A Comparison of Human and Agent Reinforcement Learning in Partially Observable Domains

Finale Doshi-Velez
Massachusetts Institute of Technology
Cambridge, Massachusetts
finale@alum.mit.edu

Zoubin Ghahramani
University of Cambridge
United Kingdom
zoubin@eng.cam.ac.uk

Abstract

It is commonly stated that reinforcement learning (RL) algorithms require more samples to learn than humans. In this work, we investigate this claim using two standard problems from the RL literature. We compare the performance of human subjects to RL techniques. We find that context—the meaningfulness of the observations—plays a significant role in the rate of human RL. Moreover, without contextual information, humans often fare much worse than classic algorithms. Comparing the detailed responses of humans and RL algorithms, we also find that humans appear to employ rather different strategies from standard algorithms, even in cases where they had indistinguishable performance to them.

Keywords: sequential decision-making; reinforcement learning; computational models; machine learning

The ability of humans to make sequential decisions under uncertainty has been widely studied in psychology and neuroscience. The field of *reinforcement learning* (RL) studies the theoretical formulation and algorithmic implementation of artificial agents that make sequential decisions to maximize their expected reward (Sutton & Barto, 1998). While RL algorithms often provide theoretical guarantees on the quality of the agent's long-term behaviour, the common lore in the RL community (Singh, 2009; Peters, Bagnell, & Schaal, 2006; Morimoto & Doya, 2005) is these approaches are painfully slow, requiring thousands of trials to learn to act in, what seem to humans, relatively simple domains.

While RL has been applied as a theoretical tool for understanding human decision making behaviour (Samejima & Doya, 2007; Daw, O'Doherty, Dayan, Seymour, & Dolan, 2006; Kakade & Dayan, 2002; Daw, Courville, & Tourtezky, 2006; Yoshida & Ishii, 2003; Acuña & Schrater, 2008; Dayan & Daw, 2008), the supposed "slowness" of RL methods has not been experimentally tested against human learning performance. Are these RL algorithms actually slower to learn than humans? To what extent is this lore biased by the fact that humans bring structural knowledge from previous experiences to new problems? For example, when entering a new building, a human will probably assume that he cannot walk through its walls, whereas RL problems would typically have to relearn this fact for each new location. Humans also tend to assume near-deterministic worlds, whereas RL algorithms are often initialized as believing all possible outcomes are equally likely.

In this paper, we focus on the approaches humans take on problems where aspects of the environment cannot be fully-observed (formally partially observable Markov decision processes (POMDPs)). POMDPs offer a more realistic scenario for decision making under uncertainty than the simpler (fully-observable) Markov decision processes, since they assume that the state of the world is known, but inferred from noisy observations. In this setting, we show that, surprisingly, when put in an identical setup on standard decision making problems, RL methods often learn faster and achieve better solutions than humans. Even more surprisingly, while human performance does improve when subjects are given contextual information about the problem, their average performance often still does not match RL methods. Our work has interesting implications for our understanding of both human and machine decision making. Without contextual information, humans may require more experience than RL algorithms to perform well even on simple problems. However, making use of context is one of the important open problems for machine learning.

Experiment

We tested two hypotheses: first, that human subjects would perform significantly better if given contextual observations, and second, that human subjects would outperform RL algorithms. Performance was evaluated as the sum of rewards obtained during the last tenth of a learning trial. We also examined which RL algorithms' behaviour most closely matched human behaviour.

Task Descriptions

The tasks consisted of two common problems in the RL literature, both formulated as POMDPs. Playing the role of the agent, the human subject—who had no initial knowledge about the structure of the problem—selected actions to take. The problem returned an observation, displayed on a computer screen, which depended on the underlying state of the environment, and an immediate reward. The subject's goal was to maximize their cumulative rewards.

Each task could be presented to the subject in two different versions. In the *with-context* version $C+$, the domain's observations had meaning in the context of the task. In the *context-free* version $C-$, observations had no meaning; the $C-$ version of the problem was meant to simulate what a RL algorithm might "see," as a computer system cannot attribute meaning or significance to particular observations.

