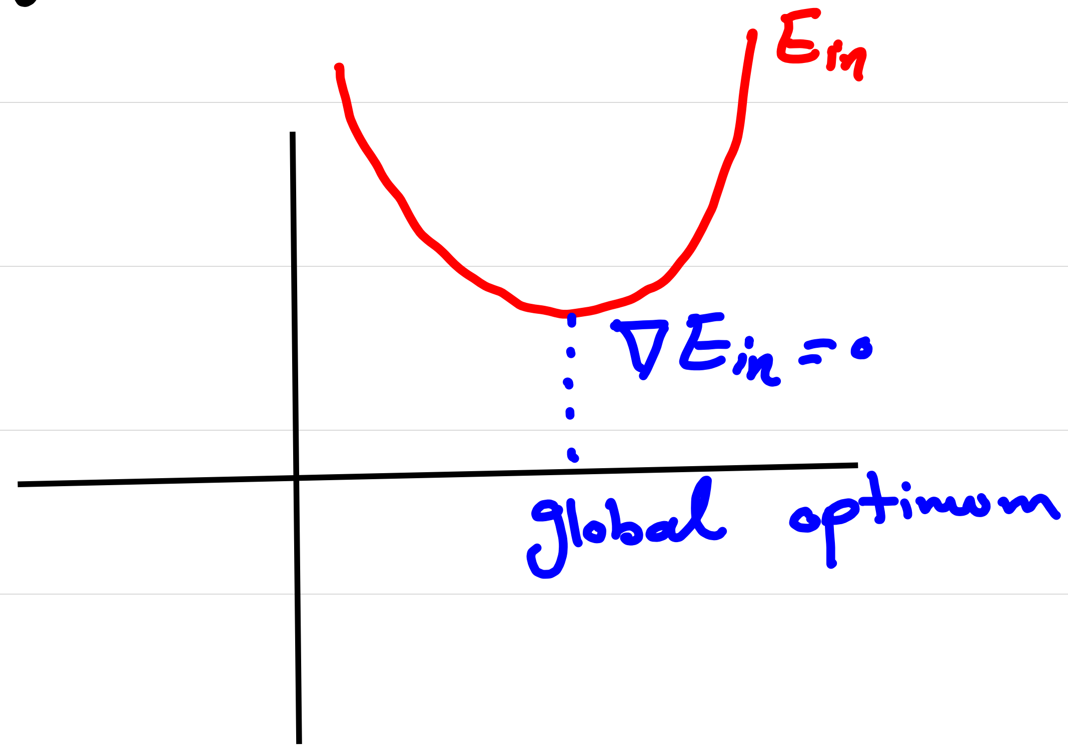
Week04 - Part03

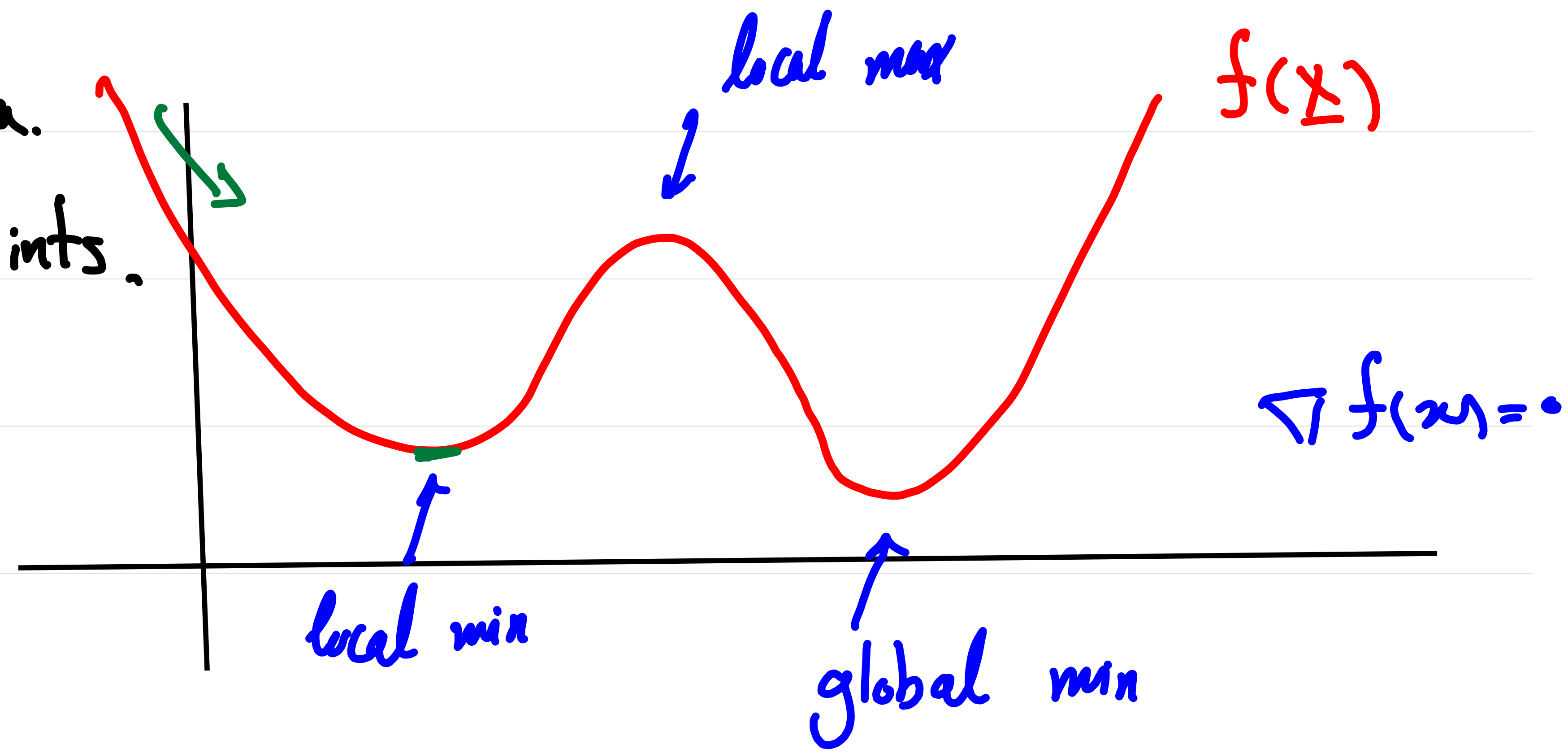
So far: For Convex functions, GD Converges to the optimal point.

