Erasmus+ lecture on Machine Learning, November 13, 2023

Simple example

$$f(\mathbf{x}) = \frac{1}{2}x^T A x - x^T b$$

β_0

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

β_1

$$f(x_1, x_2) = \frac{1}{2}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^T \begin{bmatrix} 2 & 1 \\ 1 & 20 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} - \begin{bmatrix} 5 \\ 3 \end{bmatrix}^T \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$= x_1^2 + x_1 x_2 + 10 x_2^2 - 5x_1 - 3x_2$$

$$\frac{\partial f}{\partial x_1} = 0 = 2x_1 + x_2 - 5$$

$$\frac{\partial f}{\partial x_2} = 0 = x_1 + 20x_2 - 3$$

$$x_1 = 97/39 \approx 2.49$$

$$x_2 = 1/39 \approx 0.026$$

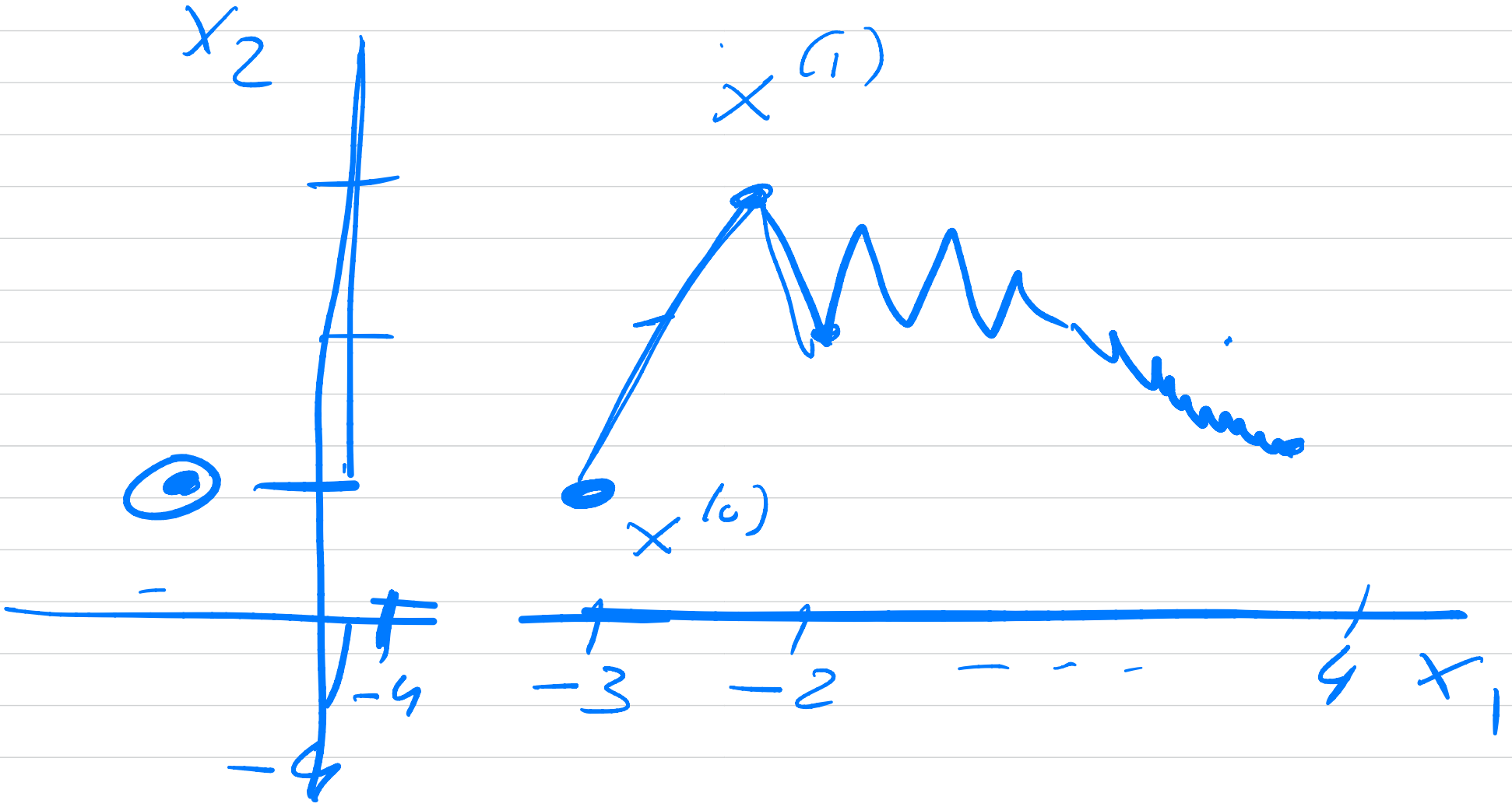
$$f(x_0) \geq f(x_1) \geq f(x_2) \geq \cdots \geq f(x^*)$$

