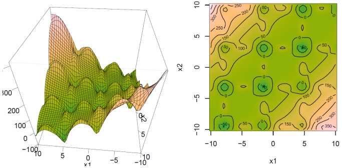
Optimization in Machine Learning

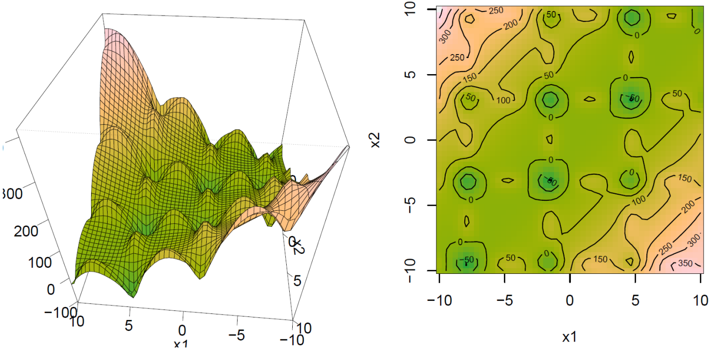
First order methods
GD – Multimodality and Saddle points



Learning goals

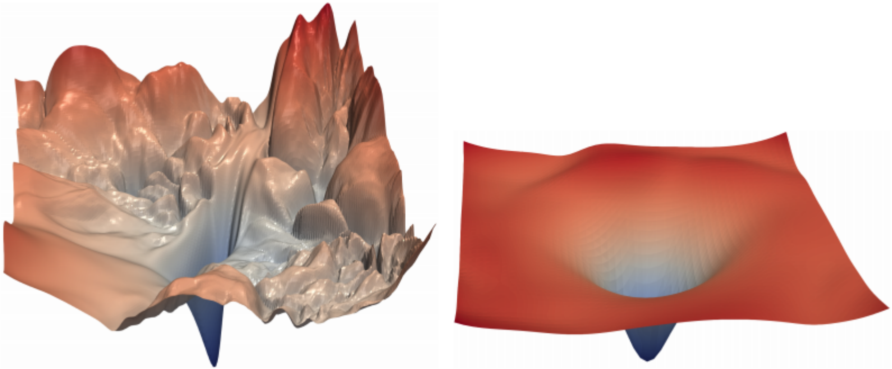
- Multimodality, GD result can be arbitrarily bad
- Saddle points, major problem in NN error landscapes, GD can get stuck or slow crawling

UNIMODAL VS. MULTIMODAL LOSS SURFACES



Snippet of a loss surface with many local optima

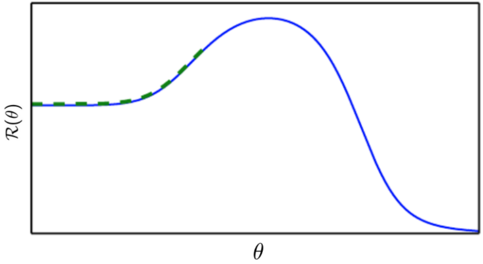
UNIMODAL VS. MULTIMODAL LOSS SURFACES / 2



In deep learning, we often find multimodal loss surfaces.
Left: Multimodal loss surface. **Right:** (Nearly) unimodal loss surface.
(Source: Hao Li et al., 2017.

GD: ONLY LOCALLY OPTIMAL MOVES

- GD makes only **locally** optimal moves
- It may move away from the global optimum

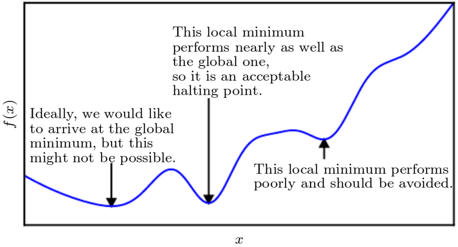


Source: Goodfellow et al., 2016

- Initialization on “wrong” side of the hill results in weak performance
- In higher dimensions, GD may move around the hill (potentially at the cost of longer trajectory and time to convergence)

