

Optimization for Machine Learning in Python

└ The derivative

└ Derivation of the parabola derivative

$$\lim_{h \rightarrow 0} \frac{(x+h)^2 - x^2}{h} = \lim_{h \rightarrow 0} \frac{x^2 + 2xh + h^2 - x^2}{h} \quad (4)$$

$$= \lim_{h \rightarrow 0} \frac{2xh + h^2}{h} \quad (5)$$

$$= \lim_{h \rightarrow 0} \frac{h(2x + h)}{h} \quad (6)$$

$$= \lim_{h \rightarrow 0} 2x + h \quad (7)$$

$$= 2x \quad (8)$$

Derive on the board.

Derivate of a parabola:

$$\lim_{h \rightarrow 0} \frac{(x+h)^2 - x^2}{h} = \lim_{h \rightarrow 0} \frac{x^2 + 2xh + h^2 - x^2}{h} \quad (9)$$

$$= \lim_{h \rightarrow 0} \frac{2xh + h^2}{h} \quad (10)$$

$$= \lim_{h \rightarrow 0} \frac{h(2x + h)}{h} \quad (11)$$

$$= \lim_{h \rightarrow 0} 2x + h \quad (12)$$

$$= 2x \quad (13)$$

Optimization for Machine Learning in Python

└ The derivative

└ The derivate of a polynomial

What is the derivative of the function $f(x) = x^2$?

$$\frac{df(x)}{dx} = 2x^{2-1}$$

(14)

Derivate of a polynomial $f(x) = x^n$ [DFO20]:

$$\frac{df(x)}{dx} = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} = \lim_{h \rightarrow 0} \frac{(x+h)^n - x^n}{h} \quad (15)$$

$$= \lim_{h \rightarrow 0} \frac{\sum_{i=0}^n \binom{n}{i} x^{n-1} h^i - x^n}{h} \quad (16)$$

$$= \lim_{h \rightarrow 0} \frac{\sum_{i=1}^n \binom{n}{i} x^{n-1} h^i}{h} \quad (17)$$

$$= \lim_{h \rightarrow 0} \sum_{i=1}^n \binom{n}{i} x^{n-1} h^{i-1} \quad (18)$$

$$= \lim_{h \rightarrow 0} \left(\binom{n}{1} x^{n-1} \right) + \sum_{i=2}^n i \binom{n}{i} x^{n-i} h^{i-1} \quad (19)$$

$$= \frac{n!}{1!(n-1)!} x^{n-1} = nx^{n-1}. \quad (20)$$

Optimization for Machine Learning in Python

└ Optimization in many dimensions

└ The gradient

- Gradients point in the steepest ascent direction.
- To find the gradient, we must compute the partial derivate with respect to every input.
- A vector collects all derivatives.

The gradient lists partial derivatives with respect to all inputs in a vector. For a function $f: \mathbb{R}^n \rightarrow \mathbb{R}$ of n variables the gradient $\nabla f: \mathbb{R}^n \rightarrow \mathbb{R}^n$ is defined as

$$\nabla f = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix}. \quad (46)$$

Optimization for Machine Learning in Python

└ Optimization in many dimensions

└ The gradient of the Rosenbrock function

Recall the Rosenbrock function:

$$f(x, y) = (a - x)^2 + b(y - x^2)^2 \quad (52)$$

$$\nabla f(x, y) = \begin{pmatrix} -2a + 2x - 4byx + 4bx^3 \\ 2by - 2bx^2 \end{pmatrix} \quad (53)$$

On the board, derive:

$$f(x, y) = (a - x)^2 + b(y - x^2)^2 \quad (54)$$

$$= a^2 - 2ax + x^2 + b(y^2 - 2yx^2 + x^4) \quad (55)$$

$$= a^2 - 2ax + x^2 + by^2 - 2byx^2 + bx^4 \quad (56)$$

$$\Rightarrow \frac{\partial f(x, y)}{\partial x} = -2a + 2x - 4byx + 4bx^3 \quad (57)$$

$$\Rightarrow \frac{\partial f(x, y)}{\partial y} = 2by - 2bx^2 \quad (58)$$

Temporary page!

\LaTeX was unable to guess the total number of pages correctly. As there was some unprocessed data that should have been added to the final page this extra page has been added to receive it.

If you rerun the document (without altering it) this surplus page will go away, because \LaTeX now knows how many pages to expect for this document.