__/__/____

Introduction to Numerical Optimization:

Week 1

Data everywhere

Data science → Extracting useful info from raw data.

Uses ML Tools.

Data Engineering / Data science / Data Analyst

How to collect and store data → Making decision From that data. → Make visualisation From data.

→ Backbone of Datascience:

- Mathematics
  1. Linear Algebra
  2. Calculus
  3. Optimization.
  4. Statistics.
- Computer - Algo, Computation, Storage.

__/__/____

→ Broad classification of ML prob:

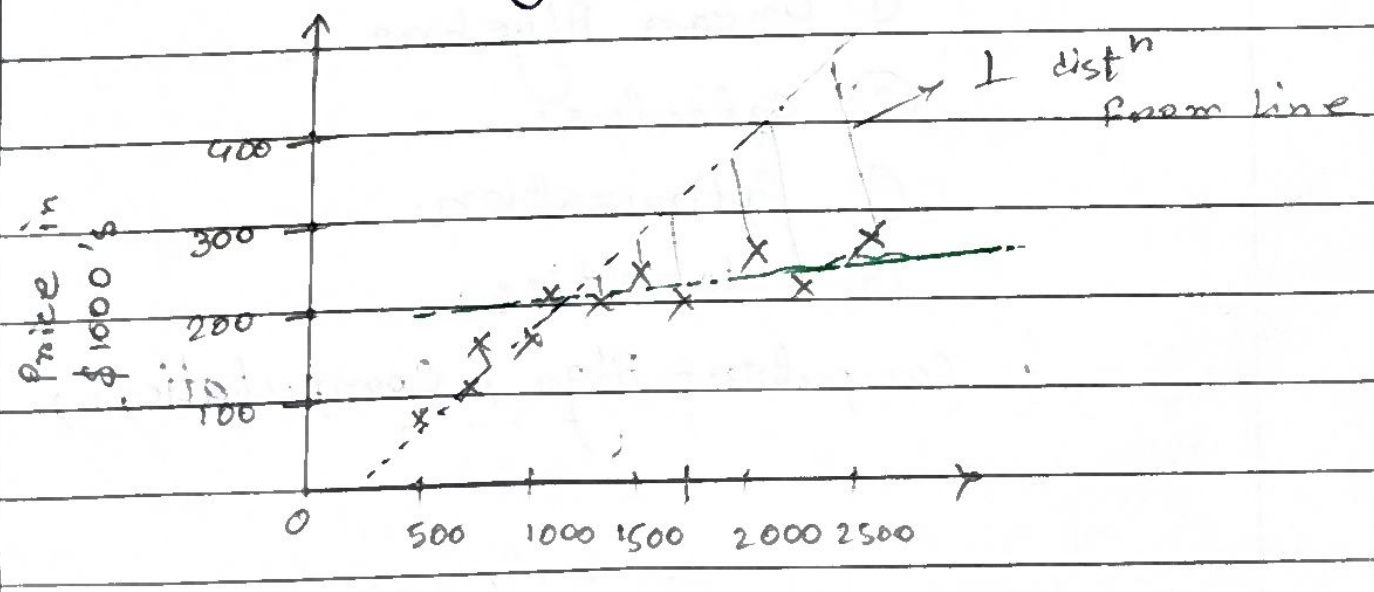
- Model fitting
- Classification.

→ Application:

- Weather Forecast
- recommendations
- Computer vision
- Pattern Recognition.

Why Optimization Matters?

- Regression Model
- Suppose a few data point $\{(x_i, y_i)\}$ have been given in the plane.



__/__/____

we collect the data of some house of price and size and plot in graph.

Based on the price we predict the price or get an approx range of it.

- Suppose we guess there is a linear relation.

$$price(p) = \alpha \cdot size(s) + \beta$$

↳ similar to straight line Eq^n.

Note: This is not the absolute model to predict price. There are more model.

Different Model give different result.

There are NO unified model

There is NO unified prediction

Q. Which α and β to choose?

Bring in optimization perspective

↳ Ok, I have a model, with that model, can I pass it through all the data point

↳ Typically very unlikely

__/__/____

of course, there will be error.
↳ Mathematical one

We can take 1 distn from the line and sum them up, a valid error indeed.

We can also use (Mis)feet

price = (α)(size) + β

direct correlation b/w price ↔ size.

$$\xi (\alpha(\text{size}) + \beta - \text{price})^2$$

↓

some of the term +ve and some -ve which cancle out so, we square.

- Typically we need/want least error in prediction.

- This is achieved by finding a straight line $y = \alpha x + \beta$ such that.

__/__/____

$f(a,b) = \sum_{i=1}^{n} (ax_i + b - y_i)^2$ is minimized.

Concept:

- Optimising problem is typically minimising or maximising Fn of some veriables.