In the first problem, the tigerworld task (Kaelbling, Littman, & Cassandra, 1995), players were confronted with two doors (see figure **??** for an illustration). Behind one door was a tiger (reward = -100); behind the other was a prize (reward = $+10$). At every iteration, players had three options:

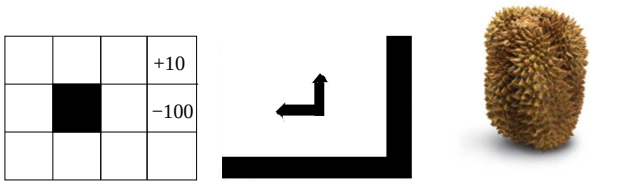
they could open one of the two doors, or they could "listen" for more information. Each listen attempt had an 85% chance of being accurate and an associated reward of -1. In the $C+$ version, the observations were images of a tiger on the left or the right of the image. In the without-context version, the image of a tiger on the right was replaced with an image of an apple, and the image of a tiger on the left was replaced by an image of a banana. The text on the actions ("listen," "open left," "open right") was also replaced by numbers ("1," "2," "3"). Opening either door reset the tiger and the prize to random positions.[1] Understanding that listening provided useful but noisy information was the key learning challenge in the tigerworld task.

(a) Illustration of the Tigerworld Problem (b) Sample $C+$ Observation (c) Sample $C-$ Observation

Figure 1: The tigerworld problem consisted of two doors. One door had a tiger behind it, the other a prize. Players could open a door or "listen" for the tiger's location. The right two images show a possible result of the "listen" action in the $C+$ and $C-$ versions, respectively.

The second problem, the gridworld task, players had to navigate from a random starting place on a 4x3 grid (Russell & Norvig, 2010) to reach the prize in the top-right corner (see figure 2 for the map). Reaching either the prize (reward = 10) or the penalty (reward = -100) square reset the player to an arbitrary position on the board. Unlike in the tigerworld task, the observations in the gridworld task were deterministic—players always saw the walls immediately around them. However, actions had stochastic effects: 80% of the time the action would execute as expected; 20% of the time the player would find themselves moved in a perpendicular direction. Reaching the prize square while navigating around the penalty square was the key learning challenge in the gridworld task. In the contextual version of the problem, the subjects saw gridcells with walls and arrows as observations (figure 2(b)) for normal cells and a happy or sad emoticon for the two reward cells. Action buttons were labeled with the compass directions; subjects reported no trouble making the association between the compass directions on the action buttons and the arrows indicating free directions to move in the observations. In the $C-$ version, each unique observation was mapped to a specific fruit. The action buttons were also numbered instead of labeled with compass directions. Rewards in both tasks were deterministic functions of the underlying hidden state.

[1]Opening a door in the original version of tigerworld results in a random observation. In pilot trials, subjects found this version very hard to learn; therefore, we augmented tigerworld with a third "reset" observation that always followed an open-door action.



(a) Illustration of the Gridworld Problem (b) Sample $C+$ Observation (c) Sample $C-$ Observation

Figure 2: The gridworld task rewarded players for reaching the top-right corner of a 4x3 grid. The right images show the observation for the bottom-right corner.

Methods

Procedure To test the first hypothesis, each subject played every task-version pair (tigerworld, with and without context; gridworld, with and without context). Subjects were informed they were learning four different tasks. Each subject played 500 iterations in tigerworld and 750 iterations in the gridworld. The simpler tigerworld problem was always presented before gridworld. The length of the experiment and the decision to present the simpler problem first were decided from an initial pilot test.

The ordering of the two versions was counterbalanced between the subjects: half the subjects received the $C+$ tasks first; half received the $C-$ tasks first. Subjects playing the $C+$ versions first had slightly better overall performance than subjects playing the $C-$ version first ($t(31) = 2.32, p < 0.05$). To check if subjects were using learned effects of actions from one task version to the next, the labels associated with the actions in the $C-$ tasks were either ordered identically as the $C+$ versions or permuted. For example, if the $C+$ version had buttons 'left,' 'right,' 'listen,' then the numbers '0,' '1,' '2' could either map to 'left,' 'right,' 'listen' (same order) or 'right,' 'listen,' 'left' (permuted order). Subjects were split evenly between these two versions; we found that changing the action mapping had no significant effect on performance ($t(31) = 0.88, p > 0.10$).

