Published in final edited form as:

Proc Mach Learn Res. 2023 August ; 216: 1047–1057.

Assessing the Impact of Context Inference Error and Partial Observability on RL Methods for Just-In-Time Adaptive Interventions

Karine Karine[1], **Predrag Klasnja**[2], **Susan A. Murphy**[3], **Benjamin M. Marlin**[1]

[1]University of Massachusetts Amherst

[2]University of Michigan

[3]Harvard University

Abstract

Just-in-Time Adaptive Interventions (JITAIs) are a class of personalized health interventions developed within the behavioral science community. JITAIs aim to provide the right type and amount of support by iteratively selecting a sequence of intervention options from a pre-defined set of components in response to each individual's time varying state. In this work, we explore the application of reinforcement learning methods to the problem of learning intervention option selection policies. We study the effect of context inference error and partial observability on the ability to learn effective policies. Our results show that the propagation of uncertainty from context inferences is critical to improving intervention efficacy as context uncertainty increases, while policy gradient algorithms can provide remarkable robustness to partially observed behavioral state information.

Keywords

Reinforcement learning; partial observability; context inference; adaptive interventions; empirical evaluation; mobile health

1 INTRODUCTION

Just-in-Time Adaptive Interventions (or JITAIs) are a class of personalized health intervention developed within the behavioral science community [Nahum-Shani et al., 2018. Hardeman et al., 2019, Battalio et al., 2021, Yang et al., 2023. Perski et al., 2022]. The primary goal of JITAIs is to provide the right type and amount of support for each individual as their personal and environmental context varies over time [Nahum-Shani et al., 2018]. JITAIs aim to accomplish this goal by using decision rules to select from among a collection of possible intervention options based on observed and inferred dimensions of an individual's state.

While current JITAI's and related adaptive intervention designs leverage increasingly sophisticated wearable sensors and machine-learning based context inference methods [Battalio et al., 2021], JITAI decision rules are still largely developed using an expert systems approach [Perski et al., 2022]. In this work, we investigate the application of neural

network-based reinforcement learning (RL) methods [Williams, 1992, Mnih et al., 2013] to the problem of learning intervention option selection policies for JITAIs using a novel simulation environment that captures key behavioral concepts including habituation and risk of disengagement with an intervention.

We focus on two foundational issues with the application of RL algorithms to JITAIs. First, we investigate the impact of context inference error on the performance of learned policies. Second, we investigate the impact of non-observability of psychological state variables on policy learning. We note that neither of these issues has received attention in prior work and current JITAIs routinely leverage machine learning-based context inferences that discard prediction uncertainty.

Our primary contributions are: (1) the development of a physical activity JITAI simulation environment that captures key aspects of the dynamics of behavior in the context of adaptive interventions; and (2) the quantitative evaluation of the impact of context inference error, context inference uncertainty and partial observability on the performance of policies learned using different categories of reinforcement learning approaches including policy gradient methods and value function methods.

Our results show that policies that leverage context inference probabilities as features can significantly outperform policies that use only the most likely context value. Second, our results show that non-observability of psychological state variables has a drastic impact on the quality of polices learned using value function methods, but a significantly more modest effect on policy gradient methods. These results have important implications for the design of RL methods for use in JITAI applications.

The remainder of this paper is organized as follows. In Section 2 we provide background on JITAIs and reinforcement learning methods. In Section 3 we present the methods used in our experiments including the description of the physical activity JITAI simulation environment. In Section 4 we present experiments and results. We conclude with a discussion in Section 5

2 BACKGROUND AND RELATED WORK

In this section we provide a brief overview of research on JITAIs and background on reinforcement learning methods.

