Learning to Track Environment State

Learning to track environment state

Marian Andrecki and Nicholas K. Taylor

M.Andrecki@hw.ac.uk

Keywords: time series prediction, unsupervised learning, state estimation, neural networks

Introduction

Reinforcement learning (RL) based on artificial neural networks (ANN) has seen great successes in recent years. A notable recent breakthrough came from [2], where an artificial agent learnt to play games from raw pixels on a screen with scores produced by Atari game console simulation. The algorithm managed to surpass human performance on many classic 1980s games. However, RL has not yet achieved similar successes for real world tasks, such as controlling robotic manipulation.

RL requires dozens of hours of gameplay in order to perform at human level. This training time is currently a significant obstacle outside simulation systems. It has been argued that pure RL is data inefficient because the reward signal – the only feedback used – is sparse and contains little information.

One approach to extract more information from agent's experiences is to train to predict future observations. In this work, we investigate learning of stochastic forward models of environments from raw sensory observations. Such models could be then used for probabilistic state estimation, future prediction, planning, and ultimately, more efficient beheviour. The overarching goal is data-efficient reinforcement learning.

Method

We attempt to extend approaches presented in [1], where a recurrent convolutional neural network was trained to reliably predict frames from Atari2600 games (over periods of hundreds of time steps). Our method enables learning of forward models of a larger class of *stochastic* environments.

The architecture relies on a Predictive Autoencoder (PAE). In this setup, an observation o_t is passed through convolutional layers of an autoencoder. Then the resulting encoding is combined with information about previous observations to arrive at current estimate of the environment – belief state bs_t. Next, the predicted states ($bs_{t+1:t+T}$) are computed in this low dimensional space; this is achieved using Recurrent Neural Network (RNN). Finally, the predicted states are decoded into observation predictions $\hat{o}_{t+1:t+T}$. The loss used during training was Mean Squared Error (MSE).

The observations are generated by a simulator of bouncing balls (see Figure 1). During the training, the network effectively learns the forward model of the simulator. As the underlying dynamics are known, we construct a Particle Filter (PF) – a well established method for state estimation. This enables us to compare our architecture to a strong baseline.

Results

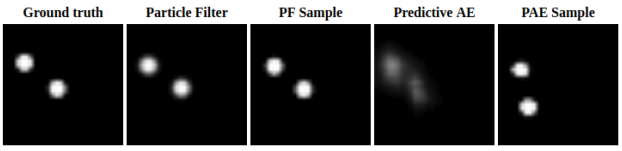

Figure 1: Observation estimates produced by PF and PAE after not observing the environment for 40 time steps. From the left: an observation with two balls (prediction target); expectation and sample observation from PF; finally, corresponding outputs from PAE. Image blur signifies uncertainty; it is not present in samples.

After training, our architecture is capable of high-quality generating predictions of future percepts. The performance is comparable to that of Particle Filter (PF). This is because PF starts with perfect knowledge of the underlying environment models. Whereas, PAE has to learn those from data. Figure 2 shows reconstruction error over time for the two methods – observations are available only intially, later methods have to rely on predictions.

To a large extent, the architecture can be treated as a *trainable* particle filter for unstructured data.

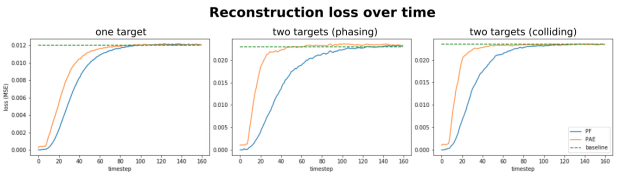

Figure 2: Plots show reconstruction loss for predictions in increasingly complex environments (left to right).

Conclusion

We evaluated a neural architecture for learning forward models of stochastic environments. Its performance was compared to a strong baseline on a tracking problem. The core finding is that predictive autoencoders are capable of propagating uncertainty in state. Further research will involve application to more complex environments, such as Atari games, and combination with Reinforcment Learning approaches.

References

[1] Silvia Chiappa, Daan Wierstra, and Shakir Mohamed. Recurrent Evironment Simulators. In *ICLR*, number 2015, pages 1–19, 2017.

[2] Volodymyr Mnih. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.