial p_{j_\delta}} \tag{25}$$

to denote the derivative, giving:

$$\frac{\partial p^{[\delta]:\langle 0:\delta-1\rangle}}{\partial p^{[d]:\langle k_0,\ldots,k_{d-1}\rangle}} = \sum_{j_\delta=0}^{3} \frac{\partial f(x_\delta)}{\partial p_{j_\delta}} \frac{\partial p^{[\delta+1]:\langle 0:\delta-1,j_\delta\rangle}}{\partial p^{[d]:\langle k_0,\ldots,k_{d-1}\rangle}} \tag{26}$$

With $\delta = 0$ and using the recursion $d-1$ times gives:

$$\frac{\partial p^{[0]}}{\partial p^{[d]:\langle k_0,\ldots,k_{d-1}\rangle}} = \sum_{j_0,\ldots,j_{d-1}} \left(\prod_{\alpha=0}^{d-1} \frac{\partial f(x_\alpha)}{\partial p_{j_\alpha}} \right) \left(\frac{\partial p^{[d]:\langle j_0,\ldots,j_{d-1}\rangle}}{\partial p^{[d]:\langle k_0,\ldots,k_{d-1}\rangle}} \right) \tag{27}$$

It is tempting to simply stick in delta functions for the term on the right; however we must be careful and use (15) instead:

$$\begin{aligned}
\frac{\partial p^{[d]:\langle j_0,\ldots,j_{d-1}\rangle}}{\partial p^{[d]:\langle k_0,\ldots,k_{d-1}\rangle}} =& \frac{\partial p^{[d]:\langle j_0,\ldots,j_{d-1}\rangle}}{\partial t^{[d]:\langle i_0-1+k_0,\ldots,i_{d-1}-1+k_{d-1}\rangle}} \\
=& \left(\prod_{\alpha=0}^{d-1}(1-\delta_{i_\delta,0}\delta_{j_\delta,0})(1-\delta_{i_\delta,n^{\langle\delta\rangle}-2}\delta_{j_\delta,3})\right)\frac{\partial t^{[d]:\langle i_0-1+j_0,\ldots,i_{d-1}-1+j_{d-1}\rangle}}{\partial t^{[d]:\langle i_0-1+k_0,\ldots,i_{d-1}-1+k_{d-1}\rangle}} \\
&+\left(1-\prod_{\alpha=0}^{d-1}(1-\delta_{i_\delta,0}\delta_{j_\delta,0})(1-\delta_{i_\delta,n^{\langle\delta\rangle}-2}\delta_{j_\delta,3})\right) \\
&\quad\times\left(2\times\frac{\partial t^{[d]:\mathcal{M}\langle i_0-1+j_0,\ldots,i_{d-1}-1+j_{d-1}\rangle}}{\partial t^{[d]:\langle i_0-1+k_0,\ldots,i_{d-1}-1+k_{d-1}\rangle}}-\frac{\partial t^{[d]:\mathcal{P}\langle i_0-1+j_0,\ldots,i_{d-1}-1+j_{d-1}\rangle}}{\partial t^{[d]:\langle i_0-1+k_0,\ldots,i_{d-1}-1+k_{d-1}\rangle}}\right) \\
=& \left(\prod_{\alpha=0}^{d-1}(1-\delta_{i_\delta,0}\delta_{j_\delta,0})(1-\delta_{i_\delta,n^{\langle\delta\rangle}-2}\delta_{j_\delta,3})\right)\left(\prod_{\alpha=0}^{d-1}\delta_{k_\delta,j_\delta}\right) \\
&+\left(1-\prod_{\alpha=0}^{d-1}(1-\delta_{i_\delta,0}\delta_{j_\delta,0})(1-\delta_{i_\delta,n^{\langle\delta\rangle}-2}\delta_{j_\delta,3})\right) \\
&\quad\times\left(2\prod_{\alpha=0}^{d-1}\delta_{i_\delta-1+k_\delta,\mathcal{M}(i_\delta-1+j_\delta)}-\prod_{\alpha=0}^{d-1}\delta_{i_\delta-1+k_\delta,\mathcal{P}(i_\delta-1+j_\delta)}\right)
\end{aligned} \tag{28}$$

The first term is easy, as it just picks out $j_\delta = k_\delta$ in the sums. To proceed in simplifying the second term, we would need the inverse mappings $\mathcal{M}^{-1}$, $\mathcal{P}^{-1}$ - but these don't exist.