2.1 JUST-IN-TIME ADAPTIVE INTERVENTIONS

As noted in the introduction, JITAIs are a class of personalized health intervention developed within the behavioral science community that aim to provide the right type and amount of support for each individual as their personal and environmental context varies over time [Nahum-Shani et al., 2018]. JITAI's and related adaptive study design have been applied in multiple critical health domains including physical activity [Hardeman et al., 2019], smoking cessation [Battalio et al., 2021, Yang et al., 2023] and addiction [Perski et al., 2022].

JITAIs are comprised of three main parts: the set of intervention components that can be provided to an individual and the specific intervention options within each component; a

set of decision time points that determine when intervention components can be provided to an individual, and a policy that determines which intervention option to select for a given individual in a given context. Many current JITAIs are sophisticated cloud-supported mobile software applications that leverage a variety of intervention components from planning to goal setting to contextually tailored messaging and content delivered from auxiliary apps (such as mindfulness and stress reduction exercises) [Perski et al., 2022, Spruijt-Metz et al., 2022].

While early JITAIs were largely based on self-report of context information, current JITAIs are increasingly making use of machine learning-based context inferences derived from data collected from smart phones and wearable sensors. For example, recent work in adaptive intervention design for smoking cessation support [Battalio et al., 2021] leverages customized wearables [Ertin et al., 2011, Kwon et al., 2021] and machine learning models for the detection of stress [Hovsepian et al., 2015] as well as smoking lapse [Saleheen et al., 2015].

Despite the sophistication of JITAIs as software applications, the complexity of component and option selection policies has remained relatively limited. While the policies are adaptive in the sense of selecting different content in different contexts, the context-to-content mappings are often hand-designed by the intervention designers. While this allows intervention designers to build selection policies that are based on behavioral theory, there is significant need for methods that can refine expert policies as well as learn de novo policies from data.

To this end, a number of domains where JITAIs are being deployed admit meaningful and continuously measurable proximal outcomes that can be used as a reward signal for reinforcement learning algorithms. For example, in the physical activity domain, wearable activity tracking devices such as FitBit devices and smart watches can be used to detect both the duration of sedentary episodes as well as steps [Spruijt-Metz et al., 2022]. We turn next to a brief review of reinforcement learning and return to a discussion of the challenges of applying RL methods in the JITAI context at the end of this section.

2.2 REINFORCEMENT LEARNING

The goal of reinforcement learning (RL) methods is to learn a policy that optimizes the selection of actions in a sequential decision making problem [Sutton and Barto, 1998]. A sequential decision making problem is formalized as a Markov decision process or MDP $(\mathcal{S}, \mathcal{A}, P, R)$ where: $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, P defines the state transition probability distribution $P(s' \mid s, a)$ and R defines the reward function $R(s, a, s')$ for taking action a in state s and then transitioning to state s'. A policy π is a function that maps states into actions. An episode in an MDP consists of a sequence of state, action, reward tuples (s_t, a_t, r_t). Starting with an initial state s_0, an episode proceeds according to the policy, state transition distribution and reward function until an absorbing state is reached [Sutton and Barto, 1998].

In this work, we focus on two classes of reinforcement learning methods: policy gradient methods and value function methods. Policy gradient methods learn a probabilistic model π_θ

mapping states into a probability distribution over actions. Value function methods instead learn the value of states or state-action pairs. The domain that we focus on in this work has a factorized state space that includes continuous dimensions, thus we focus on value function methods that can accommodate continuous state variables. We briefly review both classes of methods.

Policy Gradient Methods: The goal of policy gradient methods is to select the parameters θ of the policy π_θ to maximize the expected return of the policy: $J(\pi_\theta) = \mathbb{E}_{\tau \sim \pi_\theta}[R(\tau)]$. Here $R(\tau)$ is the return over a trajectory τ. A trajectory is a sequence of states and actions: $\tau = (s_0, a_0, s_1, a_1, \ldots s_{T-1}, a_{T-1}, s_T)$ where T is the episode length.

