

Project description

Separation of sample complexity between fully connected and convolutional neural networks (LTHC)

Description and objectives

Sample complexity is the fundamental quantity in probably approximately correct (PAC-) learning. In the algorithm-dependent setting (i.e. one is given an algorithm A), this quantity describes, roughly speaking, the minimum number of i.i.d. samples that A must receive in order to guarantee (w.h.p.) that the output hypothesis has small population risk even over the worst-case ground-truth distribution.

The aim of this project is to make first steps towards improving upon recent results (<https://arxiv.org/abs/2010.08515>) which show a (multiplicative) separation of sample complexity between fully connected (FC-NNs) and convolutional networks (CNNs) on the order of $\Omega(d^2)$, where d is the dimension of the input space. In particular, establishing such results requires identifying learning tasks (distributions) that are efficiently learnable with CNNs and simultaneously difficult to learn with FC-NNs.

For proving strong upper bounds, this prior work crucially relies on the fact that only a very simple class of CNNs is considered. This makes it hard to assess the implications of these results for more realistic settings. Hence in this project we will first investigate empirically which separation can be observed in realistic scenarios and -- most importantly -- for which hard distributions this separation is attained.

One key obstacle is that the candidate learning tasks need be complex enough such that they are hard to learn for FC-NNs, while being simple enough to be amenable to theoretical analysis.

Secondly, these findings will then be used as a baseline for further mathematical investigations with the aim of establishing separation of sample complexity results.

Prerequisite:

Experiments: python + pytorch or similar. Theory: Learning theory, probability theory, linear algebra, some optimization.

Lab and supervisor:

LTHC, Thomas Weinberger

Number of students: one