After signing a consent form, subjects were shown the interfaces and given a chance to familiarise themselves with it. They were also told the following information:

- Each task was unique and unrelated to the other tasks.
- Actions could have stochastic effects, but there were no adversarial effects.
- Past (especially recent) observations could be important.
- They could take notes or use a calculator if they wished.
- There was no time limit.
- The trials would be long enough that they should feel free to spend time exploring.

After all trials were complete, subjects were interviewed on how they approached the problem. They were encouraged to explain any sketches or computations they had made. Finally, subjects were asked if they had realised that the tasks were paired (3 of 16 subjects did). Each version took subjects

15-20 minutes to complete; the entire set of tasks took most subjects 60-90 minutes. Subjects were allowed to take breaks between tasks.

To test our second hypothesis, we collected a fresh group of subjects. Each subject played one version (with/without context) of tigerworld for 3000 iterations and one version of gridworld for 2000 iterations. The trial lengths were chosen based on pilots showing that human subjects varied greatly in their of learning rates and "inspiration" moments. Half the subjects played the $C+$ tigerworld scenario and the $C-$ gridworld scenario; the other half played the $C-$ tigerworld scenario and the $C+$ gridworld. Subjects were given the same instructions as in the first experiment. These longer trials lasted 90-150 minutes; subjects were encouraged to take breaks whenever they wished to avoid fatigue.

Aparatus The subjects participated in the study by using a mouse to click buttons displayed on a computer screen. The display had three elements. A large central pane showed the current observation (updated after each action). Above the observation pane was a panel that showed subjects their immediate reward after each action (cumulative rewards were not shown). Finally, a set of action-selection buttons were located below the observation window.

Subjects could not access prior histories of actions, observations, or rewards; however, they were provided pen and paper. Subjects could also use of a calculator (none did).

Participants To test the first hypothesis, that context had a significant effect in human learning, 16 subjects (13 male, 3 female) were recruited from the University of Cambridge Engineering Department. To test the second hypothesis, eight additional subjects were recruited from the University of Cambridge Engineering Department. Finally, three additional subjects (2 male, 1 female) participated in a pre-trial pilot. Participants were compensated for their participation; a prize was also offered for the highest score.

Results

Effect of Context

We had hypothesized that subjects would perform better in $C+$ versions of each problem. Performance was evaluated based on the sum of all immediate rewards received during a trial. Subjects performed significantly better with context than without, paired $t(31) = 2.99, p < 0.005$, with a mean benefit of $1,243$ points in the final cumulative reward.

The total reward gained over time is shown in Fig. 3. The trials are broken into blocks of 50 iterations, and the shaded regions show the standard error of the mean. The upward trends in all curves indicates learning occurred during the course of the task. In the tigerworld problem, the $C-$ case started with a low initial reward, but by the end, the human subjects were performing as well with context as without (although still suboptimally). In contrast, the human subjects on the $C-$ version of gridworld never matched the $C+$ performance: many subjects inferred the grid when given contextual observations, but only one inferred the map when fruit images were substituted for the wall images.

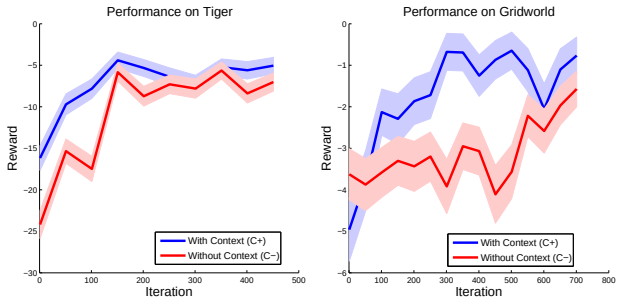


Figure 3: Reward in each phase of the trial. Blocks consisted of 50 iterations for both problems. Shaded regions show the standard error of the mean.

Comparison to Reinforcement Learning Algorithms

We next compared human performance to three approaches from RL: RMAX (Brafman & Tennenholtz, 2002), u-tree (McCallum, 1995), and iPOMDP (Doshi-Velez, 2009). The first, RMAX, builds a model of the world's dynamics, choosing optimistic rewards for parts of the world it has not seen, and then uses the model to make decisions. RMAX is designed for fully-observable problems, that is, problems with no hidden information. To apply RMAX to our domains, we use a history of recent observations as a proxy for the state, a technique often used for tackling partially-observable problems (Breslow, 1996; Lin & Mitchell, 1992).