Different policy gradients methods use different definitions of the return $R(\tau)$. In this work we focus on the basic REINFORCE algorithm, which uses a return based on the discounted sum of rewards to go. Policy gradient methods learn the parameters of the policy using a Monte Carlo approximation to the gradient of the expected return function using M sampled trajectories per gradient update [Sutton et al., 1999. Williams, 1992] as shown below where γ is the discount rate and $G_t\left(\tau^{(i)}\right)$ is the reward to go function.

$$\theta_{t+1} \leftarrow \theta_t + \alpha \hat{\nabla} J(\pi_\theta) \tag{1}$$

$$\hat{\nabla} J(\pi_\theta) = \frac{1}{M} \sum_{i=0}^{M-1} \sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t^{(i)} | s_t^{(i)}) G_t\left(\tau^{(i)}\right) \tag{2}$$

$$G_t\left(\tau^{(i)}\right) = \sum_{k=t}^{T-1} \gamma^{t-k} r_t \tag{3}$$

One of the interesting properties of REINFORCE as a pure Monte Carlo policy gradient method is that the correctness of the above learning rule and the convergence of the learning algorithm hold in the case where both the policy π_θ is modeled using a non-linear function approximator and we only have access to partially observed state vectors s_t' relative to the full MDP state s_t. While REINFORCE is known to have high variance, more sophisticated policy gradient methods such as Actor-Critic methods do not have convergence guarantees in continuous state spaces with partially observed state. We also note that while methods like the use of a baseline in the return formulation can also decrease variability, we do not see convergence issues in our experiments when using sufficiently large M.

Value Function Methods: While policy gradient methods aim to directly learn an optimal policy, value function methods such as Q-learning aim to learn the value of state-action pairs and derive a policy by selecting actions that have maximal value in each state [Sutton and Barto, 1998]. In classical Q-learning for discrete state spaces, the state-action value function $Q(s, a)$ is simply a lookup table. More generally, Q-learning can be applied using a function approximator for $Q(s, a)$, which allows Q-learning to be extended to

continuous state spaces. For example, the Deep Q Network (DQN) approach uses a deep neural network to approximate $Q(s, a)$ [Mnih et al., 2013].

DQN approaches learn using backpropagation applied to a regression loss $\ell(\delta_t)$ that is a function of the temporal difference error $\delta_t = r_t + \gamma \cdot \max_{a' \in \mathcal{A}} Q(s_{t+1}, a') - Q(s_t, a_t)$. Fully online learning can be applied after taking each action, but performance can be improved in number of ways including minimizing the loss applied to the temporal difference computed from a batch of examples sampled from a replay buffer and using a second copy of the Q network that is updated more slowly in place of $Q(s_{t+1}, a')$ deBruin et al., 2015, Schaul et al., 2016.

In this work we use the Dueling DQN variant with a replay buffer as an example approach of this class. In the Dueling DQN approach, the Q network is split into two components: a state value function $V(s)$ and a state-dependent advantage function $A(s, a)$. The $Q(s, a)$ value is computed by summing the state value and the advantage value: $Q(s, a) = V(s) + A(s, a)$. The average advantage value $\bar{A}(s) = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} A(s, a)$ can also be subtracted from the raw advantage value $A(s, a)$ to improve identifiability [Wang et al. 2016]. The model is again learned by minimizing a loss on the temporal difference error. This approach also uses more slowly updated copies of these networks when computing the target $Q(s_{t+1}, a')$ values.

We note that unlike standard Monte Carlo policy gradient methods, Q-learning methods including the Dueling DQN have the ability to learn from trajectories that were not sampled from the current model parameters. This off-policy learning ability allows Q-learning methods to use a replay buffer and provide better sample efficiency. However, Q-learning methods have the significant drawback that their convergence is not guaranteed in a setting where the state is partially observed and state-action values are represented using non-linear function approximators, including neural networks.

