Efficient Credit Assignment

Treating the Action Distribution as an Action improves Actor-Critic

—

Anirudh Goyal
Alex Lamb

Markov Decision Process

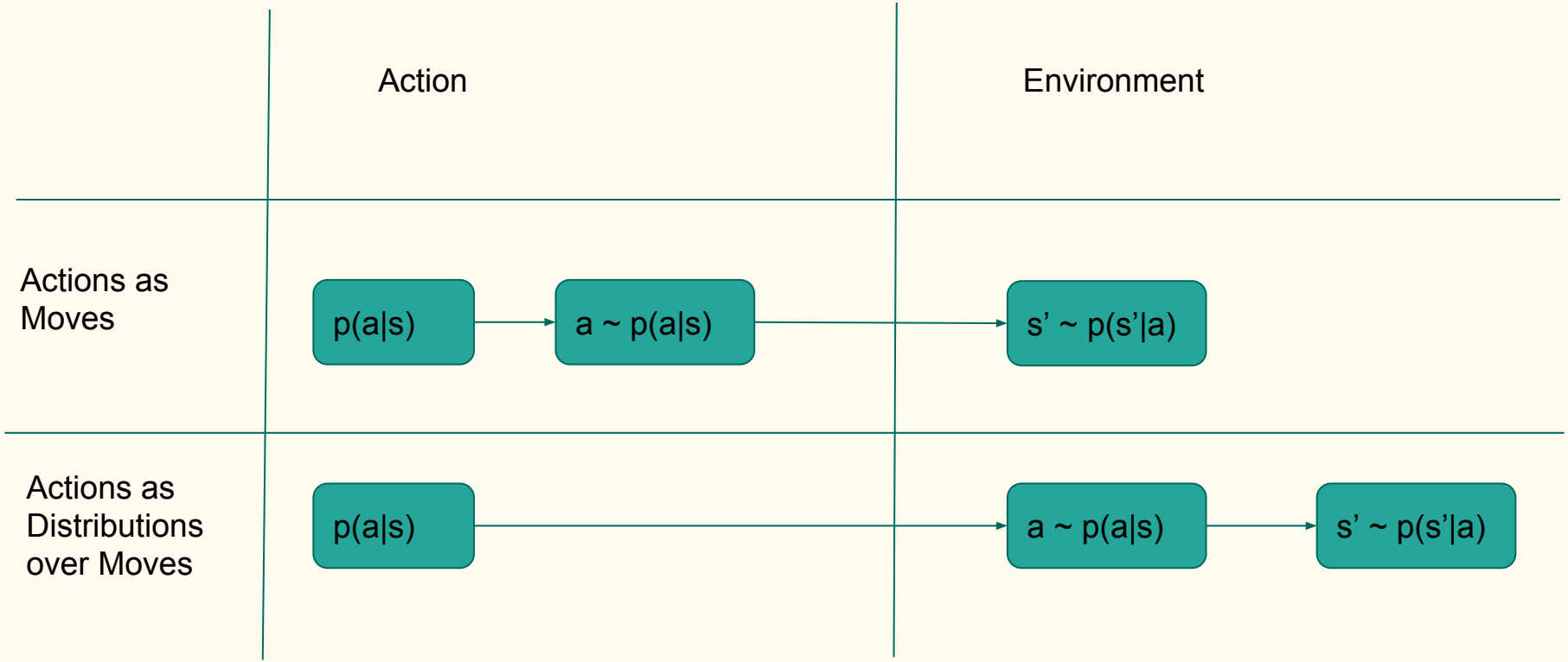
-Suppose that we have an agent that can take one of many different potential moves on each step.

-A standard way to model this is to treat the selected move as the action.

-However, we could also have our agent output a probability distribution over moves as its action on each time step.

-In this alternative formulation, sampling the move that the agent takes is treated as part of the environment.

Illustration of Proposed Modification



Where could this help us?

-When we have lots of rarely sampled actions that lead to bad states.

-With moves-as-actions, if any action is sampled and the gradient tries to push it down, all other probabilities will be pushed up by the action of the softmax.