Specifying how much past history to consider adds an additional parameter to RMAX; the u-tree algorithm tries to dynamically learn the window size: it uses a series of statistical tests to increase the number of past observations considered if it enables the agent to improve its overall rewards. Like RMAX, u-tree builds a model using each of these (now variable length) past histories as states and solves the model to select actions. Finally, iPOMDP also builds a model of the world first, but it does not assume that the world is fully-observable; indeed, it assumes that the number of hidden states could be potentially unbounded. While iPOMDP correctly models the true partially-observable nature of the problems, it must search over a much larger class of models.

We had hypothesized that subjects would perform better than the RL algorithms when given context and worse when not given context. We tested both performance on the last tenth of the data as well as compared the performance of the subjects and the algorithms for each block of 50 interactions during the trial. On the tigerworld problem, the algorithms outperform the subjects without context both in the last tenth of the trial, $t(1602) = -4.82, p < 0.005$ and in each block of the learning process, $t(59) = -12.79, p < 0.005$. More surprisingly, the algorithms also outperformed subjects *when they had context* both in the last tenth of the trial, $t(3003) = -5.76, p < 0.005$ and throughout the learning process, $t(59) = -10.92, p < 0.005$. These results directly

contradict conventional wisdom that while an RL algorithm might eventually produce a superior solution than a human, they generally learn more slowly.

The left pane of Fig. 4 compares the performance of the three RL algorithms to human subjects without context on the tigerworld problem. As before, the shaded regions show the standard error of the mean, and averages are computed for blocks of 50 iterations; the expected optimal performance—computed by applying value iteration to each domain—for an agent that knew the domain is given by the dashed line (note that the expected optimal performance is the average performance an optimal agent would gain over many runs; individual runs can exceed this value). What is striking is how quickly RMAX and iPOMDP algorithms achieve near-optimal performance; u-tree, testing variable window lengths, learns slower but also ultimately bests the human subjects.

Recall that the key challenge in the tigerworld problem was learning that the observations of where the tiger was located were noisy: repeated measurements were needed to ascertain the tiger's location to a reasonable degree of accuracy. The gridworld problem tested a different challenge: building a map of a domain where actions sometimes "slipped" or had unexpected results. As seen in the right pane of Fig. 4, the difference in performance in the gridworld problem is much less clear. We found no significant difference between the RL algorithms and the human subjects: still, it is interesting to note that even when given the context of the walls and corridors, the human subjects did not outperform the algorithms, which did not have access to this information. The difference in the RMAX algorithm's performance through the learning process (again measured as the performance in each block of 50 interactions) was significantly greater than the human subjects' performance, $t(39) = -3.81, p < 0.005$. Finally, it is interesting to note that the iPOMDP algorithm performs the most poorly in this domain. The extra complexity of having to explicitly consider the partial observability in these relatively simple domains results in a much slower learner.

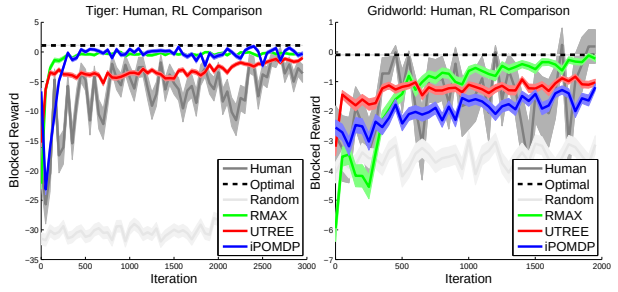


Figure 4: Reward for each trial block of 50 iterations. Shaded regions show the standard error of the mean. For RMAX, the window size was 2.