2.3 RL FOR JITAIS

Prior work on RL methods for JITAIs has largely focused on contextual bandit methods [Paredes et al., 2014, Rabbi et al., 2015, Tewari and Murphy, 2017, Yom-Tov et al., 2017]. These methods aim to select actions that maximize the immediate expected reward, thus discounting longer term effects of actions. However, adaptive health intervention domains can have significant long term and delayed effects. To address this challenge Liao et al. [2020] develop an extended bandit-like algorithm that uses a model-based proxy reward to imitate the longer term effect of actions. Gönül et al. [2021] propose an RL method that uses modified eligibility traces that aim to credit intervention components that the participant actually engaged with. The core RL algorithm used is based on Q-learning, but assumes that discrete states are provided by an auxiliary state classifier.

While both Liao et al. [2020] and Gönül et al. [2021] represent improvements over contextual bandit methods in terms of their ability to model longer term effects of actions, both approaches condition on context variables as if they are known without uncertainty, which is the specific issue we study in this work. Further, through the use of the auxiliary state classifier, Gönül et al. [2021] avoid issues that arise when composing Q-learning

methods with function approximation under partial observability, which we also address directly.

Finally, we note that Liao et al. [2020] articulate multiple important practical challenges with the deployment of RL methods for JITAIs including the need for methods that can learn quickly from limited interactions with single individuals. In this work our primary goal is to quantify the fundamental limits imposed by context inference error and partial observability. As a result, we do not consider restrictions on the number of simulated interactions with a user or restriction on the number of episodes of training. Our results should be interpreted as establishing upper bounds on the performance achievable by methods that impose further constraints.

3 METHODS

In this section we describe the physical activity JITAI simulation environment that we use in this work as well as the context error and partial observability conditions that we study. We also describe in detail the reinforcement learning agents used in our experiments.

3.1 PHYSICAL ACTIVITY JITAI SIMULATION ENVIRONMENT

We design a JITAI simulation environment taking inspiration from recent work in the area of contextualized messaging based intervention studies for promoting walking as a form of physical activity [Hardeman et al., 2019, Spruijt-Metz et al., 2022]. Below we describe the state, action, and dynamics of the physical activity JITAI simulation.

State and Actions: A contextualized messaging intervention leverages a pool of messages that aim to provide support in different contexts. The choice of whether and what type of message to send at each time step depends on the individual's context c_t. We select stressed/not stressed as an example binary context variable in our simulation. As discussed in the previous section, such context variables are often derived from sensor-based inferences [Hovsepian et al., 2015]. To reflect the fact that the true context is not known to the reinforcement learning agent, we use $\mathbf{p}_t$ to denote an inferred probability distribution over the context, and l_t to represent the most likely context value according to $\mathbf{p}_t$.

In addition to the stressed/not stressed context variable, we model two additional psychological state variables: habituation h_t and disengagement risk d_t. Intuitively, habituation models the extent to which the effect of the intervention is attenuated through prior exposure to the intervention. Disengagement risk facilitates modeling a common problem with adaptive interventions: in response to factors such as perceived lack of utility, intervention participants sometimes completely abandon the use of an intervention. We discuss the dynamics of these variables in the next section.

We summarize the variables in the simulation and their value ranges in Table 2 (note that Δ^1 indicate the probability simplex for a binary variable). The simulation includes a total of four actions as summarized in Table 1 Action $a = 0$ is the null action where no message is sent. Action $a = 1$ corresponds to sending a non-context tailored message. Actions $a = 3$ and $a = 4$ correspond to sending messages tailored to context 0 and 1 respectively. Note that

based on the numerical context and action values, $a_t = c_t + 2$ corresponds to the selection of a message that is tailored for the correct context. In response to taking an action in a given state at time t, we observe a reward in the form of a step count s_t.

