This is the title of the conference paper

Andrea Stocco[*]
Psychology Department
University of Washington
Seattle, WA 98195
stocco@uw.edu

Patrick J. Rice
Psychology Department
University of Washington
Seattle, WA 98195
pjrice@uw.edu

Abstract

The *title* should be a maximum of 100 characters, and something really compelling/descriptive.

The *abstract* should be a maximum of 2000 characters of text, including spaces (no figure is allowed). You will be asked to copy this into a text-only box; and it will appear as such in the conference booklet. Use 11 point type, with a vertical spacing of 12 points. The word **Abstract** must be centered, bold, and in point size 12. Two line spaces precede the abstract. I hope the .sty takes care of the vertical spacing, etc., because it is not obvious how to implement that here.

Keywords: lets not have swear words here huh

Acknowledgements

We are deeply indebted to Person1 and Person2 and all the other persons that contributed to distracting me while I tried to work...

[*] This is here as an example in case I need to use it in the future. Use footnote for providing further information about author (webpage, alternative address)—*not* for acknowledging funding agencies.

1 Model-free RL

It is commonly known that model-free reinforcement learning (RL) is a powerful approach for obtaining an optimal long-term action policy in the absence of transition probability and reinforcement functions. In other words, a model-free RL agent must interact with an unknown environment (i.e. sample the environment repeatedly through action) in order to construct an optimal control policy, based on the pattern of reward received by interaction with the environment. This policy is oriented such that the agents actions consider both immediate and future reward, optimized to maximize some value over time. The key idea that enables an agent to determine an optimal policy within an unknown environment is that of temporal-difference (TD) learning (Sutton and Barto, 1988). Under TD learning, an agent estimates the values of states on-line (during exploration of the environment) through the updating of a state-dependent value function. This value function estimates the value of the current state, and is updated once the agent performs an action in order to transition to a different state, which results in the receipt of some reward. Upon reward receipt, the value function is updated by the weighted sum of the estimated value of the reward of all future steps (beginning from the current state, modulated by some learning rate alpha. These estimated state values can then be directly used to implement a policy (e.g. epsilon-greedy) to select actions on the basis of those that lead to states that are estimated to return the greatest value over time.

The ideas behind TD methods have since been expanded, including a proposal by Watson (1989) that defined a TD control algorithm now known as Q-learning. Q-learning is an off-policy method that allows the agent to choose to take non-optimal actions while still estimating an optimal value function. In comparison to the TD method described above, Q-learning carries two essential differences: 1. the method considers the value of state-action (SA) pairs, rather than the value of states alone (however, the two formalisms are equivalent (Sutton and Barto, 1998)); and 2. for each time-step, the learned action-value function is updated (in part) by the estimated value of the best action available, rather than the actual action taken. By updating based on the best action available while allowing the agent to make inferior choices, this procedure increases the rate of learning in the face of a suboptimal action selection process.

Both TD learning and Q-learning have been shown to converge to the optimal value function with probability 1 (CITE THIS). As such, methods of this class provide BLAH

However, in some circumstances, these model-free RL methods produce bizarre (suboptimal? Unintended? Incomplete?) results. As defined, these methods emphasize learning of rewarding actions updating the value of a state/SA pair increases the likelihood that the agent will choose the action that leads to that state/SA pair when it is next given the opportunity to do so. As a result, even though it converges to an optimal value function, an agent still does not have complete knowledge of its environment namely, it does not know much (if anything) about suboptimal states/SA pairs.[DISCUSS EXPLORATION PROBLEMS HERE]. This is not an issue while the agent has full access to its actions, but what if learned good (i.e. optimal) states/SA pairs are blocked from the agent? In this case, the agent cannot take the actions it usually would by following its policy and value function, and as a consequence, cannot act optimally within the new environment. In essence, the agent has no knowledge of how to navigate bad options how to choose the least bad, when forced to.

2 Probabilistic Stimulus Selection task

A situation in which this circumstance arises is when modeling a classical psychology task paradigm, the Probabilistic Stimulus Selection (PSS) task. The PSS task is a repetitive, two-alternative forced-choice task made up of two consecutive phases a training phase in which a participant repeatedly makes choices between fixed pairs of stimuli, and a test phase where the participant is presented with new combinations of options. Across both phases, there are six possible stimuli, implemented as symbols that are difficult to describe (in order to make memorization of each stimulus history of success more difficult). Each stimulus carries an intrinsic probability of success, ranging linearly from 20 percent to 80 percent. During the training phase, the stimuli are presented a fixed pairs, for a total of three sets: [[20 80],[30 70],[40 60]], and participants receive feedback regarding the outcome of their decision directly after making a selection. Participants are instructed to attempt to maximize their success by choosing what they believe to be the correct option on each trial. Once a participants performance reaches a predefined criterion (or they have completed a maximum of six training blocks), the test phase begins. During the test phase, participants are shown all possible combinations of the six stimuli(fifteen total, four times each, for a total of 60 trials), and do not receive feedback upon selection. From the test phase, two different measures are calculated: the participants Choose accuracy, (the probability of choosing the higher valued alternative of any given fixed set from the training session, when it is paired with any other alternative), and the participants Avoid accuracy (the probability of choosing the lower valued alternative of any given fixed set from the training session, when it is paired with any other alternative). These measures can generally be interpreted to be the participants tendencies to pursue reward and avoid punishment, respectively.

When the participant in the PSS task is modeled as a Q-learning agent that employs an epsilon-greedy policy, some alarming results are observed. The agent

3 Basal ganglia anatomy

3. How can we fix this problem? One way is to look at the BG. Although the BG is often interpreted in terms of RL components, it does have peculiar physiological characteristics that are not easily explained in terms of model-free RL. One if this is the two pathways

4 Basal ganglia-inspired reinforcement learning

4. Let's put the two pathways in the RL model. Magic! Now the bias disappears

5 But wait, there's more

5. But the two pathways are also anti-correlated. What happens if we anticorrelate actions and anti-actions? Even more magic! The algorithm learns faster.

6 Conclusions

Conclusions

limitations

future directions - replicate/show better performance on other BG tasks?