Techniques for Sample-Efficient Reinforcement Learning

by

William Fairclough Whitney

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Computer Science

New York University

September, 2021

Professor Kyunghyun Cho

To my dog Weierstraß, with affection. add dedication

Acknowledgements

- all my collaborators

- housemates

Abstract

CONTENTS

LIST OF FIGURES

List of Tables

1 | INTRODUCTION

Reinforcement learning (RL) provides a framework for systems which learn to make decisions under uncertainty about their environment. Such a system must simultaneously determine which of all possible environments it is interacting with and solve for an optimal strategy in that environment. This uncertainty is the core challenge of reinforcement learning, and one of the most fundamental questions of the field is how much experience a learning system requires to resolve its uncertainty and perform well. We call this the sample complexity of reinforcement learning.

Classically reinforcement learning was restricted to environments with small, countable state and action spaces.[1] In this setting a variety of algorithms were developed with robust guarantees on their performance relative to the ideal policy and on the amount of data required to approach perfect behavior. Bandit algorithms for environments without temporal dependence and dynamic programming for those with non-trivial dynamics both yielded practical success. However, the specter of exponentially-increasing sample complexity limited these approaches to low-dimensional settings.

The combination of deep learning with reinforcement learning in the last decade has given rise to the new subfield of "deep reinforcement learning", which leverages function approximation to scale reinforcement learning to tasks with large or uncountable state or action spaces. Deep reinforcement learning has led to dramatic results across a range of domains, from board games like Go to complex multiplayer computer games like StarCraft, from controlling automated balloons to manipulating Rubik's cubes, and even to abstract tasks like chip design.

A key unifying feature of these most impressive deep RL results is the availability of simulators. Each of these domains admits the construction of a simulation of sufficient fidelity that a policy may be trained in simulation and then deployed on the real task with little to no fine tuning. Furthermore, these simulations are inexpensive relative to collecting "real" data, reducing the cost of training a policy by orders of magnitude. By turning data collection into computation, simulation freed deep RL from paying attention to sample efficiency, as it was more important how computationally fast an algorithm was than how many hours or years of (virtual) experience it consumed.[2]

1.1   THE COST OF FREE DATA

Our reliance on simulation comes with hidden costs. Simulation-based RL has generated spectacular results, and simulation should be a primary tool in any effort to solve a real-world task. However, the availability of cheap simulation has shaped what the deep RL community studies by reinforcing work on simulatable domains and the large-data regime. This has resulted in slower progress on questions in the sample-limited regime. Sample-efficient reinforcement learning is important due to the fundamental scientific importance of understanding sample complexity and the intractibility of simulating every domain of interest.

SAMPLE COMPLEXITY IS FOUNDATIONAL.   The study of sample complexity in reinforcement learning addresses fundamental questions about what information is needed to reliably solve a problem. The observation that realistic, high-dimensional tasks are solvable with small sample sizes may be surprising, given that there are theoretical lower bounds requiring a number of samples which is linear in the number of states or exponential in the horizon [Du et al. 2020]. This disagreement requires explanation, and it suggests that real-world problems contain a significant amount of structure which makes them easier to solve. Understanding this structure in more

detail, as well as how it interacts with the inductive biases of the function approximators we use, could lead to significant breakthroughs.

SIMULATORS ARE EXPENSIVE. While many simulators already exist that are computationally inexpensive, many important systems of interest are computationally difficult to simulate with sufficient precision for transfer to the real world. Simulations of fluid dynamics, deformable materials, and large numbers of contacts (to name a few) can be significantly slower than real-time. Furthermore, the development of new realistic simulators is *financially* expensive not only because of the engineering of the simulator itself, but additionally due to the need to create and maintain a close correspondence between the simulation and the physical system of interest. These costs inhibit the use of simulators for domains which are too complex or too niche.

This thesis consists of work done in the last several years which makes deep RL somewhat more capable in the sample-limited regime. With this line of work I hope to unlock the wide range of environments which currently lack high-fidelity simulators. Beyond simply enabling RL in more complex environments, improvements to sample efficiency has the potential to allow agents to adapt on the fly to the specifics of the environment or objectives they interact with. A home robot might come to know the best way to pick up *your* cups, or an automated factory could produce more rapidly over the course of a production run as the networked assembly arms pool experience about manipulating each component part. In these settings learning more efficiently can be directly translated into consumer experience and dollars saved.

1.2   ELEMENTS OF SAMPLE-EFFICIENT LEARNING

Reinforcement learning can be thought of as comprising two distinct processes: data collection and policy optimization. Data collection, also known as *exploration*, is when the agent interacts with the environment in order to gain more information about the optimal policy. Policy opti-

mization is the process of using data collected from the environment to produce a policy that achieves as much reward as possible. There are three main variations on the RL setting which lead to interactions between these two steps: the *online* or *regret* setting, the *learn-and-deploy* setting, and the *offline* or *batch* setting. These settings are discussed in more detail in Section 2.2. In order that an RL algorithm overall be sample efficient, both exploration and policy optimization must themselves be efficient.