Dynamics: We focus on simulating the dynamics of habituation and disengagement and how they relate to the effect of the intervention components. We model habituation as increasing with each message sent up to an upper limit and decaying towards zero when messages are not sent. We model disengagement risk as increasing only when incorrectly contextualized messages are sent and decaying towards zero only when uncontextualized or correctly contextualized messages are sent. We provide the update equations for these state variables below. The parameters of the update equations are described in Table 3

$$h_{t+1} = \begin{cases} (1-\delta_h)\cdot h_t & \text{if } a_t = 0 \\ \min(1, h_t + \epsilon_h) & \text{otherwise} \end{cases}$$

$$d_{t+1} = \begin{cases} d_t & \text{if } a_t = 0 \\ (1-\delta_d)\cdot d_t & \text{if } a_t = 1 \text{ or } a_t = c_t + 2 \\ \min(1, d_t + \epsilon_d) & \text{otherwise} \end{cases}$$

We model the reward in terms of the surplus step count generated beyond a potentially context dependent baseline level μ_c. We model incorrectly contextualized messages and not sending a message as generating zero surplus reward. We model uncontextualized actions and correctly contextualized actions as providing base surplus rewards ρ_1 and ρ_2 that are attenuated by the habituation level h_t. Specifically, as the habituation level increases, the fraction of the base reward that is realized decreases. While increasing disengagement risk does not have an immediate effect on reward, if the disengagement risk reaches the value 1, we simulate the occurrence of a disengagement event that terminates the episode. This delayed effect can have a significant impact on total reward over an episode. The maximum length of an episode is set to 50 time steps.

$$s_{t+1} = \begin{cases} \mu_{c_t} + (1 - h_{t+1})\cdot \rho_1 & \text{if } a_t = 1 \\ \mu_{c_t} + (1 - h_{t+1})\cdot \rho_2 & \text{if } a_t = c_t + 2 \\ \mu_{c_t} & \text{otherwise} \end{cases}$$

We model the true context as a purely random Bernoulli process. At each time step we sample $c_t \sim \text{Bernoulli}(0.5)$. To model a sensor-derived inference for c_t, we follow a two step process. We sample a normally distributed context-dependent scalar feature $x_t \sim \mathcal{N}(c_t, \sigma^2)$ where σ models the uncertainty in the feature given the context. We next compute the context probability distribution $\mathbf{p}_t$ given the sampled feature value x_t as $p_{ct} = P(C_t = c \mid x_t)$ simulating the application of probabilistic context classifier. Finally, we set the most likely context to $l_t = \text{argmax}_c\, p_{ct}$. We vary the feature noise standard deviation parameter σ from 0.4 to 2. This generates context inference errors varying from 10% to 41%. Figure 1 shows the effect of the feature noise standard deviation parameter σ on the context inference error rate.

3.2 CONTEXT INFERENCE AND PARTIAL OBSERVABILITY CONDITIONS

We consider six different scenarios in terms of the observations that are provided to the RL agent during learning. The full state consists of the triple (c_t, h_t, d_t). We consider the case where c_t is not directly observed and we instead provide the agent with either the most likely inferred context l_t as an input, and the case where c_t is not directly observed and we provide the agent with information about the inferred probability distribution over the context variable $\mathbf{p}_t$ as input. Specifically, since the distribution $\mathbf{p}_t$ is over a binary variable, we supply p_{0t} (the probability that the context is 0 as the feature. Further, we consider the case where the state variables h_t and d_t are both observed and the case where neither is observed. When h_t and d_t are not observed we augment the state with a time indicator variable i_t. In our experiments we use a time indicator variable $i_t = \text{mod}(t, k)$. This choice enables the agent to take different actions based on a cyclic notion of time within an episode. We experimented with different values of k and found little difference between different small values of k. We use $k = 2$ in our experiments.

In our experiments, the scenarios described above are labeled as follows: C-H-D: c_t, h_t, d_t observed. L-H-D: l_t, h_t, d_t observed. P-H-D: p_t, h_t, d_t observed. C-T: c_t, i_t observed. L-T: l_t, i_t observed. P-T: p_t, i_t observed.

