ECE239AS Project Milestone Report

Dueling Network Architectures for Deep Reinforcement Learning

Sarat Bhargava
805219546
sarat2895@ucla.edu

Eden Haney
005221845
ehaney@ucla.edu

Radhika Nayar
805226085
radhica@ucla.edu

Abstract

Most approaches for improving Deep RL's performance focus on the learning algorithm itself while using standard Neural Networks not necessarily designed for reinforcement learning. Here we propose to study and implement a potential improvement to the standard neural networks used in Deep Q-learning applications, known as the dueling network architecture. We expect this network to outperform classic implementations of other Deep Q-learning algorithms in Atari game environments.

1 Introduction

1.1 Background

Deep Reinforcement Learning (Deep RL) has been making waves in the machine learning community due to many recent successes. However, most implementations of Deep RL use standard neural networks, such as convolutional networks, multilayer perceptrons (MLPs), long short-term memory networks (LSTMs) and autoencoders. The main focus of recent advances has been on designing improved RL algorithms, or simply on incorporating existing neural network architectures into RL methods, instead of improving the learning algorithm itself. The Dueling Network Architecture for Deep Q-learning (*dueling architecture*) leverages a neural network architecture that is customized for model-free RL (1) to improve performance of standard deep Q-learning algorithms.

For this project, we aim to understand, survey and implement the dueling architecture, a neural network architecture for model-free reinforcement learning (1). This architecture, visualized in figure 1, advances a new network by improving already established algorithms such as the robust Double Deep Q-learning (Double DQN) algorithm (2) and Deep Q-Networks (DQN) (3). Thus, a true understanding of the dueling architecture requires a solid understanding of the algorithms on which it is built.

1.2 Architecture

The dueling architecture is a single Q-network architecture where the lower layers are convolutional, as in DQN. However, instead of following the convolutional layers with a single sequence of fully connected layers, we use two sequences (or streams) of fully connected layers. The constructed streams provide separate estimates: one for estimating the state-value function and the other for estimating the state-dependent action advantage function. The two streams are then combined to produce a single output Q-function. Similar to DQN, the dueling architecture outputs a set of Q-values, one for each action.

UCLA ECE 239AS Reinforcement Learning Spring 2020.

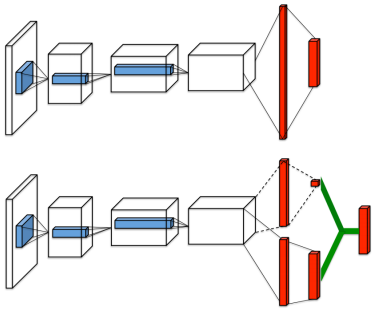

Figure 1: A popular single stream Q-network (top) and the dueling Q-network (bottom). The dueling network has two streams to separately estimate state-values and the advantages for each action; the green output module combines them. Both networks output Q-values for each action. From (1).

A key motivator behind this architecture is that for some games, it is unnecessary to know the value of each action at every time step. By explicitly separating two estimators, the dueling architecture can learn which states are valuable, without having to learn the effect of each action for each state.

1.3 Implementation

We adapted the already working OpenAI implementation of DQN to represent the dueling network and Double DQN.

We implemented the algorithms in an Arcade Learning Environment, which is composed of Atari games. The challenge was to learn to play Space Invaders, one of the Atari games, given only raw pixel observations and game rewards.

2 Preliminary Results

Here we list the preliminary results of the dueling network and Double DQN in the Space Invaders environment. As we have recently gotten the code working, we expect to be able to improve results moving forward with longer training sessions.

The first metric we examine is the score each algorithm was able to obtain. Table 1 shows the statistics of each algorithm run for five trials. These trials were done after training for 100k iterations. As we can see, the dueling architecture mean has increased from the baseline (using random weights) but has a high variance. Double DQN however has decreased and has a lower variance than both the random weights and Dueling DQN. However, because these statistics were only taken using five trials we cannot be certain of any clear patterns.

Figures 2-5 visualise the other metrics we examined. Figures 2 and 4 show the Bellman loss of Dueling and Double DQN over the 100k iterations. Here We see the Bellman loss of the dueling architecture clearly decreasing over time while Double DQN's loss increases rapidly. We are currently unsure why Double DQN is performing poorly but are working on it.

Figures 3 and 5 show the number of frames that the agent "lived" for in the game. For the dueling architecture we see that while the average number of frames doesn't increase over time, the variance does seem to decrease.

Moving forward we hope to get double DQN working properly as we would like to compare the dueling architecture to it but if we don't, it is not the focus of this project. We will also run the agents for a higher number of iterations and sample a higher number of scores to have more trust in the data.

Table 1: Score Statistics

Architecture	Mean	Variance
No Training	174.7	82.0
Double DQN	135.0	46.5
Dueling DQN	216.0	126.5

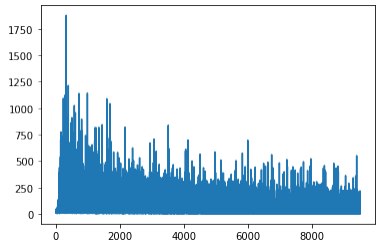

Figure 2: The Bellman loss of the dueling architecture over 100k iterations.

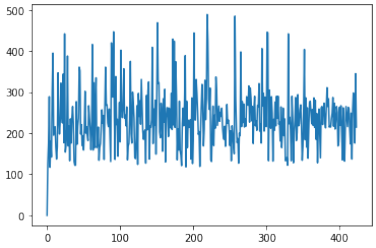

Figure 3: The number of frames alive of the dueling architecture over the number of episodes run.

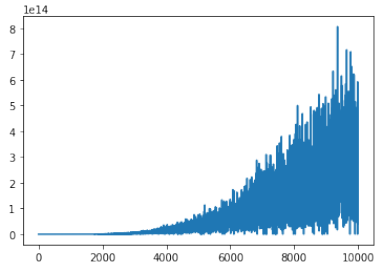

Figure 4: The Bellman loss of Double DQN over 100k iterations.

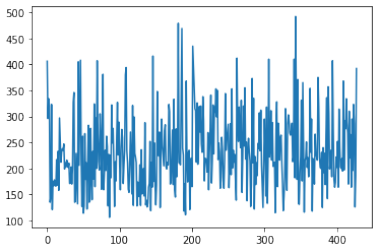

Figure 5: The number of frames alive of Double DQN over the number of episodes run.

References

[1] Z. Wang, T. Schaul, M. Hessel, H. van Hasselt, M. Lanctot, and N. de Freitas, "Dueling network architectures for deep reinforcement learning," *arXiv:1511.06581 [cs]*, Apr 2016. arXiv: 1511.06581.

[2] H. van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," *arXiv:1509.06461 [cs]*, Dec 2015. arXiv: 1509.06461.

[3] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, and et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, p. 529–533, Feb 2015.