UNIVERSITEIT VAN AMSTERDAM

Generalization Strategies in Reinforcement Learning

M. Snel

A reinforcement-learning agent learns through trial and error by interacting with the environment and observing the effect of its actions and the reward that it receives after each action. Generally, the goal of the agent is to identify the sequence(s) of actions that lead to the maximal sum of rewards, or maximal *return*. While conceptually simple, the reinforcement learning framework is a remarkably powerful one for solving sequential decision tasks. It has had numerous successful practical applications, and remains a major area of research within the machine learning community.

Formally, the *task* the agent needs to solve is modeled as a *Markov decision process* (MDP). MDPs can be *partially observable*, meaning that the current true state of the environment is only partially known. For practical purposes, it is useful if the agent can generalize from tasks it has solved to new, but similar, tasks it might encounter in the future. This thesis investigates two classes of strategies for generalization in reinforcement learning.

The first strategy focuses on a multi-task reinforcement learning setting, in which tasks are sampled from a *domain*, a distribution over tasks. Agents start by solving one or more tasks, sampled from the domain. Since tasks in the domain are related, agents are then expected to retain knowledge about solved tasks and transfer it to new tasks, also sampled from the domain, in order to solve them more quickly. In this thesis, agents explicitly leverage structure that is shared between tasks, through the use of *shaping functions*. Shaping functions provide the agent with additional informative artificial reward, on top of the reward provided by the MDP. They can be either pre-designed or learned; this thesis focuses on learning them automatically. We do so by forming a dataset that consists of the union of state-action-value pairs of observed tasks. Akin to a supervised learning setting, two key choices need to be made: the target function the shaping function should approximate, and the representation for the shaping function, i.e., the feature set. We propose three different target functions to approximate, and evaluate each on a number of artificial domains. We show empirically that which target function is best depends highly on the domain, learning algorithm, and learning parameters.

Shaping function *representations* can also be pre-designed or learned, and this thesis is the first to learn them. In order to do so, we introduce FS-TEK (Feature Selection Through Extrapolation of k-relevance), a novel feature selection algorithm. It is based on the new notion of k-relevance, the expected relevance of a feature set on a sequence of k tasks sampled from the domain. We prove that k-relevance converges asymptotically to the domain relevance of the feature set. This property is used to derive FS-TEK. The key insight behind FS-TEK is that change in relevance observed on task sequences of increasing length can be extrapolated to more accurately predict domain relevance. We demonstrate empirically the benefit of FS-TEK on a number of artificial domains.

The second strategy for generalization in reinforcement learning investigates neural controllers that exhibit a degree of robustness to changes in task. That is to say, while these controllers do not learn on the new task(s), the objective is to minimize degradation in performance with respect to the task they were

trained on. We train five recurrent neural net (RNN) architectures and a deep and shallow feedforward net (FNN) on a set of simulated locomotion tasks, and subject them to two types of perturbations at test time: sensor noise, and a switch from flat to hilly terrain. While the FNNs learn fastest, no single architecture is best at everything at test time. However, we show that the FNNs and a continuous-time RNN (CTRNN) are most robust to task changes on average, with the CTRNN significantly outperforming the others under noise perturbation. In addition, we show that training on a hill task decreases the expected drop in performance due to perturbation for the RNNs, but not the FNNs.