- E_{in} for linear / Logistic Regression is Convex.

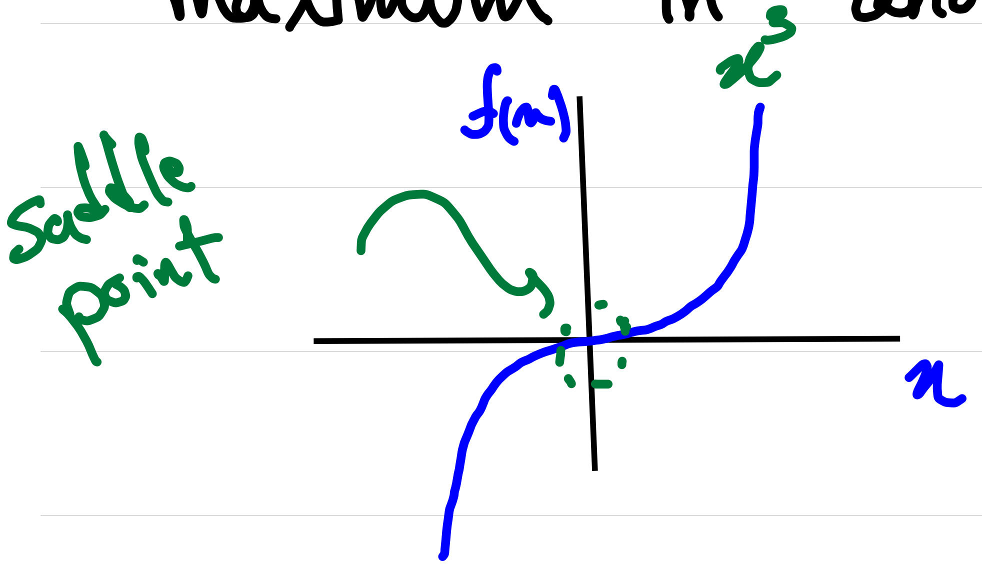


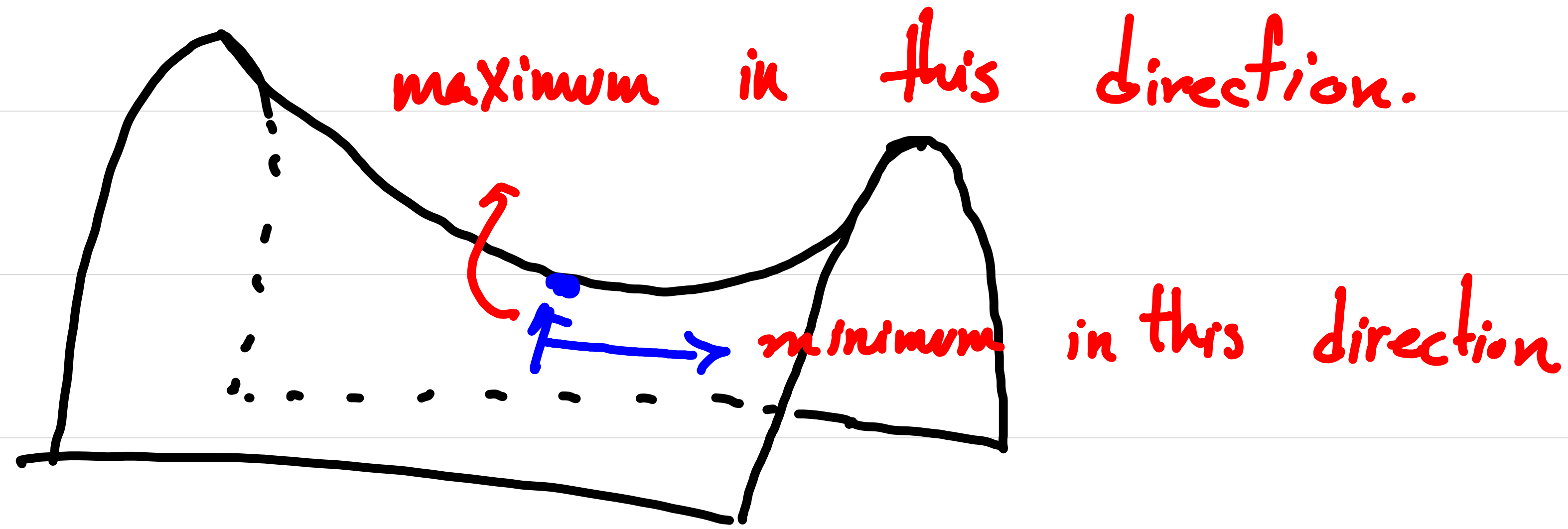
Today: Non-convex functions

- We often have multiple extrema.
- We may also have saddle points.



- A saddle point is minimum in some direction, and maximum in another direction.





- GD would have very slow progress near saddle points
  - ↳ The preset # of iterations will run out

- In high-dimensional space, it's highly likely to have saddle points.
- Most often, we are dealing with high_dimensional spaces.
  - ↳ How to use GD in this high-dimensional spaces?

