Fundamentals of Neural Networks

Seminar Data Mining

Mathias Jackermeier
Fakultät für Informatik
Technische Universität München
Email: mathias.jackermeier@tum.de

***Abstract*— In this paper we introduce the reader to neural networks–a beautiful, biology-inspired machine learning paradigm.**

***Index Terms*— test**

I. Introduction

blabla

A. Stochastic Gradient Descent

II. Gradient Descent

A. Introduction

Gradient Descent is an algorithm used to iteratively minimize functions $f : \mathbb{R}^n \to \mathbb{R}$ of multiple values.

B. Directional derivatives

Since f is a function of multiple values, it does not suffice to.

From the definition of the directional derivative it follows that it evaluates to $\nabla f \cdot u$. A rigorous proof can be found in [1], but as an intuition, the change of $f(x)$ in direction u can be thought of as u_1 times the change in x_1 plus u_2 times the change in x_2 plus ... which results in $\sum_{i=0}^{n} \frac{\partial f}{\partial x_i} u_i = \nabla f \cdot u$.

Following Goodfellow et al. [2], we can find the direction in which f decreases fastest using the directional derivative:

$$\begin{aligned} &\min_u \nabla f \cdot u \\ = &\min_u \|u\|_2 \|\nabla f\|_2 \cos\theta \end{aligned}$$

...

Our goal is to choose a Δv that minimizes $\Delta C \approx \nabla C \cdot \Delta v$. The Cauchy–Schwarz inequality tells us that $|\nabla C \cdot \Delta v|$ is constrained by $\|v\|\|\nabla C\|$ where $|\nabla C \cdot \Delta v| = \|v\|\|\nabla C\|$ if and only if $\Delta v = \eta \nabla C$. Since $\nabla C \cdot \eta \nabla C = \eta \|\nabla C\|^2 > 0$ we can choose $\Delta v = -\eta \nabla C$ to minimize ΔC.

Gradient Descent can further be extended to include the momentum of the function [1].

Following [2], one can represent neural networks as a directed cyclical graph.

Hornik [3] has shown that.

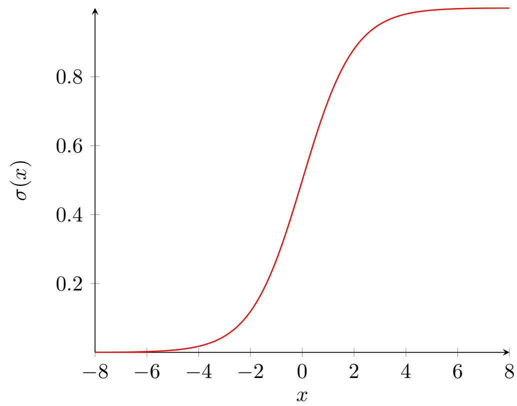
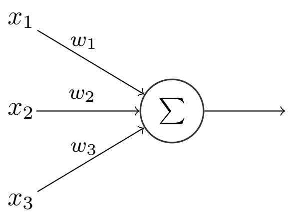


Fig. 1. The sigmoid function $\sigma(x)$



References

[1] S. Ruder, "An overview of gradient descent optimization algorithms," *CoRR*, vol. abs/1609.04747, 2016. [Online]. Available: http://arxiv.org/abs/1609.04747

[2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[3] K. Hornik, "Approximation capabilities of multilayer feedforward networks," *Neural Networks*, vol. 4, no. 2, pp. 251–257, 1991. [Online]. Available: https://doi.org/10.1016/0893-6080(91)90009-T