Recall that RMAX uses the most recent window of observations as a proxy for the hidden state. We tested the algorithms with windows ranging from only the most recent observation to the last four observations. The results for RMAX are shown in Fig. 5. In the tigerworld problem, the small window sizes yield similar (suboptimal) performance levels as the human subjects, but much more quickly. The longer window sizes result in slower learning, but they eventually out-perform the human subjects *regardless of whether the subjects had context*. RMAX's learning rate is even more striking in the gridworld problem (right panel of Fig. 5). The longer window sizes, with a large number of parameters ($O(S^2)$, where S is the number of states), are very slow to learn, but building a model reduces the need for long windows: the small-window learners quickly surpass human performance. In post-experiment interviews, most human subjects also showed maps that they had built as they played. What then distinguished RMAX? We hypothesize the crucial difference was RMAX's optimistic approach to filling in unknown parts of the model, which lead it to explore all aspects of the problem. In contrast, humans in post-experiment interviews claimed they behaved much more cautiously after discovering a -100 penalty.

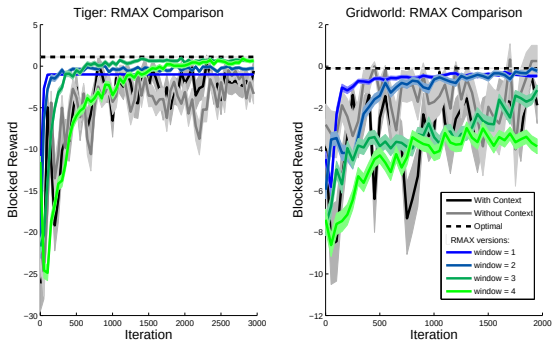


Figure 5: Comparison with RMAX over several window sizes. The dip in the human performance curve in gridworld was due to a single subject's performance.

The previous analysis found that the RL algorithms often outperform average human performance, both with and without context. When compared to the best human subject (as measured by cumulative performance on the last tenth of the trial), we find that the best subject outperformed the algorithms on gridworld when given context: the best subject scored 33 times more points in the last tenth of the trial than RMAX, the best algorithm. However, the best human did *not* outperform the best algorithm in tigerworld—RMAX scored 6 times as many points as the best human. The plots of the best human subject's performance are shown alongside the best algorithms in Fig. 6. Interestingly, the best human subject appears to learn a (suboptimal) solution the tigerworld problem slightly faster than the algorithms, but the gridworld problem takes longer to learn (though the performance is near-optimal in the end).

Algorithms Matching Human Behaviour

Finally, we examined which RL methods most closely modelled human behaviour. To evaluate how well these RL procedures predicted human subjects' behaviour, we played each

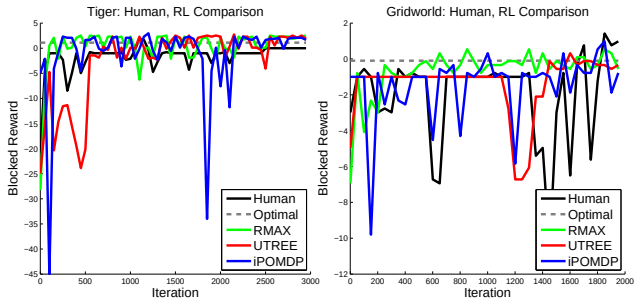


Figure 6: Performance of best human subject (with context). In both domains, the subject learns a near-optimal solution.

subject's history of actions and observations forward to the RL agent. At each iteration, the RL algorithm updated its internal state given the current history of the human subject's actions. Based on this history, the algorithm decided which action it would select next. Similarity between the algorithm's and the human subject's behaviour was assessed along two criteria: (1) whether the agent's selected action matched the subject's selected action,[2] and (2) the algorithm's *regret* given the human subject's action choice. Formally, regret is the value the agent thought the human subject lost by his or her choice of action (as computed from the algorithm's internal value function). Lower regrets imply a greater similarity between the algorithms' and the human subjects' choices.

The results in Fig. 7 show that none of the algorithms matched human behaviour very often, regardless of whether the subjects had context. The algorithms matched the subjects slightly better in the tigerworld domain than the gridworld; the low-match rates—almost always below 50%—suggest that the humans and the algorithms were employing rather different strategies, even when they had indistinguishable performance (as in gridworld). As expected, the RMAX learner had the highest regrets; its optimistic initialization made it believe that humans often under-explored problems. The u-tree algorithm had the lowest regrets, in part because it tended to be less certain about the correct action at any time.

