CS7011 : Topics in Reinforcement Learning
Summary of Imagination-Augmented Agents : Weber, Racaniere, Reichert

Sahana Ramnath : EE15B109

Abstract

This paper (Weber et al., 2017) proposes Imagination-Augmented Agents (*I2As*), a novel architecture for deep reinforcement learning combining model free and model based aspects. Rather than using the environment's model directly to get the policy, I2As learn to interpret the model's predictions, and use them as additional context for their deep policy networks. I2As show improved data efficiency, performance, and robustness to model misspecification.Experiments were conducted and reported on various domains including Sokoban and MiniPacman.

1. Introduction

Standard model-based methods usually suffer from model imperfections due to function approximation and model assumptions. These problems are unavoidable in complex domains and hence model free methods have always dominated model based methods.
I2A aims to address these by learning to interpret these imperfect predictions. It doesn't rely solely on the predicted reward; it uses the additional information a trajectory can contain, beyond the reward sequence(such as an informative subsequence that might help to solve a subproblem, which did not result in a higher reward).They can be trained on low-level observations with little domain knowledge, similar to model-free agents. I2A learns in an end-to-end way to extract useful knowledge gathered from model simulations without relying exclusively on simulated returns.It achieves better performance with less data, even with imperfect models.

2. I2A Architecture

2.1. The Imagination Core and Rollout Module

In order to 'imagine' trajectories, I2A uses environment models : models that, given information from the present,can be queried to make predictions about the future. These simulate imagined trajectories, which are interpreted by a neural network and provided as additional context to a policy network.It rolls out the environment model over multiple time steps into the future, by initializing the imagined trajectory with the present time real observation, and subsequently feeding simulated observations into the model.
The actions chosen in each rollout result from a rollout policy $\hat{\pi}$ (explained in the next section). The environment model and $\hat{\pi}$ constitute the *imagination core module*, which predicts the next time step. The imagination core produces n trajectories ($\hat{T_1}$,..., $\hat{T_n}$) each of which is a sequence of features (f^{t+1},....,$f^{t+\tau}$), where t is the current time, τ the length of the rollout, and f^{t+i} the output of the environment model (predicted observation/reward). The rollout encoder ε processes the imagined rollout as a whole and returns a rollout embedding $e_i = \varepsilon(\hat{T_i})$. An aggregator A combines the different rollout embeddings as $c_{ia} = A(e_1, ..e_n)$.

2.2. The policy module

This is the final part of I2A : a network that takes the information c_{ia} from model based predictions, as well as the output c_{mf} of a model free path (a network which only takes the real observation as input), and outputs the imagination-augmented policy vector π and estimated value V.

3. Experiments and Setup

3.1. Rollout Strategy

One rollout is performed for each possible action in the environment.All subsequent actions are produced by the shared rollout policy $\hat{\pi}$. $\hat{\pi}$ is obtained by distilling the imagination-augmented policy π(policy distillation into a smaller model free network used for this).

3.2. I2A components, environment models

The rollout encoder ε is an LSTM with a convolutional encoder which sequentially processes each trajectory T. The features $\hat{f}_t$ are fed in reverse order. The aggregator concatenates all the summaries. The model free path is a standard CNN.
The environment model defines a distribution which is optimized by using a negative log likelihood loss l_{model}. This can be pretrained or jointly trained with the I2A architecture by adding l_{model} as an auxillary loss.

3.3. Agent training and baseline agents

Using a fixed pretrained environment model, I2A architecture is trained with A3C. The main baseline agent is the standard model free architecture CNN.

3.4. Results

Experiments were conducted on Sokoban and MiniPacman. Results were much better as compared to standard MCTS baelines.

References

Weber, Théophane, Racanière, Sébastien, Reichert, David P, Buesing, Lars, Guez, Arthur, Rezende, Danilo Jimenez, Badia, Adria Puigdomenech, Vinyals, Oriol, Heess, Nicolas, Li, Yujia, et al. Imagination-augmented agents for deep reinforcement learning. *arXiv preprint arXiv:1707.06203*, 2017.