Observative (loss) Fn
Decision veriables (Parameters)
Constraints (optional)

minimize $f(x)$
subject to $x \in S$ and
where $f: \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a map, $S \subseteq \mathbb{R}^n$ is a nonempty constraint set.

Eg: If we go from class to Mess and have three routes, not always the shortest route is better.
If the objective is workout. then take the longest route.

__/__/____

Types of Problem:

- Unconstraint vs constraint
- Convex vs Non-convex

Unconstraint	Constraint
↓	↓
We give a model a whole playground and tell them to figure out themselves, what the patterns are	We give a model a whole playground with some directions/hint which make the model recognise patterns with our guidance

Q. Why convexity Matters? → we will look into it later

convex prob → easy prob.

non-convex prob → extreamly Heard prob.

__/__/__

Example:

- Geometric problem: Given a fixed perimeter, what is the largest area that can be covered?

suppose, we take rectangle → L, B

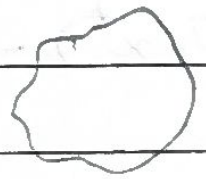
maximize $f(L,B) = L \times B$

subject to $2(L+B) = 10$

$S = \{(L,B) \in R^2 : 2(L+B) = 10\}$

a 5

→ 2(a+b)



(→ We use calculus. For the area

__/__/____

Solution Concept:

Solution

↳ ~~f(x)~~

In optimization we want a point x^* such that

$f(x^*) \leq f(x)$ for all x

when it is a minimization prob.

~~x^* ⟷ Minimizer~~

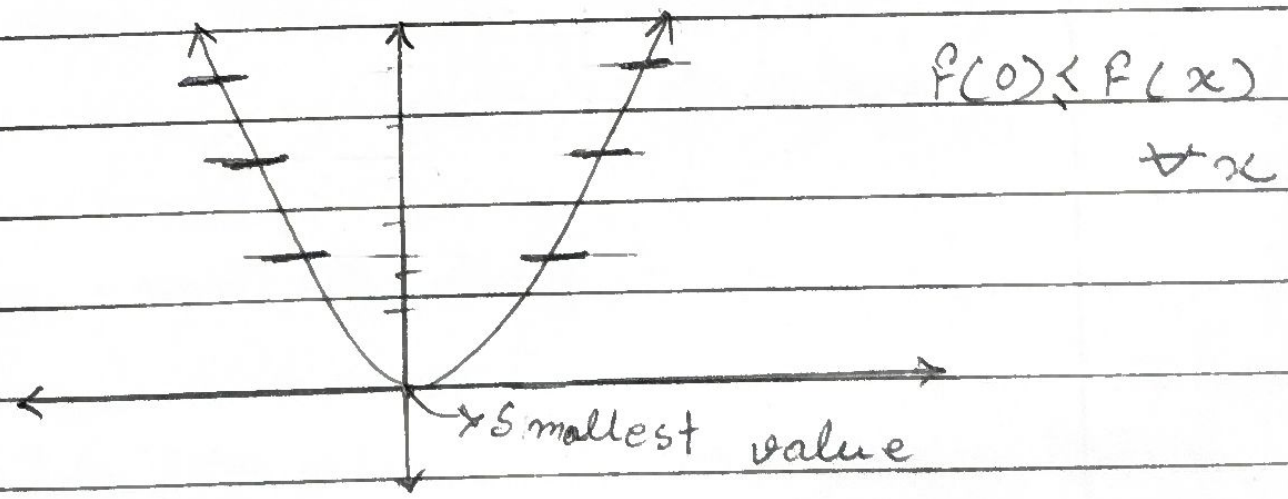
~~$f(x^*)$ ⟷ Minimum value~~

x^* → Minimizer

$F(x^*)$ → Minimum value.

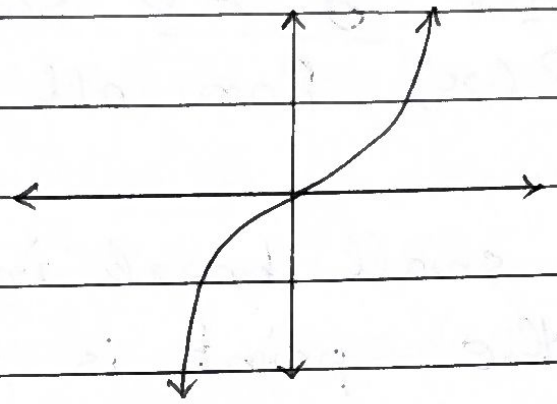
Eg:

minimize $f(x) = x^2$ over $S = \mathbb{R}$



__/__/____

if $f(x) = x^3$



The minimum is

$-\infty$

↳ unknown

so, we say

it doesn't

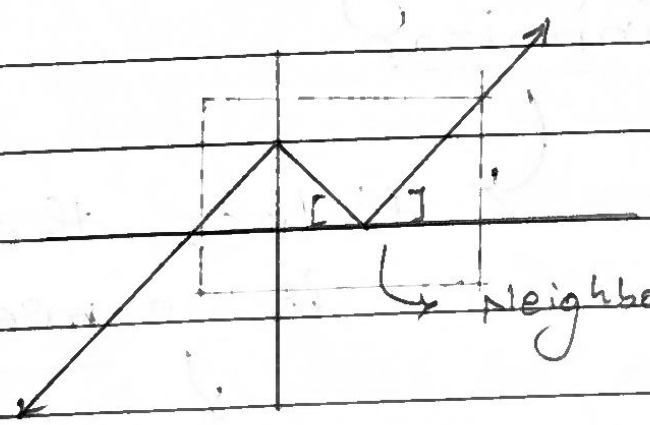
have minimize

or even maximizer

if we see other

side of the graph

$$f(x) = \begin{cases} x+2 & x \leq 0 \\ |2-x| & x \geq 0 \end{cases}$$



Here we use

Local Minimizer

$f(x) \geq f(2)$ $\forall x$ around

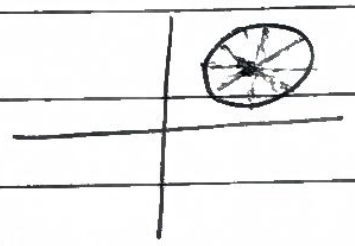
$\forall x \in [1, 3]$

Note: There is a slight break in in the almost continuous function.

__/__/____

- A point $x^* \in S$ is called a local minimizer if there exist $\delta > 0$ such that $f(x^*) \leq f(x)$ For all $x \in B_\delta(x^*) \cap S$

$\delta \rightarrow$ that small break in the Function and the point is typically



Why it is ~~not~~ Important ?

→ Most of the numarical Algorithm we will study we have Local Minimizar.

↳ Unless the problem is convex problem

↓

convexity will gurentee that local minimizar is global minimizar

__/__/____

Important thing in optimization:

- When can we say sol^n exist?

→ When there is global Minimizer, local is considered in the absence of global minimizer and presence of convexity.

- If a pt is sol^n, how do we know it?

- How to Find Sol^n? → Main Topic of this course

↓

Numarically

Lecture Takeaway: Numarical optimization

↓

Does Sol^n exist,
What kind of Sol^n (local/Global),
How to Find it Numarically?

Data Science, its Back Bone, Application, Classification:

Most ML Training prob are solve via numarical optimization of a loss fn.

__/__/__

A review of Calculus and Linear Algebra

- Derivative
- $F: R \to R$, the derivative $F'(x_0) = \frac{d}{dx} F(x_0)$

$$= \lim_{t \to 0} \frac{F(x_0+t) - F(x_0)}{t}$$

- calculus of derivative.
  - $(F \pm g)'(x) = F'(x) \pm g'(x)$
  - $(Fg)'(x) = F'(x)g(x) + g'(x)F(x)$
  - $(1/g)'(x) = \frac{g'(x)}{(g(x))^2}$
- $F''(x) = (F')'(x)$

$F(x) = \sin(x)$	$F(x) = x^2$
$F'(x) = \cos(x)$	$F'(x) = 2x$
$F''(x) = -\sin(x)$	$F''(x) = 2.$

Taylor's Expansion Theorem:

- If a fn is differentiable at around point x_0

$$f(x) \approx$$

1st order derivative.

If x is ve... this is a very ... of $f(x)$

double differentiable at around

$$+ f'(x_0)(x-x_0) + \frac{1}{2} f''(x_0)(x-x_0)^2$$

$$f(x) \approx f(x_0) + f'(x_0)(x-x_0)$$

- If x is very close to x_0 this is a very good **approximater** of $f(x)$

$$f(x) \approx f(x_0) + f'(x_0)(x-x_0) + \frac{1}{2} f''(x_0)(x-x_0)^2$$

- If double derivative exist
- x is very close to x_0.