-So we may need to wait a long time for these rarely sampled values to get all of their values pushed down, since pushing down one pushes up the others.

-However if our action is a probability distribution over moves, the critic can learn that different rare actions are bad and push them down even if they aren't sampled.

Where could this potentially hurt us?

-Suppose we have two moves A/B. If our model's action is p(A)=0.51, p(B)=0.49, then our model will get a very weak signal to push towards the better state.

-A given probability vector will lead to a wide variety of different moves.

-So it may be hard to make progress when probability values are similar.

-May need to compensate by injecting another noise source in the actor network.

Experiment

-Evaluate actor-critic with moves as actions (standard) against probability distributions as actions (ours).

-Compare on a simple game with four actions. Conceptually we would expect the biggest benefit to come when we have a large action space with lots of similar low probability actions.

-Atari might be an interesting use case for this - however for the homework we wanted something where we could get results in $<$30m.

Task

-We trained an actor-critic network on the frozen lake task.

-Some states are “holes” which lead to end of episode. Reward is reaching goal.

-Used OpenAI gym.

Model

-Actor-Critic with 1-step TD error.

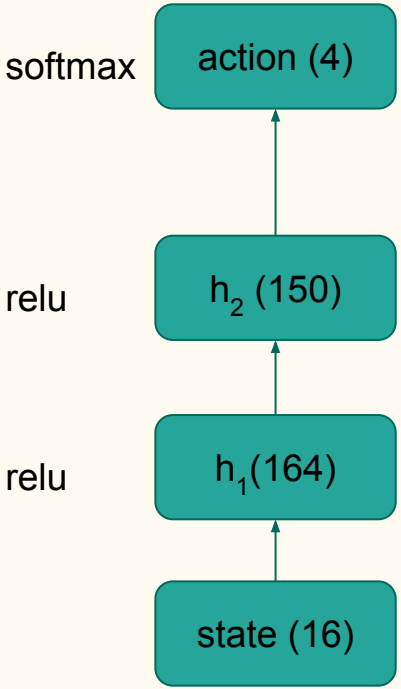
-Initially scan over all states to push the critic to have a uniform value over states.

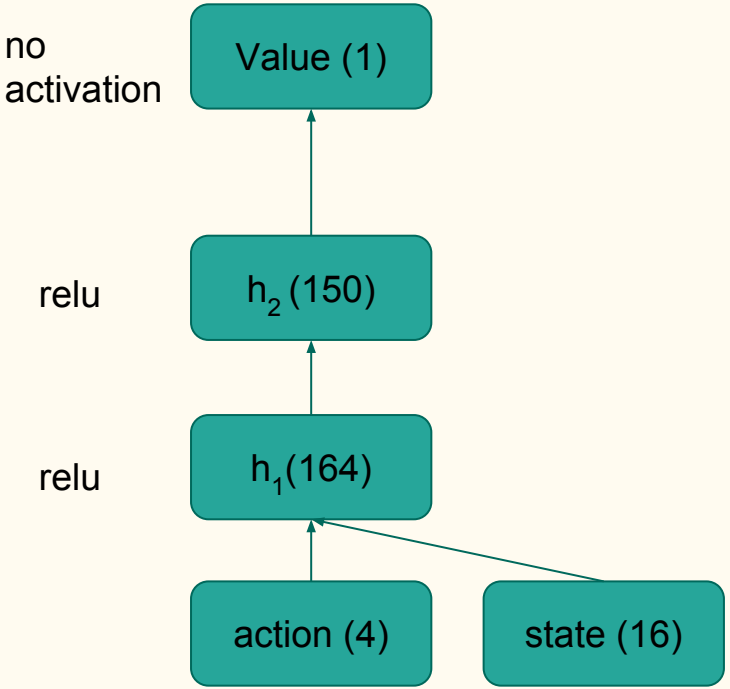
-Optimize using nesterov's accelerated gradient descent.

-Use experience replay with a buffer size of 80. Note: why don't you oversample winning games in your experience replay buffer?