We are left with:

$$\begin{aligned}
\frac{\partial p^{[0]}}{\partial p^{[d]:\langle k_0,\ldots,k_{d-1}\rangle}} =& \left(\prod_{\alpha=0}^{d-1}(1-\delta_{i_\delta,0}\delta_{j_\delta,0})(1-\delta_{i_\delta,n^{\langle\delta\rangle}-2}\delta_{j_\delta,3})\right)\left(\prod_{\alpha=0}^{d-1}\frac{\partial f(x_\alpha)}{\partial p_{k_\alpha}}\right) \\
&+\left(1-\prod_{\alpha=0}^{d-1}(1-\delta_{i_\delta,0}\delta_{j_\delta,0})(1-\delta_{i_\delta,n^{\langle\delta\rangle}-2}\delta_{j_\delta,3})\right) \\
&\quad\times\sum_{j_0,\ldots,j_{d-1}}\left(\prod_{\alpha=0}^{d-1}\frac{\partial f(x_\alpha)}{\partial p_{j_\alpha}}\right)\left(2\prod_{\alpha=0}^{d-1}\delta_{i_\delta-1+k_\delta,\mathcal{M}(i_\delta-1+j_\delta)}-\prod_{\alpha=0}^{d-1}\delta_{i_\delta-1+k_\delta,\mathcal{P}(i_\delta-1+j_\delta)}\right)
\end{aligned} \tag{29}$$

The algorithm here is therefore split:

1. Case #1: the point is interior, i.e. away from the boundary in **all** dimensions. To check this, check that $i_\delta \neq 0 \cap i_\delta \neq n^{\langle\delta\rangle} - 2$ for **all** $\delta = 0, \ldots, d-1$.

   Here the case easy, and does not require recursion! The answer is:

   $$\prod_{\alpha=0}^{d-1}\frac{\partial f(x_\alpha)}{\partial p_{k_\alpha}} \tag{30}$$

   which can be evaluated using (9).

2. Case # 2: here there is no apparent simplification, and we must resort to a recursive approach to evaluate (26).

2.2.1 Pseudocode for case #2

1. function **iterate_deriv_p**(δ,d,$(j_0,\dots,j_{d-1})$,$p^{[d]:\langle 0:d-1\rangle}$):
   // This evaluates $\partial p^{[\delta]:\langle 0:\delta-1\rangle}/\partial p^{[d]:\langle k_0,\dots,k_{d-1}\rangle}$ = left side of (26)
   (a) if $\delta == d$: // We have a complete set of idxs to evaluate (28)
      i. if $(i_\delta = 0 \cap j_\delta = 0) \cup (i_\delta = n^{\langle\delta\rangle} - 2 \cup j_\delta = 3)$ for **at least** one $\delta = 0,\dots,d-1$:
         A. if $\mathcal{M}\langle i_0 - 1 + j_0 : i_{d-1} - 1 + j_{d-1}\rangle == \langle i_0 - 1 + k_0 : i_{d-1} - 1 + k_{d-1}\rangle$: return 2
         B. if $\mathcal{P}\langle i_0 - 1 + j_0 : i_{d-1} - 1 + j_{d-1}\rangle == \langle i_0 - 1 + k_0 : i_{d-1} - 1 + k_{d-1}\rangle$: return -1
         C. else: return 0
      ii. else: return 0
   (b) else: // Go deeper
      i. $\partial p^{[\delta+1]:\langle 0:\delta-1,0\rangle}/\partial p^{[d]:\langle k_0,\dots,k_{d-1}\rangle}$ = **iterate_deriv_p**(δ,d,$(j_0,\dots,j_\delta = 0,\dots,j_{d-1})$,$p^{[d]:\langle 0:d-1\rangle}$)
      ii. $\partial p^{[\delta+1]:\langle 0:\delta-1,1\rangle}/\partial p^{[d]:\langle k_0,\dots,k_{d-1}\rangle}$ = **iterate_deriv_p**(δ,d,$(j_0,\dots,j_\delta = 1,\dots,j_{d-1})$,$p^{[d]:\langle 0:d-1\rangle}$)
      iii. $\partial p^{[\delta+1]:\langle 0:\delta-1,2\rangle}/\partial p^{[d]:\langle k_0,\dots,k_{d-1}\rangle}$ = **iterate_deriv_p**(δ,d,$(j_0,\dots,j_\delta = 2,\dots,j_{d-1})$,$p^{[d]:\langle 0:d-1\rangle}$)
      iv. $\partial p^{[\delta+1]:\langle 0:\delta-1,3\rangle}/\partial p^{[d]:\langle k_0,\dots,k_{d-1}\rangle}$ = **iterate_deriv_p**(δ,d,$(j_0,\dots,j_\delta = 3,\dots,j_{d-1})$,$p^{[d]:\langle 0:d-1\rangle}$)
      v. return $\sum_{j_\delta=0}^{3} \frac{\partial f(x_\delta)}{\partial p_{j_\delta}} \frac{\partial p^{[\delta+1]:\langle 0:\delta-1,j_\delta\rangle}}{\partial p^{[d]:\langle k_0,\dots,k_{d-1}\rangle}}$
2. To start: **iterate_deriv_p**(0,d,$(j_0,\dots,j_{d-1})$,$p^{[d]:\langle 0:d-1\rangle}$)