LOCAL MINIMA

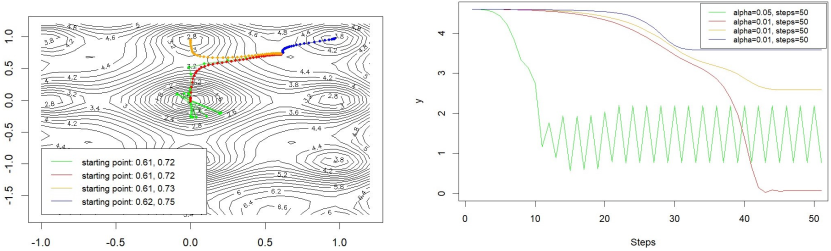
- **In practice:** Only local minima with high value compared to global minimium are problematic.



Source: Goodfellow et al., 2016

LOCAL MINIMA / 2

- Small differences in starting point or step size can lead to huge differences in the reached minimum or even to non-convergence



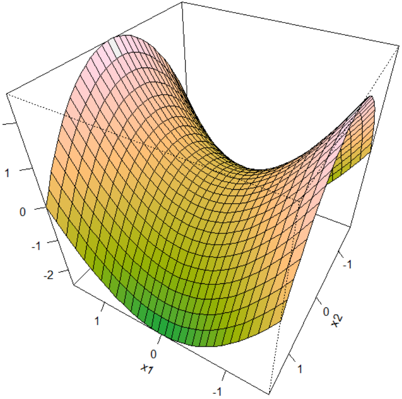
(Non-)Converging gradient descent for Ackley function

GD AT SADDLE POINTS

Example:

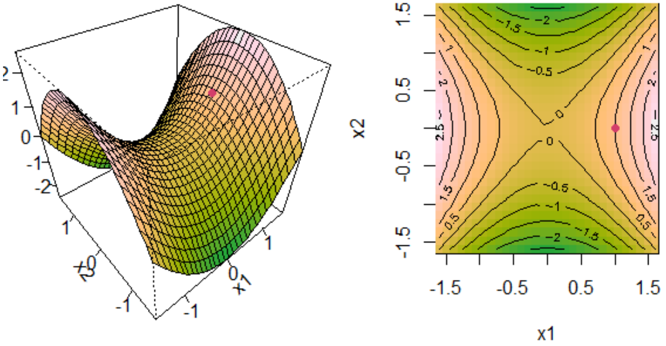
$$\begin{aligned} f(x_1, x_2) &= x_1^2 - x_2^2 \\ \nabla f(x_1, x_2) &= (2x_1, -2x_2)^\top \\ \boldsymbol{H} &= \begin{pmatrix} 2 & 0 \\ 0 & -2 \end{pmatrix} \end{aligned}$$

- Along x_1, curvature is positive ($\lambda_1 = 2 > 0$).
- Along x_2, curvature is negative ($\lambda_2 = -2 < 0$).



EXAMPLE: SADDLE POINT WITH GD

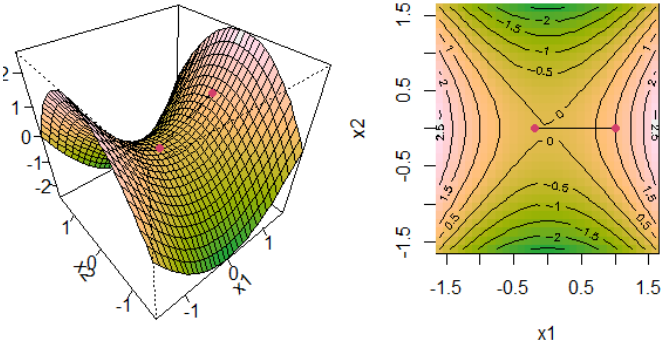
- How do saddle points impair optimization?
- Gradient-based algorithms **might** get stuck in saddle points



