
Adversarial Exploration Policies for Robust Model Learning

Eric V. Jang
CSCI-2980HUG
Department of Computer Science
Brown University
Advisor: John F. Hughes

Abstract

Deep neural networks can be applied to model-based learning problems over continuous state-action spaces $S \times A$. By training a prediction network $\hat{f}_\tau : S \times A \rightarrow S$ on saved trajectory data, we can approximate the true transition function f of the underlying Markov decision processes. \hat{f}_τ can then be used within optimal control and planning algorithms to “predict the future”.

Robustness of \hat{f}_τ is crucial. If the robot (such as an autonomous vehicle) spends most of its exploration time in a small region of $S \times A$, then \hat{f}_τ may not be accurate in regions that the robot does not encounter often (such as collision trajectories). However, gathering enough training data to fully characterize f over $S \times A$ is very time-consuming, and tends to result in many redundant samples.

In this work, I propose exploring $S \times U$ using an “adversarial policy” $\pi_\rho : S \rightarrow A$ that guides the robot into states and actions that maximize model loss. Policy parameters ρ and model parameters τ are optimized in an alternating minimax game via stochastic gradient descent. Robot simulation experiments demonstrate that adversarial exploration policies improve model robustness with respect to the time the robot spends sampling the environment.

Keywords. Reinforcement Learning, Deep Learning, Robotics, Optimal Control

1 Implementation Notes

Full project writeup and source code can be found at <http://evjang.com/articles/adv-e2c>.

I implemented this in Python and C++, and used the following frameworks: ProtoBufs, TensorFlow, Numpy, Matplotlib, GLUT, BoxBot (a soft robotics simulator I wrote).