Finally, we note that the stochasticity of the problems (seen in the individual problem traces in Fig. 6) resulted in high reward variances and that different under-exploring policies could also result in large reward variations (seen in the high standard errors in Fig. 4. The analysis in this section shows that there are differences in how humans and RL agents explore given these high variances.

Discussion

A significant advantage that RL algorithms have over humans is that they do not get bored, fatigued, or disheartened. In a long series of experiments in which subjects may accrue large costs before ultimately learning a good strategy, these factors often caused humans to settle for sub-optimal or reasonable solutions instead of seeking better solutions. In contrast, the RL algorithms were more persistent; in general they not only learned as quickly or quicker than human subjects, but they also refined their solutions more than human subjects. Thus, we find that contrary to conventional wisdom about these simple algorithms—that they learn slowly—these algorithms often learn significantly faster than human subjects.

The quantitative performance curves matched post-experiment interviews in which the subject (like many others) produced an accurate map of the gridworld—despite the transition uncertainty and location ambiguity—but found it very difficult to reason about the observation uncertainty in tigerworld. The algorithms treated both of these forms of uncertainty equivalently; thus they learned in proportion to the overall level of uncertainty. We can conclude that either humans require more experience to learn than supposed, RL algorithms are faster learners, or both.

Our work is consistent with studies showing humans have difficulty planning under uncertainty, though none directly compare human and algorithm performance in multi-state partially-observable domains. For example, handling location ambiguity was found to be the primary bottleneck for humans trying to perform spatial navigation tasks (Stankiewicz, McCabe, & Legge, 2004). Gureckis and Love (2009) found slightly noisy rewards encouraged exploration, but humans are generally poor at handling randomness, even in fully-observable settings. Finally, Acuña and Schrater (2008) hypothesised that humans may learn slowly on bandit-type problems because they consider a wider set of underlying structures, even when they are told that the problem has a particular form. They showed that human learning rates on a 1-state partially-observable problem are slower than an approach that leverages the structure of the problem (also given to the human subjects) but similar to an approach that makes fewer structural assumptions. Their results are similar to the differences we observed between the RMAX algorithm—which learned quickly due to its simple model assumptions—and the iPOMDP or u-tree—which learned more slowly.

The findings in this work are based on two standard problems in RL, with relatively small state spaces. We conjecture that without context, the advantage of RL methods over humans will persist for larger state spaces. For example, given no context, a larger gridworld is even more baffling for the human subject who was already—on average—confused by a 4×3 grid. However, in larger *more structured* state spaces, the human subject's ability to generalise and make use of context would probably give them significant advantages. For example, human subjects may infer that "stacking" actions put one object on top of another, while a simple agent may have to learn the result of a "stack" for each pair of objects. Similarly, humans may use patterns of grammar to analyse dialogues, whereas an agent might have to learn each part of a conversation separately. It remains an interesting open question as to how the learning rates of human subjects and RL agents compare on these more structured and hierarchi-

[2]The results evaluating action-selection similarity based on a softmax action-selection criterion were nearly identical to action-matching and are omitted for brevity.