__/__/__

Eg: $f(x) = e^x$, $x_0 = 0$

1st order: $e^x \approx 1 + x$

2nd order: $e^x \approx 1 + x + \frac{x^2}{2}$

Differentiability of Function of several variable:

. For a function $F: \mathbb{R}^n \to \mathbb{R}$, the partial derivative is used,

F at x with respect to x_i

fix one of the dimensions

$\mathbb{R}^2 \to$ ↑ →

$\mathbb{R}^3 \to$ ↑ →

$$\frac{\partial f}{x_i}(x) = \lim_{t \to 0} \frac{f(x + te_i) - f(x)}{t}$$

Eg: $f(x, y) = x^2 + xy + y^2$

$\frac{\partial f}{\partial x} = 2x + y.$

$\frac{\partial f}{\partial y} = x + 2y.$

$x + te_i$ is actually a vector sum.

$$\begin{bmatrix} 2 \\ 3 \end{bmatrix} + t \begin{bmatrix} 4 \\ 5 \end{bmatrix} = \begin{bmatrix} 2+4t \\ 3+5t \end{bmatrix}$$

=

__/__/____

- Derivative of $F = \underbrace{\nabla F(x)}_{\text{Gradient}}$

$$= \begin{bmatrix} \frac{\partial F}{\partial x_1} \\ \vdots \\ \frac{\partial F}{\partial x_n} \end{bmatrix}$$

→ Collection of all the partial derivatives.

→ vector.

For the above Eg:

$$\nabla F(x) = \begin{bmatrix} 2x+y \\ x+2y \end{bmatrix}$$

Let,

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

__/__/____

- Second-order partial derivative:

$$\frac{\partial^2 f}{\partial x_i \partial x_j} = \frac{\partial}{\partial x_i}\left(\frac{\partial f}{\partial x_j}\right)$$

- Hessain matrix: all second-order partial derivatives.

$$H_f(x) = \nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

Eg: $f(x,y) = x^2 + yx + y^2$

$$\frac{\partial^2 f}{\partial x \partial y} \Big/ \frac{\partial^2 f}{\partial x^2} \Big/ \frac{\partial^2 f}{\partial^2 y}$$

$$H(x,y) = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

- A matrix is a rectangular arrangement of numbers, symbols or expressions written in rows and columns, usually enclosed with brackets.

Taylor's theorem of several Variables:

- For a function $f: \mathbb{R}^n \to \mathbb{R}$, the derivative of f at x, denoted by $\nabla f(x)$ is a vector such that.

Dot product

Some term

$$f(x+h) = f(x) + \langle \nabla f(x), h \rangle + o(\|h\|)$$

$$f(x) = f(x_0) + f'(x_0)(x - x_0)$$

$$y = \frac{(y - x_0)}{h} + x_0$$

scalar analog

Very very small term

Roughly,

$$f(x+h) \approx f(x) + \langle \nabla f(x), h \rangle$$

__/__/__

- For a fn $f: R^n \rightarrow R$, the second derivative of f at x, denoted by $\nabla^2 f(x)$ is a Matrix such that,

$$f(x+h) = f(x) + \langle \nabla f(x), h \rangle + \frac{1}{2!} \langle h, \nabla^2 f(x) h \rangle + o(\|h\|^2)$$

$\langle \nabla f(x), h \rangle \rightarrow$ dot$(\nabla f(x), h)$

$\langle h, \nabla^2 f(x) h \rangle \rightarrow$ dot$(h, \nabla^2 f(x) h)$

Roughly,

$$f(x+h) \approx f(x) + \langle \nabla f(x), h \rangle + \frac{1}{2!} \langle h, \nabla^2 f(x) h \rangle$$

Note : ① This is a quadratic function.
② We need to be careful about linear fn and quadratic fn
As, they will be the function we will be frequently need data science course.

__/__/____

Dot Product :

- For $u, v \in \mathbb{R}^n$: $u \cdot v = \sum_{i=1}^{n} u_i v_i$

- $u = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$, $v = \begin{bmatrix} 3 \\ 4 \end{bmatrix}$

$$u v = (1 \cdot 3) + (2 \cdot 4) = 11$$

Transpose : Flips rows and columns.

$$(A^T)_{ij} = A_{ji}$$

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \quad A^T = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$$

__/__/__

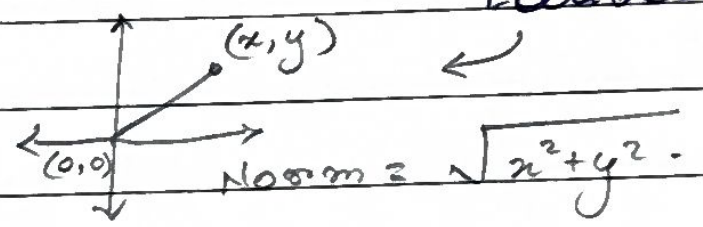
* Norm (length of a vector)

- Denoted by $\|v\|$
- For $\vec{v} = (v_1, v_2 \dots\dots, v_n)$

$$\|v\|_2 = \sqrt{v_1^2 + v_2^2 + \dots\dots + v_n}$$

↓

Euclidean Norm. or L2 Norm.



$$\|v\| = |v_1| + |v_2| + \dots\dots + |v_n|$$

- Measures the magnitude (length) of a vector.

Dot Product :

$$\|v\|_2 = \underset{1\times n \; n\times 1}{v^T v}.$$