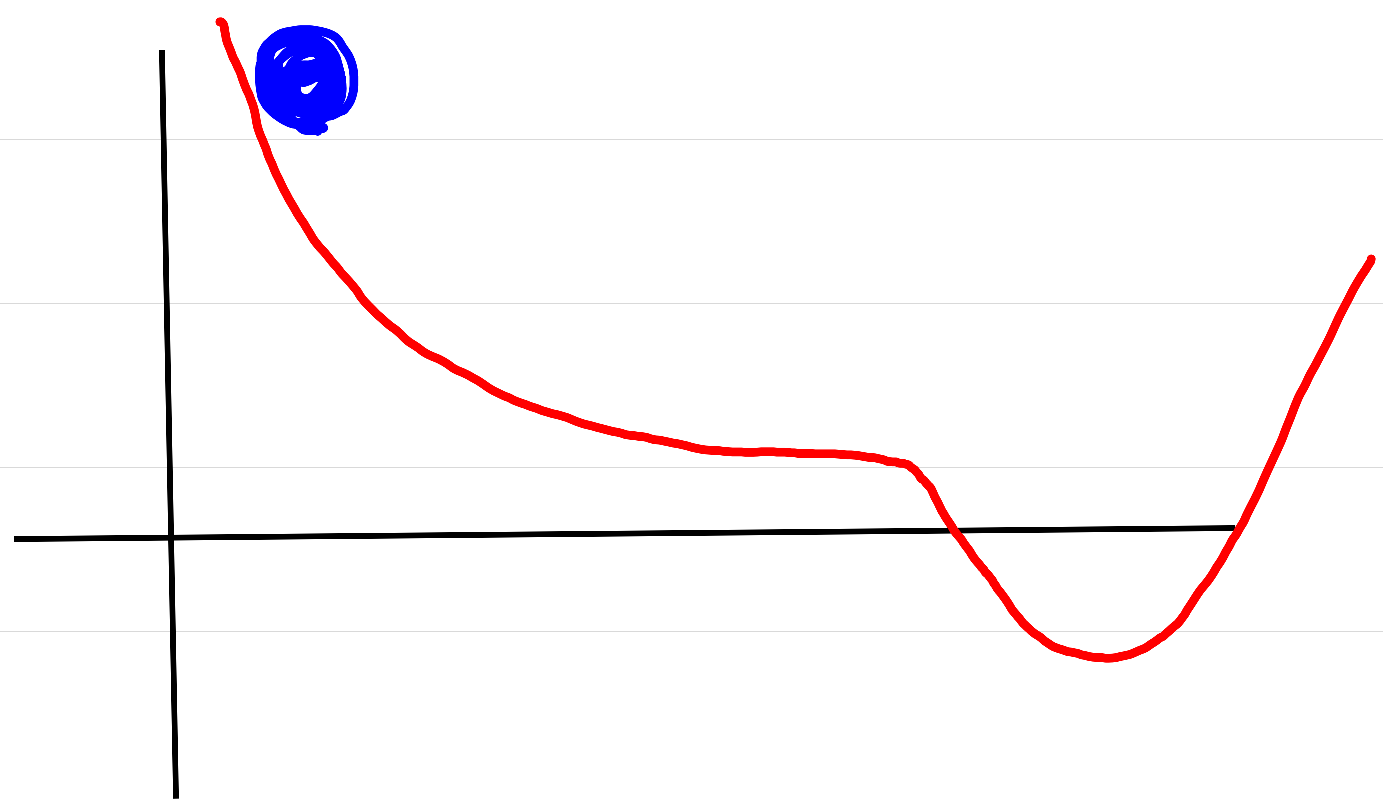
SGD with Momentum (Polyak, 1964)

Basic SGD: $\underline{g}_t = \nabla e_n(\underline{w}_t)$

$\underline{V}_t = -\underline{g}_t$

$\underline{w}_{t+1} = \underline{w}_t + \varepsilon_t \underline{V}_t$.

- New idea: add a momentum (a push) so that SGD would not stop if $\nabla e_n(\underline{w}_t) \approx 0$.



☐ when you let this heavy ball go, would it stop at the flat region? No

- Why? Inertia / Momentum

SGD + Momentum:

this approach is called "heavy ball" momentum.

$$\underline{g}_t = \nabla e_n(\underline{w}_t)$$

$$\underline{v}_t = -\varepsilon_t \underline{g}_t + \mu \underline{v}_{t-1}$$

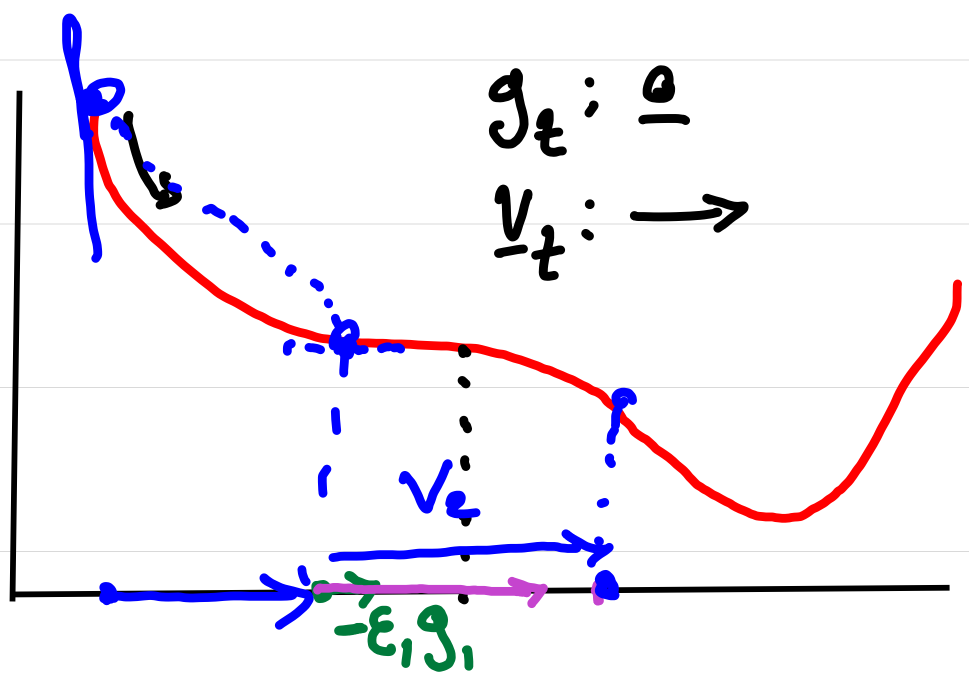
accumulation of your previous movements.

