LECTURE 4

DATE: 18 OCTOBER 2021

4. Optimization for functions of several variables I: Least Squares & Machine Learning

The Fréchet differential of a function $f: \mathbb{R}^n \to \mathbb{R}$ at a point $x \in \mathbb{R}^n$ is a linear function T s.t.

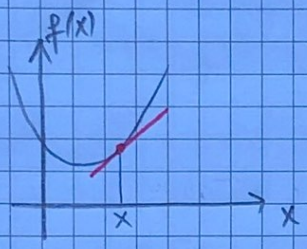
$$\lim_{y \to x} \frac{|f(y) - f(x) - T(x-y)|}{\|y-x\|} = 0$$

Meaning of Fréchet diff:

You approx. f by a linear function (locally in x!)

(Notation $df(x)(z) \overset{\text{def}}{=} T(z)$

↑ linear function in the limit above



Theorem 1:

If all partial derivatives of f are continuous at x then f is Fréchet diffable at x and $df(x)(z) = \nabla f(x) \cdot z \quad \forall z \in \mathbb{R}^n$

Furthermore if all $\frac{\partial^2}{\partial x_i \partial x_j} f$ are cont. at x then second Fréchet diff is a quadratic function (form) with matrix:

$$a_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}(x) \quad \text{(Hesse of } f\text{)}$$

Analogy:

$d = 1$	$d > 1$
$f: \mathbb{R} \to \mathbb{R}$	$f: \mathbb{R}^d \to \mathbb{R}$ several vars
f'	$\nabla f = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_d} \right)$
f''	$H_f = \left(\frac{\partial^2 f}{\partial x_i \partial x_j} \right)_{i,j=\overline{1,d}}$
	$\| \text{not} \quad \nabla^2 f$

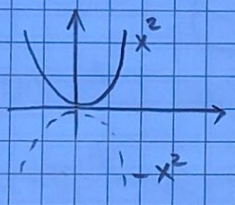
§ 4.1. Optimization for functions of several variables

For function of a single var ($d=1$):

If $\begin{cases} f'(x^*) = 0 \text{ and } f''(x^*) > 0 \Rightarrow x^* \text{ minimum} \\ f'(x^*) = 0 \text{ and } f''(x^*) < 0 \Rightarrow x^* \text{ maximum} \end{cases}$ (local)

How to remember this: think about the simplest fct.

$f(x) = x^2$ (or $f(x) = -x^2$)



$f''(x) = 2 > 0$ (or $f''(x) = -2 < 0$)

Theorem 2 (FERMAT):

$f: \mathbb{R}^d \to \mathbb{R}$, Fréchet diffable in $x^* \in \mathbb{R}^d$

If x^* is a local min / max then $\nabla f(x^*) = 0$.

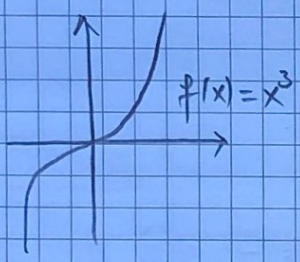
(Optimization = find min / max)

The Fermat: classical approach to optimization:
compute f', find x^* s.t. $f'(x^*) = 0$. Then, compute (establish) sign of $f''(x^*) \lesseqgtr 0$

Remarks:

There exist critical points ($f' = 0$ or $\nabla f = 0$) which are neither minima nor maxima.

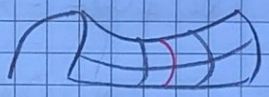
$d = 1$



$d = 2$

$f(x_1, x_2) = x_1^2 - x_2^2$

($(0,0)$ saddle point)



Positivity for functions of several variables?

Def: A quadratic function (form) $Q: \mathbb{R}^m \to \mathbb{R}$

(with matrix $A = (a_{ij})$)

is
- positive definite if $Q(x) > 0 \quad \forall x \in \mathbb{R}^d \setminus \{0_{\mathbb{R}^d}\}$
- negative definite if $Q(x) < 0 \quad \forall x$
- indefinite if $Q(x_1) > 0, Q(x_2) < 0$

also we say that Q is

positive semi definite if $Q(x) \geq 0$

negative semi definite if $Q(x) \leq 0$

Theorem 3 (SYLVESTER) : Crit for positive / negative definite if $A = (a_{ij})$ is the matrix of Q