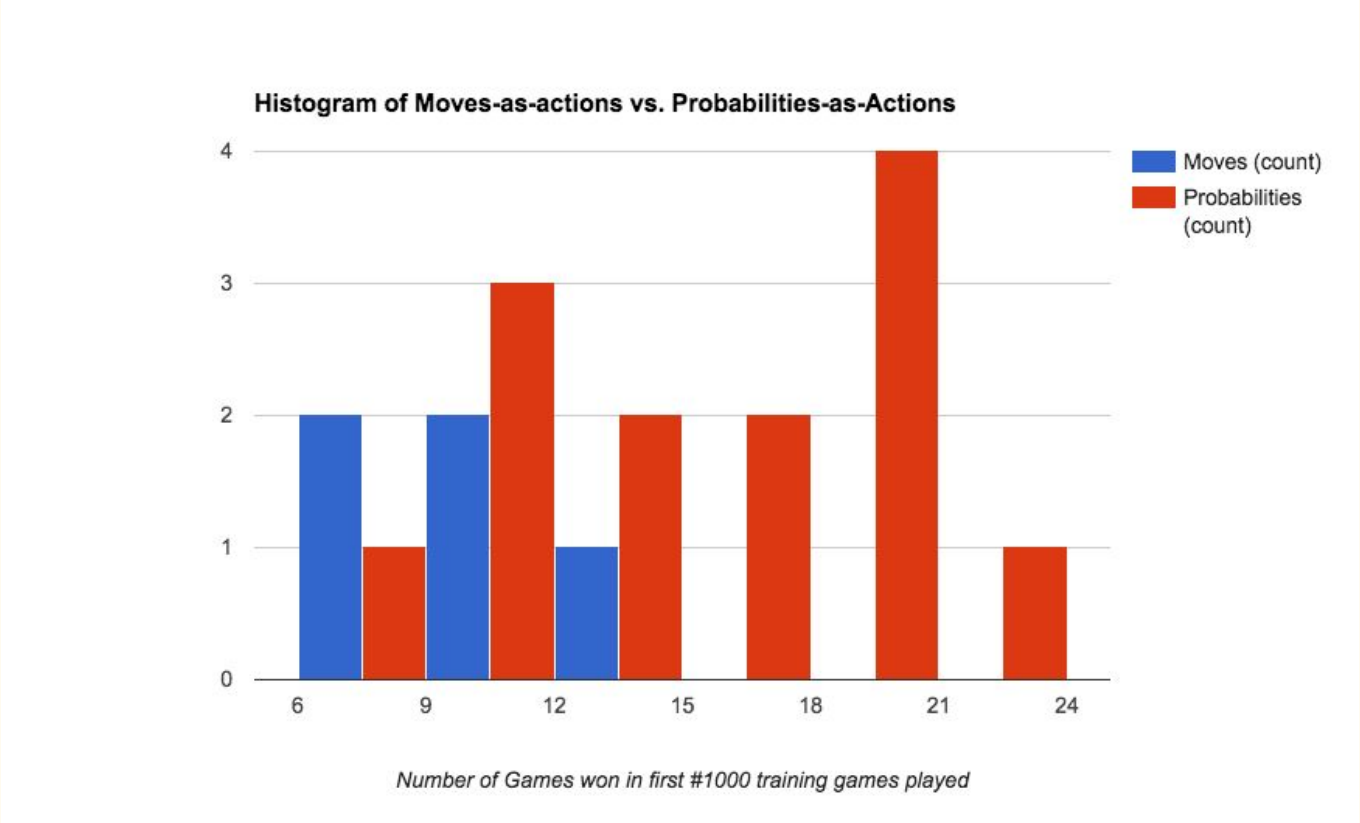
-Discount factor of 97.5%

Model: Actor and Critic Networks





Results - Winning Rate



Results - TD Error

Conclusions

-We've introduced a general way of reformulating MDPs so that the action corresponds to a distribution over moves rather than the selected move.

-We've discussed conceptual arguments for why this ought to help and hurt in different scenarios.

-We've presented preliminary experimental evidence for a single case where giving probabilities as actions outperforms giving selected moves.