2.3 The derivative with respect to x

We seek the derivative with respect to the k-th component x_k of $\boldsymbol{x}$:

$$\frac{\partial p^{[0]}}{\partial x_k} \tag{31}$$

Consider generally differentiating the left side of the recursion relation (19):

$$\frac{\partial p^{[\delta]:\langle 0:\delta-1\rangle}}{\partial x_k} \tag{32}$$

If $k = \delta$, then this may be evaluated using:

$$\frac{\partial p^{[k]:\langle 0:\delta\rangle}}{\partial x_k} = \frac{\partial f\left(x_k; p^{[k+1]:\langle 0:k-1,0\rangle}, p^{[k+1]:\langle 0:k-1,1\rangle}, p^{[k+1]:\langle 0:k-1,2\rangle}, p^{[k+1]:\langle 0:k-1,3\rangle}\right)}{\partial x_k} \tag{33}$$

where the points $p^{[k+1]:\langle 0:k\rangle}$ on the right hand side must be evaluated using (19) as before.

If $k < \delta$, then we must evaluate the derivative using (19):

$$\frac{\partial p^{[\delta]:\langle 0:\delta-1\rangle}}{\partial x_k} = \sum_{j_\delta=0}^{3} \frac{\partial f(x_\delta)}{\partial p_{j_\delta}} \frac{\partial p^{[\delta+1]:\langle 0:\delta\rangle}}{\partial x_k} \tag{34}$$

Starting at $\delta = 0$, we must therefore apply the recursion (34) k times:

$$\frac{\partial p^{[0]}}{\partial x_k} = \sum_{j_0,\dots,j_{k-1}} \left(\prod_{\alpha=0}^{k-1} \frac{\partial f(x_\alpha)}{\partial p_{j_\alpha}} \right) \frac{\partial p^{[k]:\langle 0:\delta\rangle}}{\partial x_k} \tag{35}$$

and then use (33):

$$\frac{\partial p^{[0]}}{\partial x_k} = \sum_{j_0,\dots,j_{k-1}} \left(\prod_{\alpha=0}^{k-1} \frac{\partial f(x_\alpha)}{\partial p_{j_\alpha}} \right) \frac{\partial f\left(x_k; p^{[k+1]:\langle 0:k-1,0\rangle}, p^{[k+1]:\langle 0:k-1,1\rangle}, p^{[k+1]:\langle 0:k-1,2\rangle}, p^{[k+1]:\langle 0:k-1,3\rangle}\right)}{\partial x_k} \tag{36}$$

This can be evaluated using the 1D result (10) for the second term; unfortunately, this requires $d-1-k$ levels of further recursion (19) to determine the $p^{[k+1]:\langle 0:k\rangle}$.