Red dot: Starting location

EXAMPLE: SADDLE POINT WITH GD

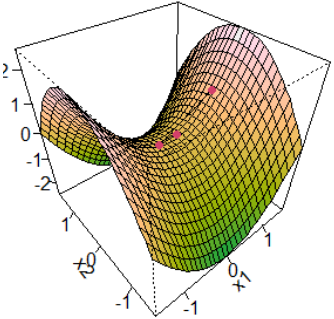
- How do saddle points impair optimization?
- Gradient-based algorithms **might** get stuck in saddle points

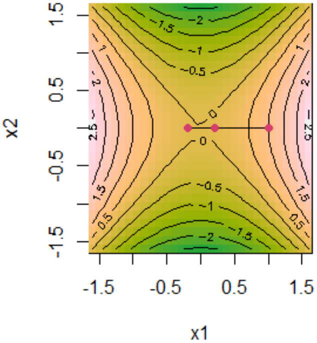


Step 1 ...

EXAMPLE: SADDLE POINT WITH GD

- How do saddle points impair optimization?
- Gradient-based algorithms **might** get stuck in saddle points

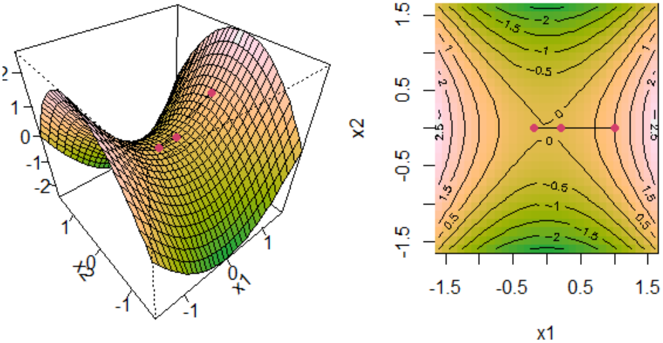




... Step 2 ...

EXAMPLE: SADDLE POINT WITH GD

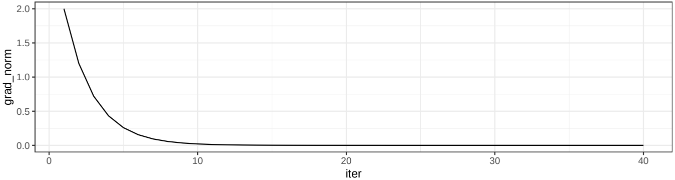
- How do saddle points impair optimization?
- Gradient-based algorithms **might** get stuck in saddle points



... Step 10 ... got stuck and cannot escape saddle point

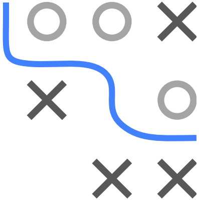
EXAMPLE: SADDLE POINT WITH GD

- How do saddle points impair optimization?
- Gradient-based algorithms **might** get stuck in saddle points



... Step 10 ... got stuck and cannot escape saddle point

SADDLE POINTS IN NEURAL NETWORKS



- For the empirical risk $\mathcal{R} : \mathbb{R}^d \rightarrow \mathbb{R}$ of a neural network, the expected ratio of the number of saddle points to local minima typically grows exponentially with d

- In other words: Networks with more parameters (deeper networks or larger layers) exhibit a lot more saddle points than local minima

- **Reason:** Hessian at local minimum has only positive eigenvalues. Hessian at saddle point has positive and negative eigenvalues.

SADDLE POINTS IN NEURAL NETWORKS / 2

- Imagine the sign of each eigenvalue is generated by coin flipping:
  - In a single dimension, it is easy to obtain a local minimum (e.g. “head” means positive eigenvalue).
  - In an *m*-dimensional space, it is exponentially unlikely that all *m* coin tosses will be head.
- A property of many random functions is that eigenvalues of the Hessian become more likely to be positive in regions of lower cost.
- For the coin flipping example, this means we are more likely to have heads *m* times if we are at a critical point with low cost.
- That means in particular that local minima are much more likely to have low cost than high cost and critical points with high cost are far more likely to be saddle points.
- “Saddle points are surrounded by high error plateaus that can dramatically slow down learning, and give the illusory impression of the existence of a local minimum” (Dauphin et al. (2014)).