11.1 Optimization in Deep Learning

Lecture based on "Dive into Deep Learning" **http://D2L.AI** (Zhang et al., 2020)

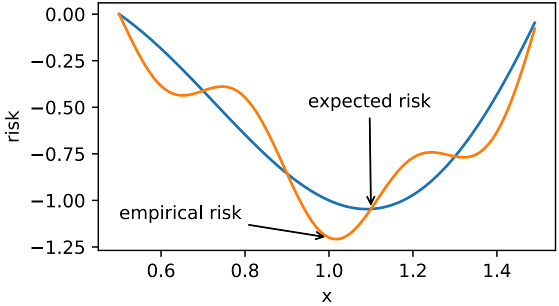
Prof. Dr. Christoph Lippert

Digital Health & Machine Learning

- On one hand, training a complex deep learning model can take hours, days, or even weeks.
- The performance of the optimization algorithm directly affects the model's training efficiency.
- On the other hand, understanding the principles of different optimization algorithms and the role of their parameters will enable us to tune the hyperparameters in a targeted manner to improve the performance of deep learning models.
- Almost all optimization problems arising in deep learning are *non-convex*.

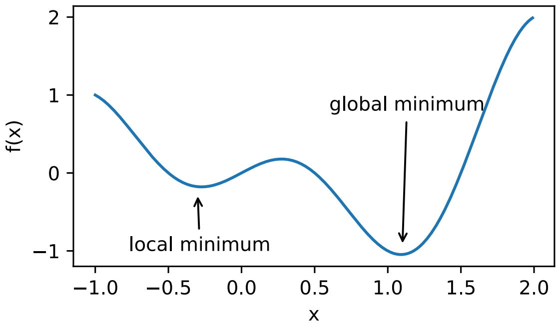
- In ML, we define a **loss function** first.
- We use an **optimization algorithm** in attempt to minimize the loss.
- The loss is the **objective function** of the **optimization problem**.
- By convention most optimization algorithms are concerned with **minimization**.
- To **maximize** an objective flip the sign.
- the goal of statistical inference (and thus of deep learning) is to reduce the **generalization error** or **expected error** (or **test error**).
- We need to pay attention to **overfitting** in addition to optimizing the **training error**.

Machine Learning is concerned with finding a suitable model, given a finite amount of data.

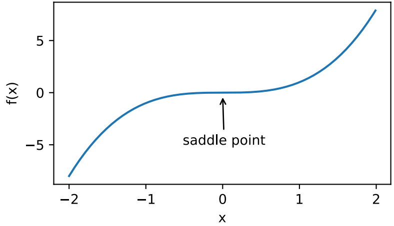


- For the objective function $f(x)$, if the value of $f(x)$ at x is smaller than the values of $f(x)$ at any other points in the vicinity of x, then $f(x)$ is a **local minimum**.
- If the value of $f(x)$ at x is the minimum of the objective function over the entire domain, then $f(x)$ is the **global minimum**.
- The objective in deep learning usually has many local minima
- numerical solution obtained may only minimize the objective function locally, rather than globally
- The gradient vanishes as we approach a minimum.

$$f(x) = x \cdot \text{cos}(\pi x) \text{ for } -1.0 \leq x \leq 2.0,$$

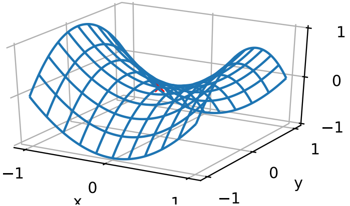


At a saddle point all gradients vanish but it is neither a global nor a local minimum.



$$(x) = x^3$$

Its first and second derivative vanish for $x = 0$.



$$f(x, y) = x^2 - y^2$$

It has its saddle point at $(0, 0)$.
This is a maximum with respect to y and a minimum with respect to x.

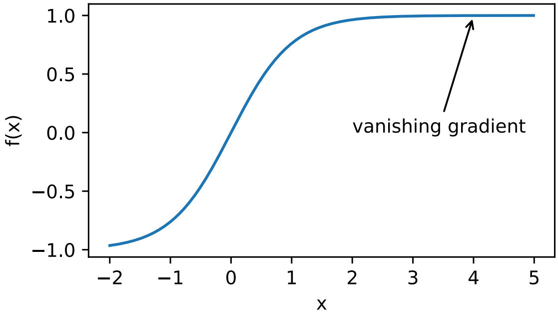
We assume that $f : \mathbb{R}^k \to \mathbb{R}$.

x could be a local minimum, a local maximum, or a saddle point at a position where the function gradient is zero:

- When the eigenvalues of the function's Hessian matrix at the zero-gradient position are **all positive**, we have a local **minimum** for the function.
- When the eigenvalues of the function's Hessian matrix at the zero-gradient position are **all negative**, we have a local **maximum** for the function.
- When the eigenvalues of the function's Hessian matrix at the zero-gradient position are **negative and positive**, we have a **saddle point** for the function.

Vanishing Gradients

- $f(x) = \tanh(x)$ and $x = 4$.
- $f'(x) = 1 - \tanh^2(x)$ and thus $f'(4) = 0.0013$.
- Optimization will make very little progress.
- This is one of the reasons that training deep learning models was hard prior to the introduction of the ReLu.



Summary

- Minimizing the training error does *not* guarantee that we find the best set of parameters to minimize the expected error.
- The optimization problems may have many local minima.
- The problem may have even more saddle points, as generally the problems are not convex.
- Vanishing gradients can cause optimization to stall.
  - Often a reparameterization of the problem helps.
  - Good initialization of the parameters can be beneficial, too.