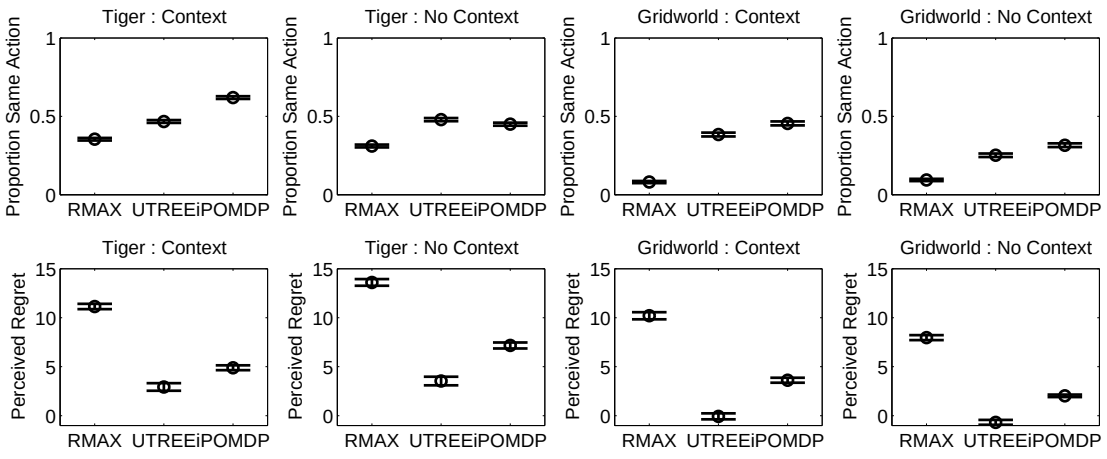


Figure 7: Proportion of same actions and perceived regret of the yoked learners. A higher proportion of same actions indicates a greater similarity between the human's and agent's decisions; likewise lower regrets indicate that the agent valued actions similarly to the human subjects. Means are shown with 95% confidence intervals.

cal learning domains. The importance of context in human learning also suggests that for work trying to build more data-efficient artificial agents (Fei-Fei, Fergus, & Perona, 2006), learning and leveraging contextual information may be key factor to achieving better learning performance. An exciting avenue for future work would be to better understand how humans leverage context when learning a task, rather than focusing simply on their rates of learning.

Acknowledgments

Finale Doshi-Velez was funded by the Marshall Scholarship. Additional funds were provided by the Computational and Biological Learning Laboratory at Cambridge University.

References

Acuña, D., & Schrater, P. R. (2008). Structure learning in human sequential decision-making. I *NIPS*.

Brafman, R. I., & Tennenholtz, M. (2002). R-max – a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, *3*.

Breslow, L. (1996). *Greedy utile suffix memory for reinforcement learning with perceptually-aliased states* (teknisk rapport). Navy Research Center Laboratory.

Daw, N. D., Courville, A. C., & Tourtezky, D. S. (2006). Representation and timing in theories of the dopamine system. *Neural Comput.*, *18*(7), 1637–1677.

Daw, N. D., O'Doherty, J. P., Dayan, P., Seymour, B., & Dolan, R. J. (2006). Cortical substrates for exploratory decisions in humans. *Nature*, *441*(7095), 876–879.

Dayan, P., & Daw, N. D. (2008). Decision theory, reinforcement learning, and the brain. *Cognitive, Affective, and Behavioral Neuroscience*.

Doshi-Velez, F. (2009). The infinite partially observable Markov decision process. I *Nips*.

Fei-Fei, L., Fergus, R., & Perona, P. (2006). One-shot learning of object categories. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *28*(4), 594–611.

Gureckis, T. M., & Love, B. C. (2009). Learning in noise: Dynamic decision-making in a variable environment. *Journal of Mathematical Psychology*, *53*(3), 180–193.

Kaelbling, L. P., Littman, M. L., & Cassandra, A. R. (1995). Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, *101*, 99–134.

Kakade, S., & Dayan, P. (2002). Dopamine: generalization and bonuses. *Neural Networks*, *15*, 549-559.

Lin, L.-J., & Mitchell, T. M. (1992). *Memory approaches to reinforcement learning in non-markovian domains* (teknisk rapport). Carnegie Mellon University.

McCallum, A. (1995). Instance-based utile distinctions for reinforcement learning with hidden state. I *Icml*.

Morimoto, J., & Doya, K. (2005). Robust reinforcement learning. *Neural Computation*, *17*(2), 335-359.

Peters, J., Bagnell, D., & Schaal, S. (2006). *Towards a new reinforcement learning?*

Russell, S., & Norvig, P. (2010). *Artificial intelligence: A modern approach*. Prentice Hall.

Samejima, K., & Doya, K. (2007, April). Multiple Representations of Belief States and Action Values in Corticobasal Ganglia Loops. *New York Academy Sciences Annals*, *1104*.

Singh, S. (2009). *RL is slow.* Available from `http://umichrl.pbworks.com/RL-is-slow`

Stankiewicz, B. J., McCabe, M., & Legge, G. (2004). Studying human spatial navigation processes using pomdps. I *AAAI workshop on learning and planning in markov processes: Advances and challenges* (s. 97-102).

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction.* MIT Press.

Yoshida, W., & Ishii, S. (2003). A model-based reinforcement learning: a computational model and an fmri study. I *ESANN* (s. 313-318).