Efficient exploration means rapidly acquiring information about the optimal policy. This means collecting data that is diverse, such that valuable states and actions will be quickly uncovered, but also biasing data collection towards states and actions which are more likely to be optimal. If done well exploration spans the environment and then collects evidence to allow the agent to discard poor policies and differentiate between actions which are optimal and those which are merely good.

Efficient policy optimization means squeezing as much performance as possible out of the data which is currently available. While this is often discussed in the narrow frame of model-free RL, this process might include building models, learning representations, or even meta-learning. In principle one might hope to feed some prior beliefs along with whatever data has been collected from the environment and get back the best policy which is supported by that data. Recent work in off-policy RL and the offline RL setting have made progress towards this vision, but it remains some way off.

1.3 OVERVIEW

This thesis consists of work on improving the sample efficiency of reinforcement learning through three main directions: (1) collecting more diverse data without confounding policy learning; (2) learning representations which capture structure in the environment; and (3) studying policy optimization in the batch setting to develop policy improvement operators that are robust to limited

data. The motivating challenge through much of this work is robotic manipulation using low-dimensional positions or high-dimensional images as observations and with continuous-valued action spaces. However, the findings described here are applicable more widely across reinforcement learning.

The rest of the thesis is organized as follows:

- Chapter 2 introduces background on the problem of sample-efficient RL and how it relates to the several settings under which RL has been studied.

- Part I describes the role of exploration in sample-efficient RL, with Chapter 3 illustrating how exploration techniques adapted from deep RL to the sample-limited robotic setting can improve sample efficiency and performance.

- Part II discusses the role of representation in reinforcement learning, with connections to representation learning more generally in Chapter 4 and work on representations for sample-efficient RL in Chapter 5.

- Part III studies policy optimization in the offline setting, first describing the impact of over-paraterized models in offline bandit problems in Chapter 6, then studying the repeated application of policy improvement operators in the full offline RL problem in Chapter 7.

NOTES

1. Or more accurately, environments admitting assumptions that allow them to be treated as such. A 2-D continuous state space can be treated as a discrete grid of states with arbitrarily little waste under the assumption that the environment changes sufficiently slowly, for example.

2. Of course, not everyone in the deep RL community ignores sample efficiency. Notably many people working on robotics have maintained remarkable discipline in only running simulated benchmarks for physically plausible amounts of time, but there is also a small but vibrant community working on reaching human Atari performance with human-like amounts of experience.

2 | The Many Settings of Reinforcement Learning

2.1 Notation

2.2 Regret and Deployment

Reinforcement learning actually consists of several different settings.

- The *online* or *regret* setting. Here an RL agent learns by continually interacting with the environment with the goal of maximizing the total reward earned across all time. This gives rise to the explore-exploit tradeoff when acting: at each moment, the agent may choose to take an action which is uninformative but leads to greater short-term reward, or one which will yield more information at the cost of lower reward.

- The *learn-and-deploy* setting. This setting consists of distinct learning and deployment phases. In the learning phase, the agent is not required to perform well and may collect whatever data is most informative. In the deployment phase, the policy is fixed and should be as close to optimal as possible. This setting schematically breaks down into pure data collection followed by pure policy optimization, though in practice policy optimization is an essential tool during exploration.

7

- The *offline* or *batch* setting. This consists only of pure policy optimization given a fixed dataset of environment interactions collected by an extrinsic behavior policy. After learning from this data in whatever way it sees fit, an algorithm produces a fixed policy with the objective of earning as much reward as possible. Though described as a reinforcement learning setting, it does not include any actual reinforcement as the agent never learns from its own interactions with the environment. However, this makes the offline RL setting uniquely valuable for isolating how much can be learned from particular data.

2.3   Simulated and Physical Environments

2.4   Discounting

Part I

Exploration and Sample Efficiency

3 | Decoupled Exploration and Exploitation Policies

{sec:deep}

Part II

Representations and Auxiliary Tasks

4 | Evaluating Learned Representations

{sec:representation-e

5 | Dynamics-Aware Embeddings

{sec:dyne}

Part III

Improving Performance with Batched

Data

6 | Offline Contextual Bandits with Overparameterized Models

{sec:offline-bandits}

7 | OFFLINE RL WITHOUT OFF-POLICY EVALUATION

{sec:offline-rl}

8 | Conclusion

{sec:conclusion}

Bibliography

Du, S., Kakade, S., Wang, R., and Yang, L. F. (2020). Is a good representation sufficient for sample
  efficient reinforcement learning? *ArXiv*, abs/1910.03016.