$$x^{(0)} = \begin{bmatrix} -3 \\ 1 \end{bmatrix}$$

$$x_i^{(1)} = x_i^{(0)} - \gamma_0 \nabla f(x_i^{(0)})$$

$$i = \{1, 2\}$$

$$x_i^{(k+1)} = x_i^{(k)} - \gamma^{(k)} \nabla f(x_i^{(k)})$$


x_2
$x^{(1)}$
$x^{(0)}$
-4
-4
-3
-2
4
x_1

Gradient with momentum

$$m\frac{d^2x}{dt^2} + \mu\frac{dx}{dt} = -\nabla V(x)$$

$$\frac{d^2x}{dt^2} \approx \frac{x(t+\Delta t) + x(t-\Delta t) - 2x(t)}{(\Delta t)^2}$$

$$= \frac{x_{t+\Delta t} + x_{t-\Delta t} - 2x_t}{(\Delta t)^2}$$

$(x(t+\Delta t),\ x(t-\Delta t))$

$$\frac{dx}{dt} \approx \frac{x_{t+\Delta t} - x_t}{\Delta t}$$

$$\frac{m\left(x_{t+\Delta t} + x_{t-\Delta t} - 2x_t\right)}{(\Delta t)^2} + \frac{\mu\left(x_{t+\Delta t} - x_t\right)}{\Delta t} = -\nabla V(x)$$

Define $\Delta x_{t+\Delta t} = x_{t+\Delta t} - x_t$

$\Delta x_t = x_t - x_{t-\Delta t}$

$$\Delta x_{t+\Delta t} = \gamma \Delta x_t + \delta - \frac{(\Delta t)^2}{m + \mu \Delta t} \nabla V(x)$$

$$\Delta x_{t+\Delta t} = -\gamma \nabla V(x) + \delta \underbrace{\Delta x_t}_{x_t - x_{t-\Delta t}}$$

$$\begin{cases} \beta^{(i+1)} = \beta^{(i)} - \eta \nabla C(\beta^{(i)}) \end{cases}$$

$x_t \rightarrow \beta^{(i)} \qquad x_{t+\Delta t} \rightarrow \beta^{(i+1)}$

$\nabla V \rightarrow \nabla C(\beta^{(i)}) = g(\beta^{(i)}) = g$

$$\beta^{(i+1)} = \beta^{(i)} - \eta g + \underline{\delta}(\beta^{(i)} - \beta^{(i-1)})$$

hyperparameter

$\delta \in [0,1]$

algorithm

require learning $\eta(t)$

— | — momentum parameter s

require initial β, v

while stopping criterion not reached

- compute g
- compute velocity update $v^{(i)} = s(\beta^{(i)} - \beta^{(i-1)}) - \eta g$
- apply update $\beta^{(i+1)} = \beta^{(i)} + v^{(i)}$

end while

η :

- constant and choose a grid of values $\eta = [10^{-5}, 10^{-4}, 10^{-3}, \ldots]$

