**Background**

Neural networks are state of the art in numerous machine learning tasks ranging from image classification to beating world-class human players in the games of chess and go. These achievements have led to the implementation of neural networks in safety critical domains like aircraft controllers and autonomous driving. In these areas, failures can have dramatic consequences. Therefore, provable guarantees about the behaviour of the corresponding neural networks are necessary. Although the verification problem for ReLU-NNs is trivially decidable by enumerating all affine regions, it is unfortunately NP-complete [6].

**Idea**

In traditional program verification, large programs are broken down into submodules, which are easier to verify resulting in auxiliary statements. These statements can then be used to verify a property of the overall program.

The goal of this work is to explore this idea in the context of neural network verification. Challenges that arise in this context include specifying assumptions on partitions of the NN that 1. hold for the partitions (and are easy enough to be verified) and 2. are strong enough to prove the overall property (or at least assist in speeding up the verification). Manually obtaining specifications for submodules is already tedious for traditional software. For NNs it’s nearly impossible, as their sub-symbolic reasoning renders them (mostly) incomprehensible for humans.

It is therefore a promising idea to sample activation values for the NN at hand and use machine learning approaches to generate specification candidates. To this end, it may be worth-while to consider approaches for invariant learning for classical programs [2, 3, 4] as well as previous work on modular bounded model checking [7].

Further relevant literature: [4], [5] and [8]

---

**Supervisors**

Philipp Kern, philipp.kern@kit.edu
Room 203 (Geb. 50.34)

Samuel Teuber, teuber@kit.edu
Room 203 (Geb. 50.34)

**References**