$$\underline{w}_{t+1} = \underline{w}_t + \underline{v}_t$$

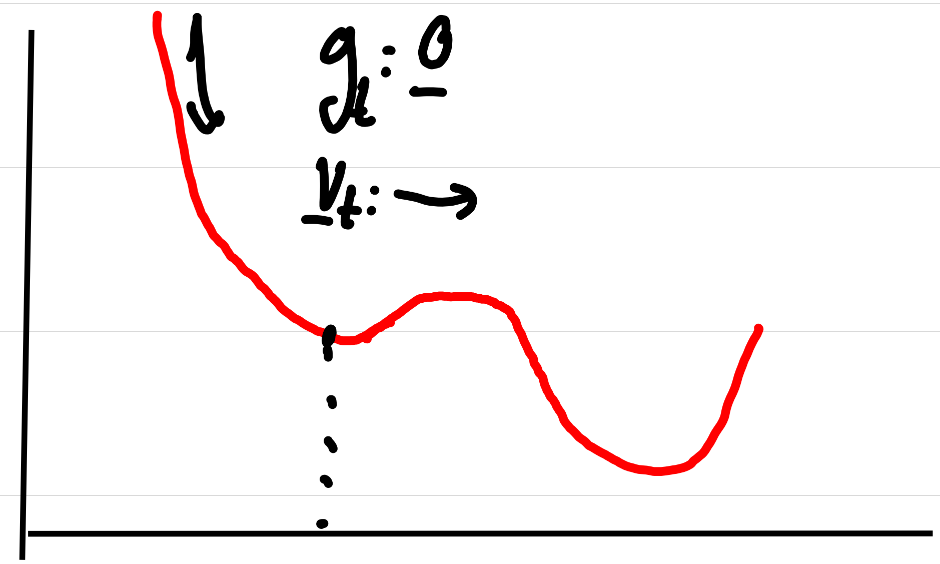
$$\boxed{\begin{array}{l} 0 < \mu < 1 \\ \text{e.g. } \mu = 0.9 \end{array}}$$

How Does Momentum Help?

① Momentum helps SGD escape flat regions, saddle points, and shallow local optimum.