- exponential decay

$$\eta_k = \eta_0 \exp(-k\eta_{k-1})$$

k could be number of iterations

- linear

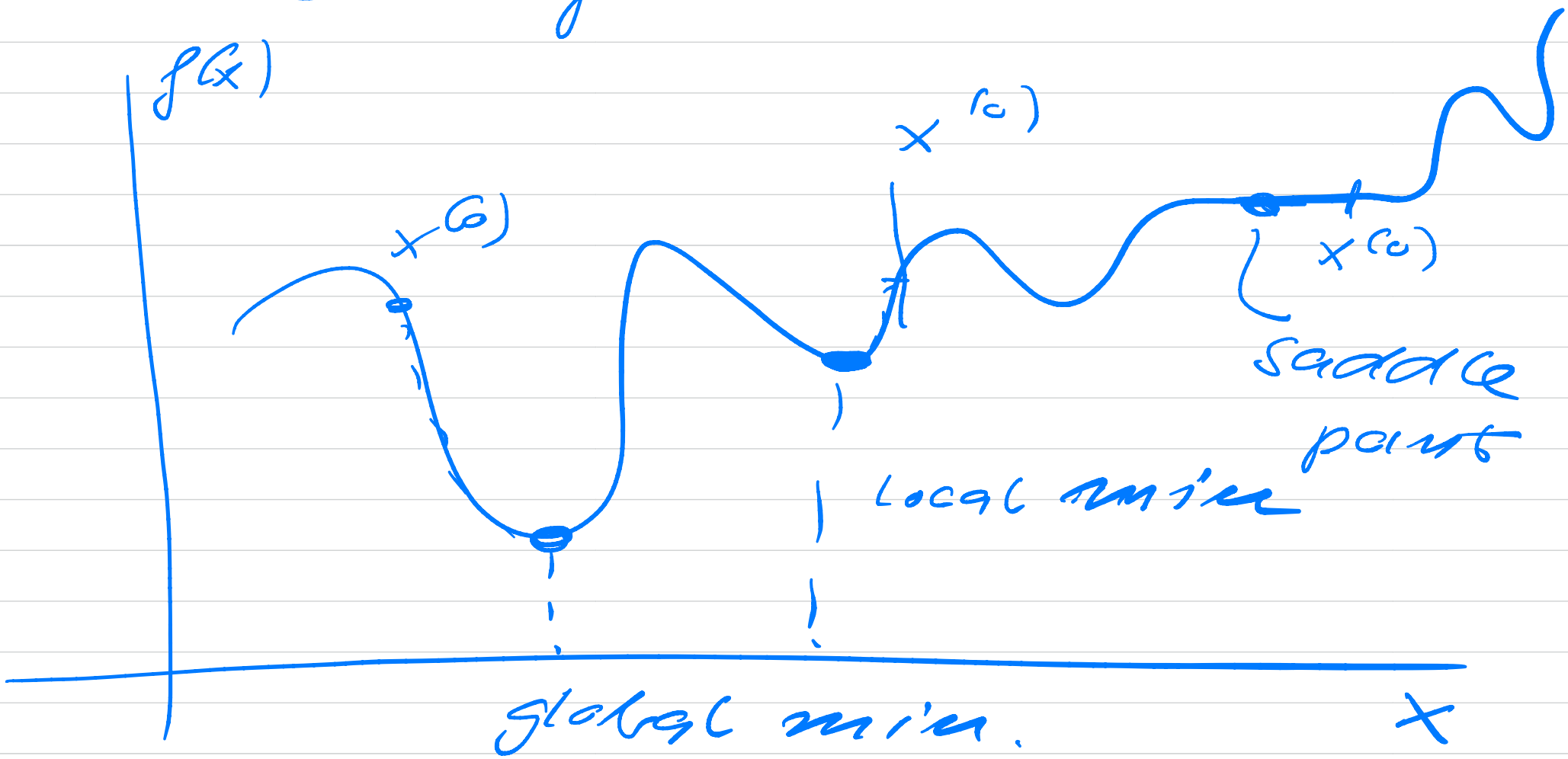
$$\eta_k = (1-\alpha)\eta_0 + \alpha\eta_T$$

$$\alpha = k/\text{const}$$

$$\eta_T \approx \frac{1}{100}\eta_0$$

Adagrad, RMSprop, Adam

stochastic gradient descent.



Split data in mini'batches

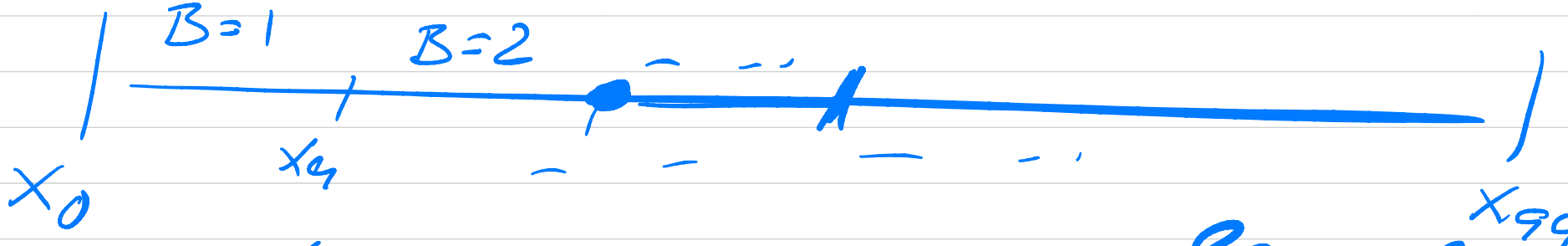
$n = 100$

$M = 10$ mini'batches

every mini'batch has

$m = n/M$ points (10 points)

$$D_{B_m} = \frac{1}{m} \sum_{i=1}^{m} DC(x_i)$$



repeat a given number of times = # epochs.

$n = 1000$

OLS : $g \propto \underbrace{X^T X}\beta - X^T y$

$X \in \mathbb{R}^{n \times n}$

FLOPs $\propto n^3 = 10^9$

Batch has only 10 points

$x^T x$ for a mini batch $(10^2)^3$

~ 1000 FLOPs

we repeat this 100 times.

$10^2 \times 10^3 \sim 10^5$ FLOPs.