Then

- $a_{11} > 0$ $\quad \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} > 0, \dots, \begin{vmatrix} a_{11} & \cdots \\ \vdots & a_{dd} \end{vmatrix} > 0$

$\Rightarrow$ Q is positive definite

- $a_{11} < 0$, $\quad \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} > 0, \dots, (-1)^d \begin{vmatrix} a_{11} & \cdots \\ \vdots & a_{dd} \end{vmatrix} > 0$

(signs alternate)

$\Rightarrow$ Q is negative definite

- otherwise criterium is not effective

Theorem 4

$f : \mathbb{R}^d \to \mathbb{R}$ twice Fréchet diffable in x^*

if $\nabla f(x^*) = 0_{\mathbb{R}^d}$ and

$H_f(x^*) = \nabla^2 f(x^*)$ is $\begin{cases} \text{positive definite} \Rightarrow x^* \text{ min} \\ \text{negative definite} \Rightarrow x^* \text{ max} \end{cases}$

§ 4.2. The Least Squares Method (GAUSS)

Given:

- a set of data (measurement)

x	$x_1 \dots x_i \dots x_m$
y	$y_1 \dots y_i \dots y_m$

- a model $y(x) = ax + b$
  ↳ = a parametrized family of functions

Goal: Find a^*, b^* such that $a^*x + b^*$ is the best fit for the given data.

This is an Optimization Problem!

$$E(a,b) = \sum_{i=1}^{m} (y_i - (ax_i + b))^2 \to \min$$

("Least squares")

minimize w.r.t. a, b!

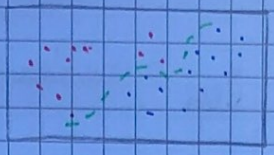
Remark: E is "quadratic" $\Rightarrow \nabla^2 E$ positive definite

So you only have to find a^*, b^* such that

$$\nabla E(a^*, b^*) = 0 \iff \begin{cases} \frac{\partial E}{\partial a}(a^*, b^*) = 0 \\ \frac{\partial E}{\partial b}(a^*, b^*) = 0 \end{cases}$$

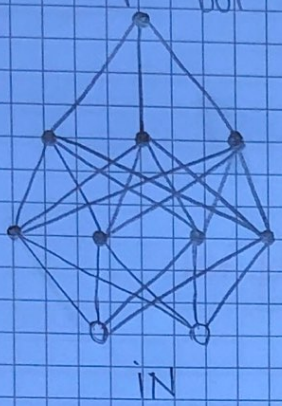
§ 4.3. Deep Learning

Example of Classification Problem



Want: train computer to "learn" frontier between blue and red

computer = artificial Neural Network



Output Layer (1 neuron in my case)

} Hidden Layer (active neuron)

Input Layer

each active neuron has an SIGMOID activation function

$$\phi(x) = \frac{1}{1+e^{-x}}$$ (x = input of neuron)

($\phi(x)$ = output)

connected neurons

$$\phi\left(\sum_j \underbrace{w_{ij}}_{\text{weights}} \underbrace{x_j}_{\text{input from neurons on previous layer}} + \underbrace{b_i}_{\text{bias}}\right) = \text{output of neuron } "i"$$

$$y_{out} = \overline{f}(x_{in})$$

entire NN = function connecting IN to OUT

In our example IN: $x_{IN} = (x1_{IN}, x2_{IN})$

↑ coordinates of a point on the map

Classification OUT: y_{OUT} = number $\in [0, 1]$

0 = you are in the red zone

1 = you are in the blue zone

Training the NN given the labeled data set

(x^i, y^i) $\quad i = \overline{1, m}$

points ↑ ↑ labels $\in \{0, 1\}$

→ Apply Least Squares to

IN to OUT function of NN

$$E(\underline{\underline{w}}, \underline{b}) = \sum_{i=1}^{m} \left(y^i - \overline{f}_{\underline{\underline{w}}, \underline{b}}(x^i)\right)^2 \longrightarrow \min$$

parameters = weights $\underline{\underline{w}}$ and biases $\underline{b}$

HOW to minimise E?

Gradient RESCENT! (Algorithm)

for $W = (\underline{\underline{w}}, \underline{b})$

"Learning rate"

(GD) $W_{n+1} = W_n - \lambda \nabla E(W_n)$