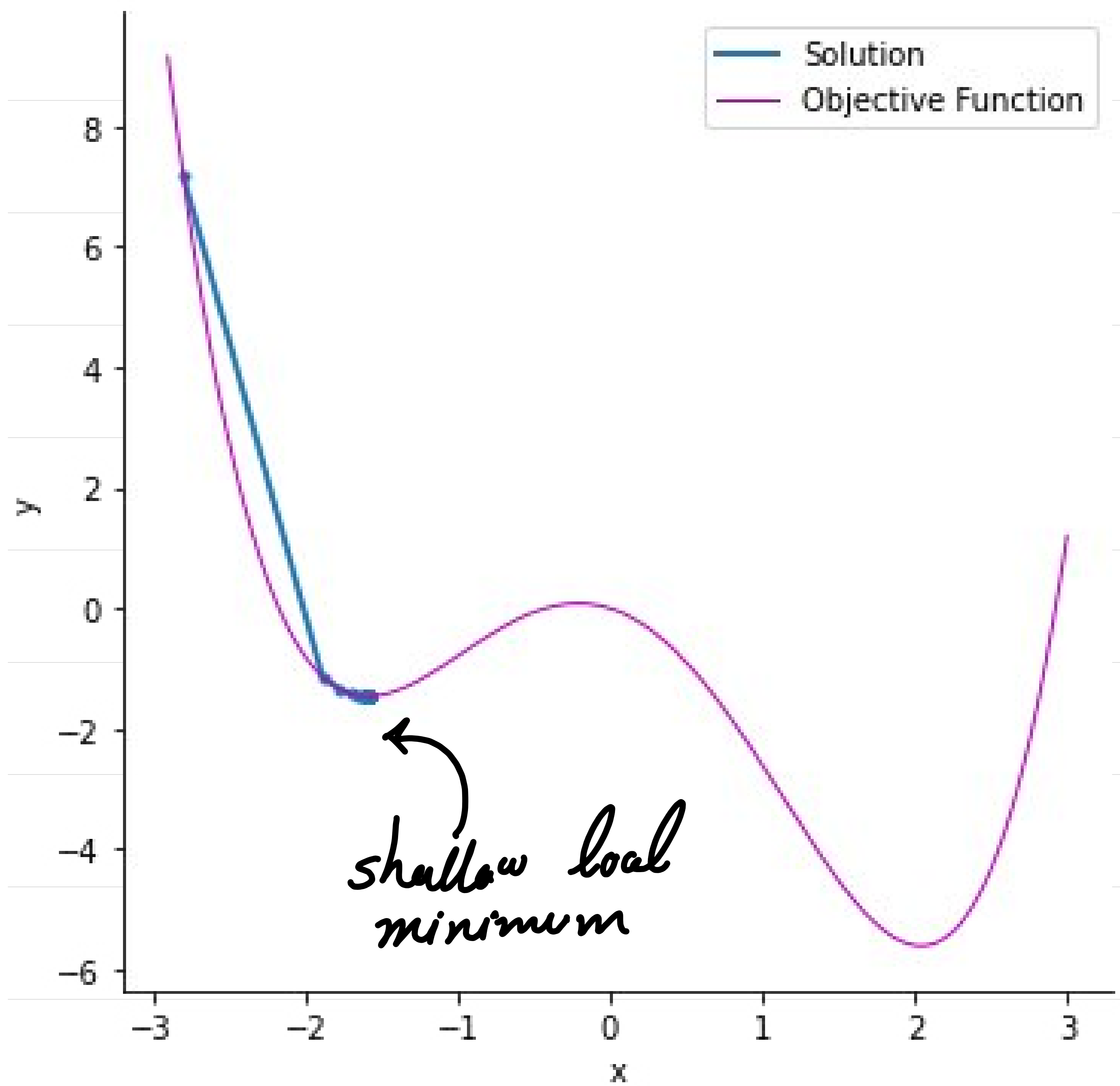


flat region/ saddle point

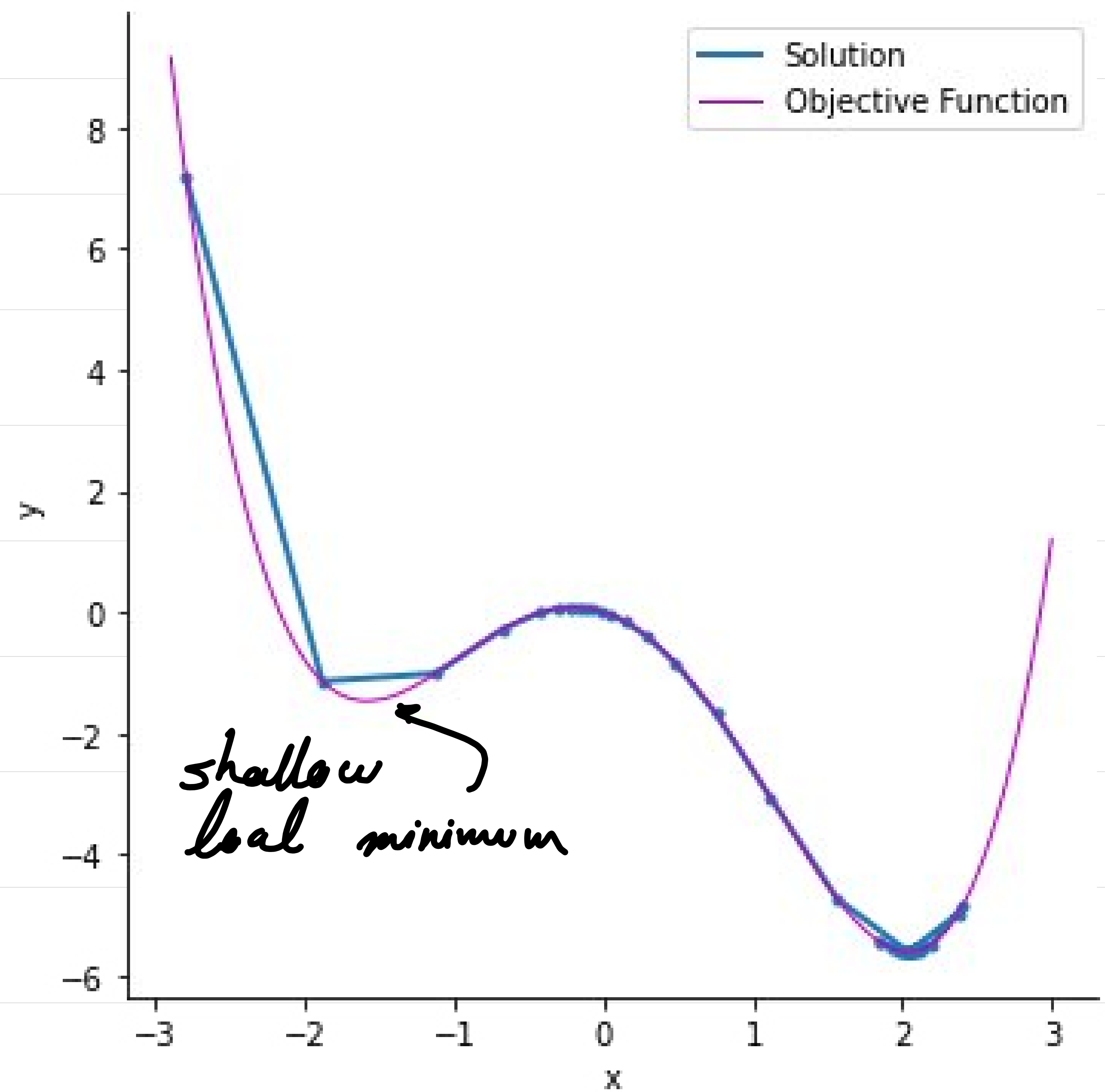


shallow local minimum

SGD



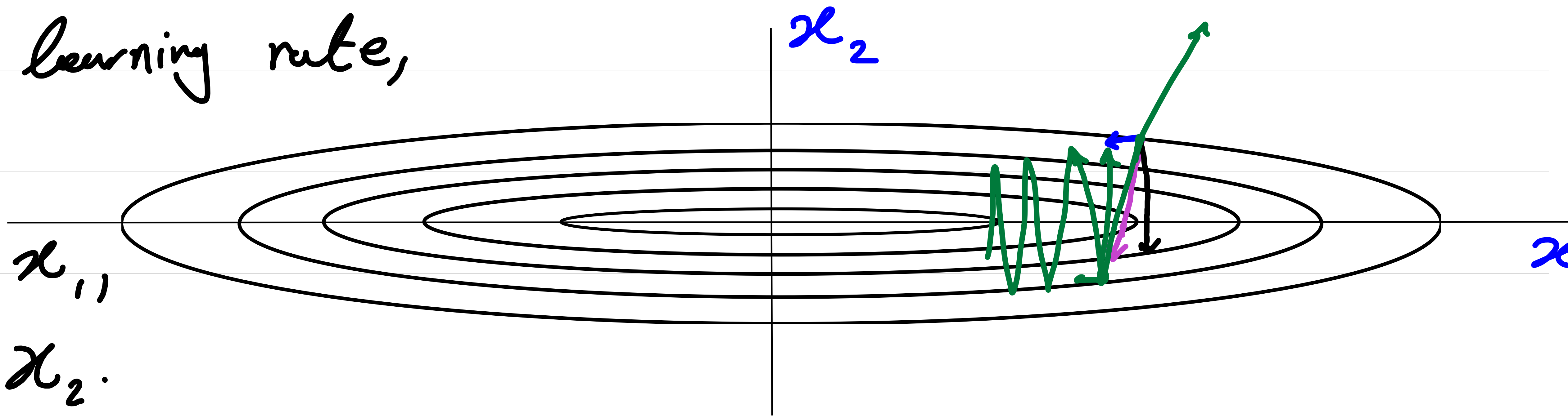
SGD + Momentum

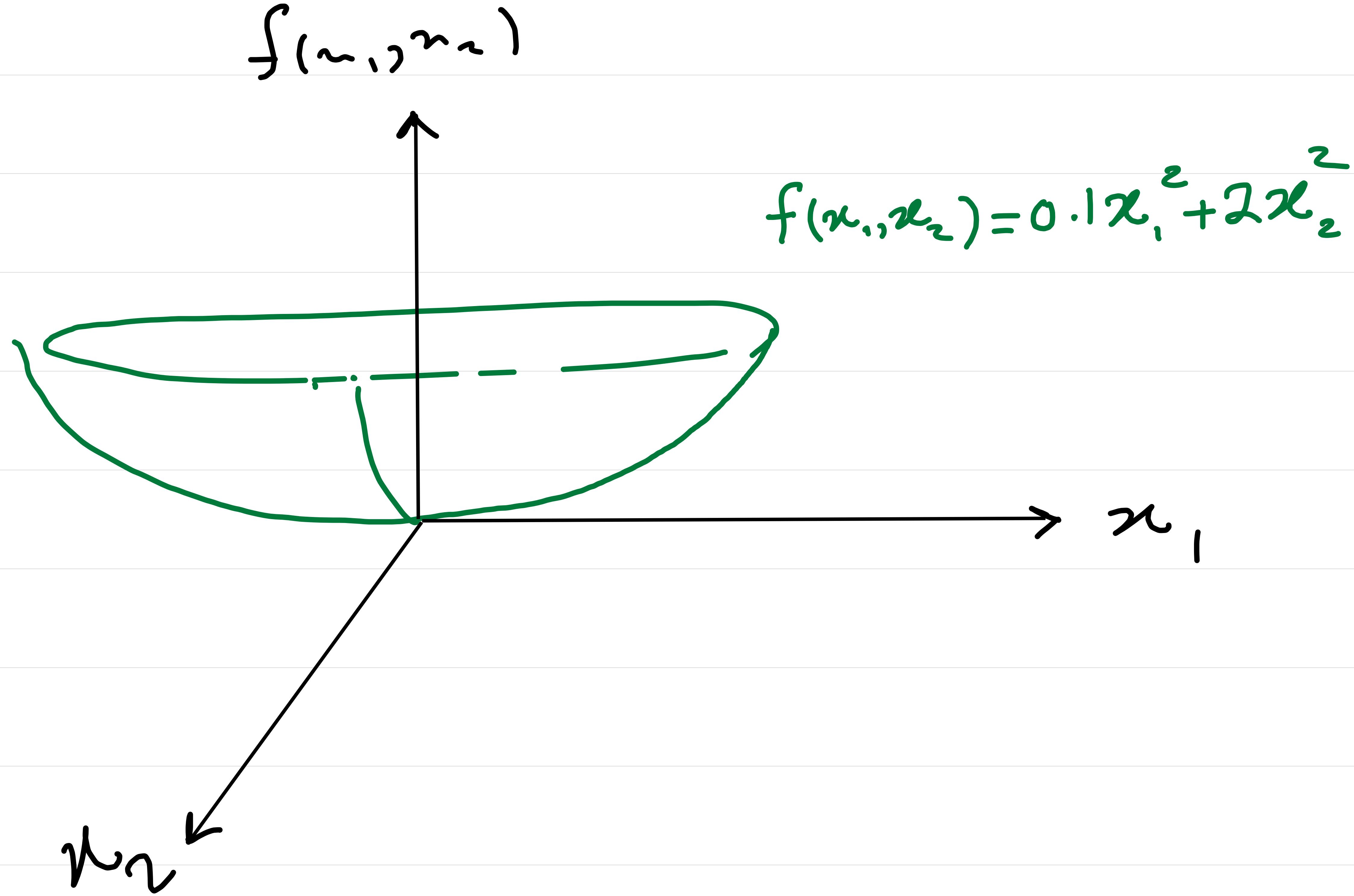


How Can Momentum Help?

2- Momentum can lead to faster convergence, even for convex functions.

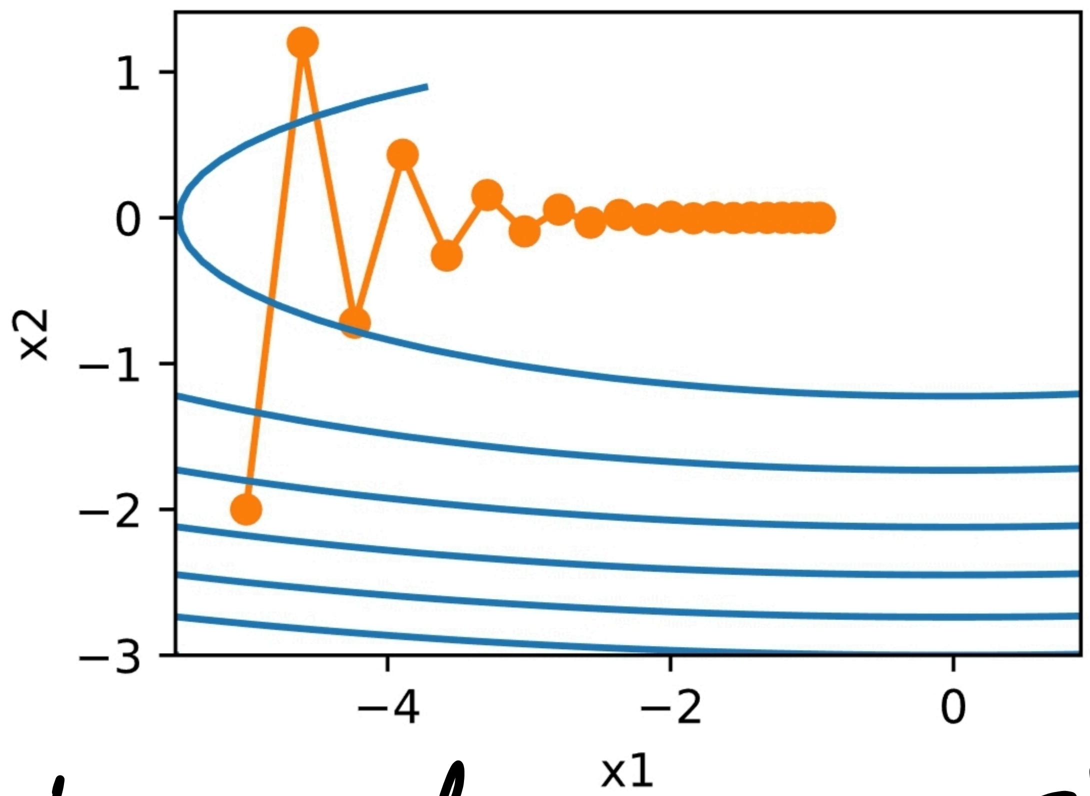
- Consider the stretched ellipsoid $f(x_1, x_2) = 0.1x_1^2 + 2x_2^2$.
  - ↳ it's minimum is at $(0,0)$
  - ↳ This function is very flat in the direction of x_1
  - ↳ The gradient in x_2 direction is much higher than x_1.
  - ↳ with a small learning rate, SGD won't diverge in x_2 direction but is very slow in x_1 direction.
  - ↳ with large learning rate, SGD progresses more rapidly in x_1, but diverges in x_2.