Also: do not forget that:

$$\frac{\partial p^{[0]}}{\partial \tilde{x}_k} = \frac{\partial p^{[0]}}{\partial x_k}\frac{\partial x_k}{\partial \tilde{x}_k} = \frac{\partial p^{[0]}}{\partial x_k}\left(s^{\langle k\rangle}_{i_k+1} - s^{\langle k\rangle}_{i_k}\right)^{-1} \tag{37}$$

since $\boldsymbol{x}$ refers to a fraction between $0, 1$.

2.3.1 Pseudocode

1. function **iterate_deriv**(δ,k,d,$\boldsymbol{x}$,$(j_0,\ldots,j_{k-1})$, $p^{[d]:\langle 0:d-1\rangle}$):
   // This evaluates $\partial p^{[\delta]:\langle 0:\delta-1\rangle}/\partial x_k$ = the left hand of (34)
   (a) if $\delta == k$: // Evaluate using (33)
      i. $p^{[k+1]:\langle 0:k-1,0\rangle}$ = **iterate**($k+1$, d, $\boldsymbol{x}$, $(j_0,\ldots,j_{k-1},j_k=0)$, $p^{[d]:\langle 0:d-1\rangle}$)
      // Recall that **iterate**($k+1,\ldots$) returns $p^{[k+1]}$
      ii. $p^{[k+1]:\langle 0:k-1,1\rangle}$ = **iterate**($k+1$, d, $\boldsymbol{x}$, $(j_0,\ldots,j_{k-1},j_k=1)$, $p^{[d]:\langle 0:d-1\rangle}$)
      iii. $p^{[k+1]:\langle 0:k-1,2\rangle}$ = **iterate**($k+1$, d, $\boldsymbol{x}$, $(j_0,\ldots,j_{k-1},j_k=2)$, $p^{[d]:\langle 0:d-1\rangle}$)
      iv. $p^{[k+1]:\langle 0:k-1,3\rangle}$ = **iterate**($k+1$, d, $\boldsymbol{x}$, $(j_0,\ldots,j_{k-1},j_k=3)$, $p^{[d]:\langle 0:d-1\rangle}$)
      v. return $\partial f\left(x_k; p^{[k+1]:\langle 0:k-1,0\rangle}, p^{[k+1]:\langle 0:k-1,1\rangle}, p^{[k+1]:\langle 0:k-1,2\rangle}, p^{[k+1]:\langle 0:k-1,3\rangle}\right)/\partial x_k$ // using (10)
   (b) else: // Recurse using (34)
      i. $\partial p^{[\delta+1]:\langle 0:\delta-1,0\rangle}/\partial x_k$ = **iterate_deriv**($\delta+1$,k,d,$\boldsymbol{x}$,$(j_0,\ldots,j_{\delta+1}=0,\ldots,j_{k-1})$,$p^{[d]:\langle 0:d-1\rangle}$)
      ii. $\partial p^{[\delta+1]:\langle 0:\delta-1,0\rangle}/\partial x_k$ = **iterate_deriv**($\delta+1$,k,d,$\boldsymbol{x}$,$(j_0,\ldots,j_{\delta+1}=0,\ldots,j_{k-1})$,$p^{[d]:\langle 0:d-1\rangle}$)
      iii. $\partial p^{[\delta+1]:\langle 0:\delta-1,0\rangle}/\partial x_k$ = **iterate_deriv**($\delta+1$,k,d,$\boldsymbol{x}$,$(j_0,\ldots,j_{\delta+1}=0,\ldots,j_{k-1})$,$p^{[d]:\langle 0:d-1\rangle}$)
      iv. $\partial p^{[\delta+1]:\langle 0:\delta-1,0\rangle}/\partial x_k$ = **iterate_deriv**($\delta+1$,k,d,$\boldsymbol{x}$,$(j_0,\ldots,j_{\delta+1}=0,\ldots,j_{k-1})$,$p^{[d]:\langle 0:d-1\rangle}$)
      v. return $\sum_{j_\delta=0}^{3}\frac{\partial f(x_\delta)}{\partial p_{j_\delta}}\frac{\partial p^{[\delta+1]:\langle 0:\delta\rangle}}{\partial x_k}$ // = right side of (34)
2. To start: **iterate_deriv**(0,k,d,$\boldsymbol{x}$,$(j_0,\ldots,j_{k-1})$,$p^{[d]:\langle 0:d-1\rangle}$)