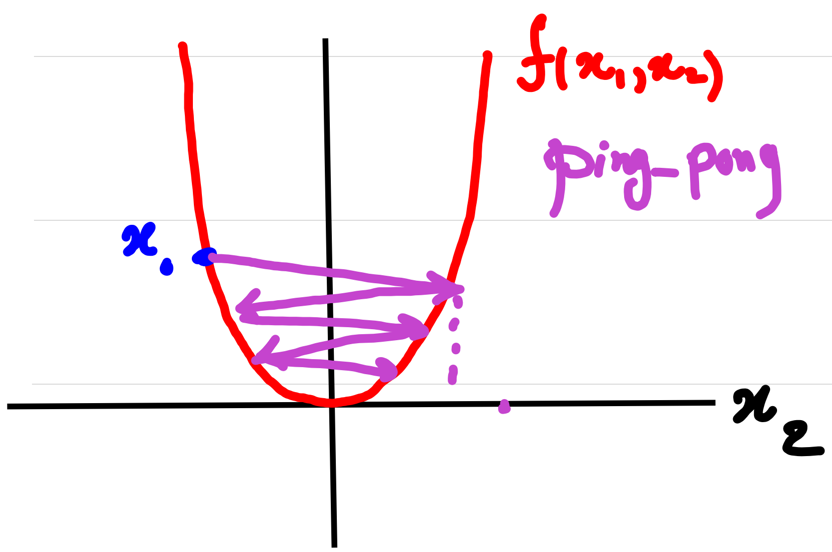


f(x1,x2)
f(x1,x2) = 0.1x1^2 + 2x2^2
x1
x2

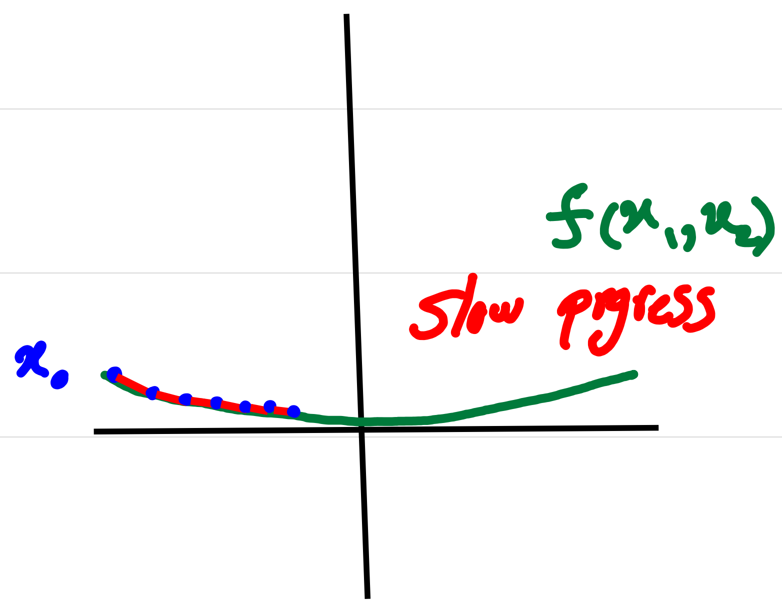
SGD with Learning rate = 0.4



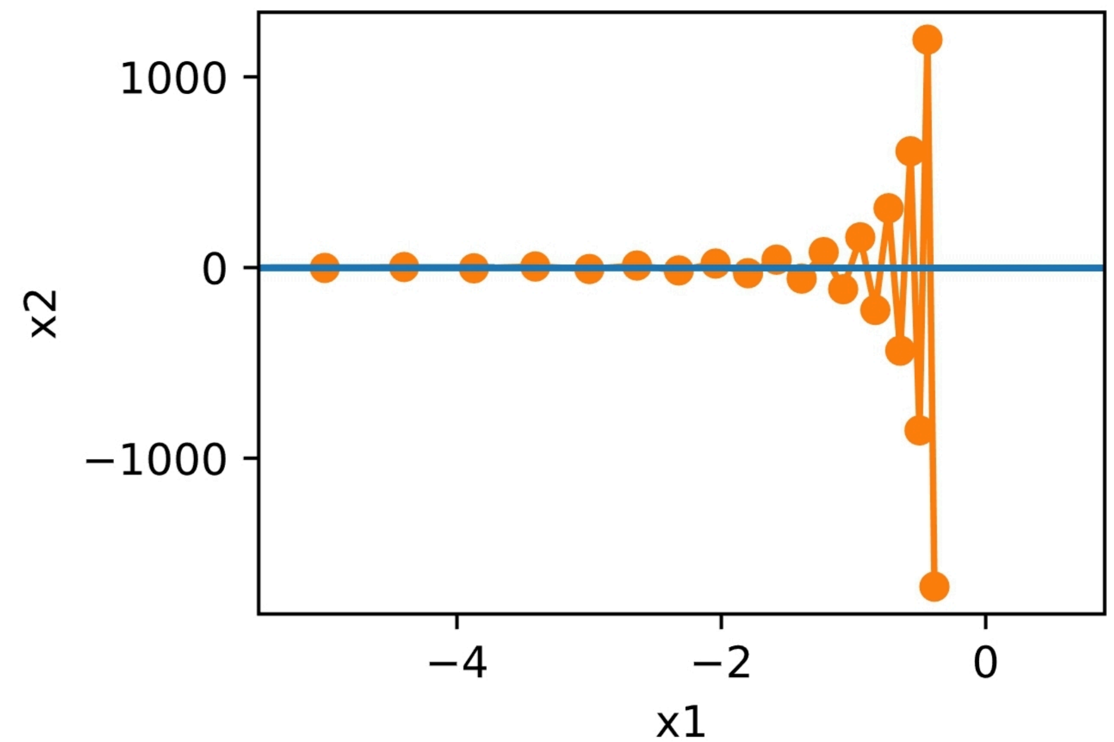
- slicing along x_2 direction



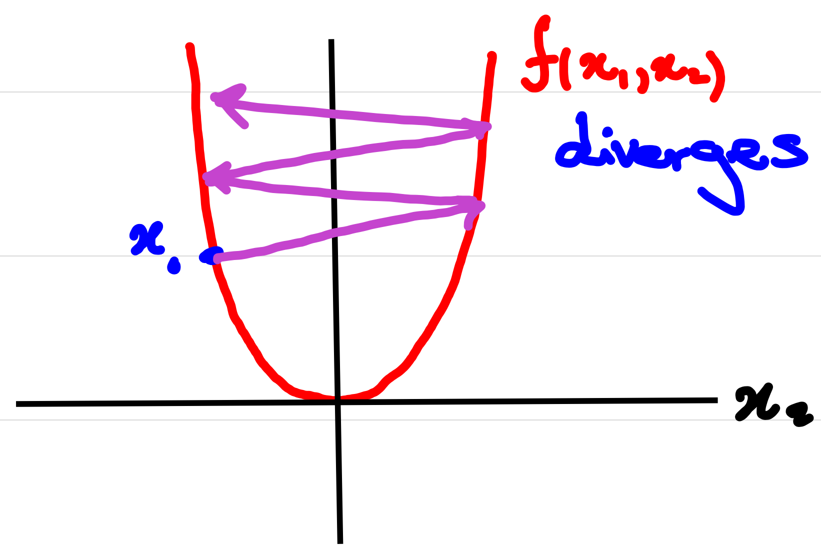
- slicing along x_1 direction



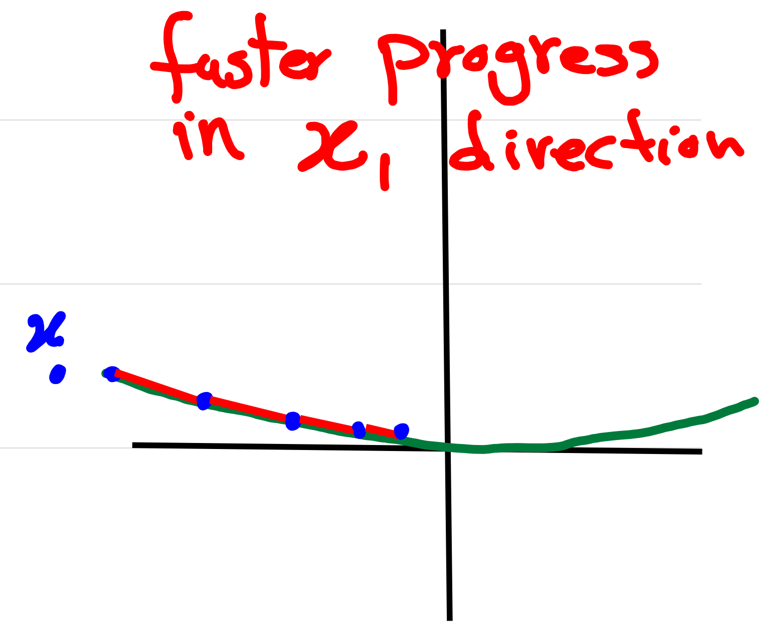
SGD with learning rate = 0.6



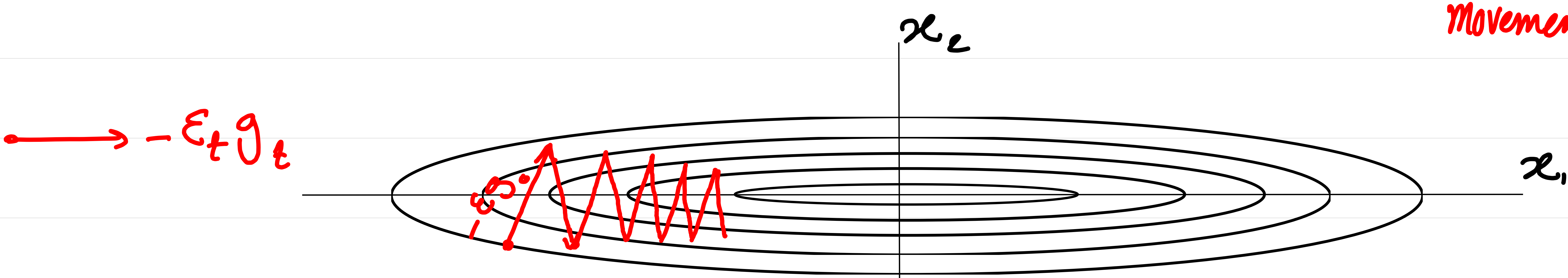
- slicing along x_2 direction



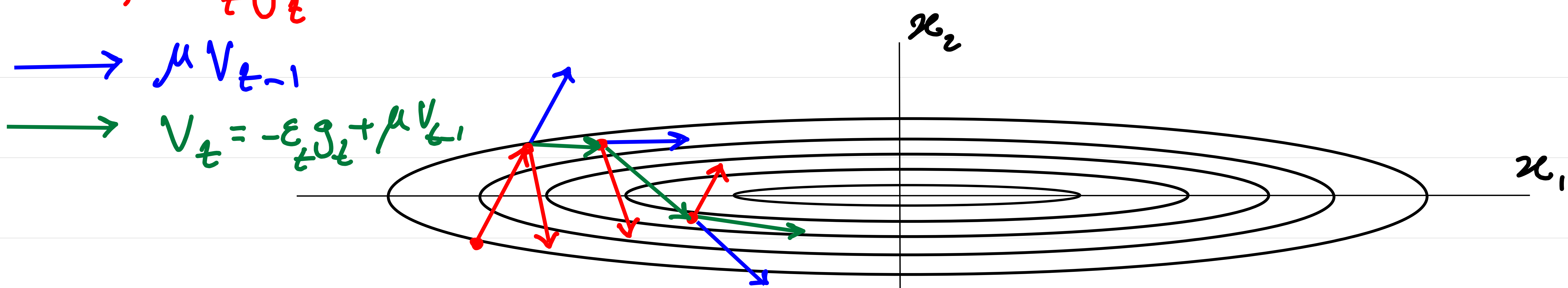
- slicing along x_1 direction



☐ SGD+Momentum : $\underline{V}_t = -\varepsilon_t \, \underline{g}_t + \mu \, \underline{V}_{t-1}$ ← accumulation of your previous movements.



SGD without Momentum



SGD with heavy ball momentum

- Heavy-ball momentum method works well in practice.
- But we don't have any theoretical proof for it.
- Nestrov, in 1983, modified the momentum and Could Prove nice theoretical guarantees.

Nestrov Momentum

- You update your location with your velocity first, and then take the gradient

$$\underline{V}_t = -\varepsilon_t \nabla e_n(\overbrace{\underline{W}_t + \mu \underline{V}_{t-1}}^{\text{intermediate point.}}) + \mu \underline{V}_{t-1}$$

$$\underline{W}_{t+1} = \underline{W}_t + \underline{V}_t$$

- Provably better convergence for convex functions.
  - Full GD: $|f(\underline{w}_t) - f(\underline{w}^*)| = O(\frac{1}{t})$
  - with Nestrov: $|f(\underline{w}_t) - f(\underline{w}^*)| = O(\frac{1}{t^2})$