We expect agents learned using the C-H-D observation set to perform the best as these agents have access to the full MDP state space. We hypothesize that as the feature noise increases, the P-H-D observation set will perform better than the L-H-D feature set as access to the context inference probability distribution provides the agent with strictly more information than the most likely context. Finally, we hypothesize a loss in performance in the scenarios where the habituation and disengagement variables can not be observed, which is a more realistic scenario as these variables can not be passively sensed and are problematic to obtain in practice even via direct self report.

3.3 REINFORCEMENT LEARNING AGENTS

In our experiments, we compare a policy gradient method to a value function method. For the value function method, we select the Dueling DQN method. We use a multilayer perceptron with two hidden layers, for both the state value and advantage functions. We perform a hyper-parameter search over hidden layers sizes [32, 64, 128, 256], batch sizes [16, 32, 64] Adam optimizer learning rates from 1e-6 to 1e-2, and epsilon greedy exploration rate decrements from 1e-6 to 1e-3. We report results using with 128 neurons on each hidden layer, Adam optimizer learning rate $lr = 5\text{e} - 4$, epsilon linear decrement $\delta_\epsilon = 0.001$, decaying ϵ from 1 to 0.01, batch size 64. The target Q network parameters are replaced every $K = 1000$ steps. The number of episodes used to learn the model is 1000.

For the REINFORCE policy network, we use a multilayer perceptron with one hidden layer. We perform hyper-parameter search over hidden layer sizes [32, 64, 128, 256], and Adam optimizer learning rates from 1e-6 to 1e-2. We report results using 128 neurons, and Adam optimizer learning rate $lr = 6\text{e} - 4$. We set the number of trajectory samples per gradient step to $M = 50$ and the number of episodes used for learning to 15,000.

4 EXPERIMENTS AND RESULTS

In this section we present experiments and results using the physical activity JITAI simulation domain and the reinforcement learning agents and scenarios introduced in the previous section. We repeat each experiment 3 times with different random seeds. All experiments use a reward discount rate of $\gamma = 0.99$. In all the experiments and for all random seeds, we first learn a policy and then compute the performance of the policy using the average over 1000 test episodes of the per-episode non-discounted total reward. We report the average performance over three seeds as well as the standard deviation of the performance over three seeds.

The Effect of Learning with Most Likely Contexts:

We begin by quantifying the impact of learning policies given the most likely context l_t instead of the true context c_t under the assumption that the habituation and disengagement variables are fully observed. In this experiment we vary the value of feature uncertainty parameter σ from 0 to 2 resulting in variation in context inference error from 0% to approximately 40%. As described in the previous section, we repeat this experiment three times for three random seeds for both DQN and REINFORCE and report performance in terms of average per-episode total reward. The results are shown as the orange lines in figures 2a and 2b for the DQN and REINFORCE agents. As we can see, the best performing policies are obtained when the context inference error rate is 0 so that $l_t = c_t$. As the context inference error rate increases, the performance of both the DQN and REINFORCE agents drops quickly. We can see that at a context inference rate of 40%, both agents experience a drop in reward due to using most likely contexts, of approximately 50% relative to using true contexts.

The Effect of Learning with Context Probabilities:

We next quantify the impact of learning policies given access to context inference probabilities $\mathbf{p}_t$ instead of the true context c_t under the assumption that the habituation and disengagement variables are fully observed. We contrast access to context inference probabilities with access only to most likely inferred contexts. We use the same experimental procedure as for the previous experiment. The results are shown as the blue lines in figures 2a and 2b for the DQN and REINFORCE agents. As expected, the best performing policies are again obtained when the feature uncertainty level is $\sigma = 0$ and the context inference error rate is 0 so that p_t effectively carries the same information as c_t. As the context inference error rate increases, the performance of both the DQN and REINFORCE agents using $\mathbf{p}_t$ again decreases.

However, as we can see from the figures, the performance of the agents with access to $\mathbf{p}_t$ clearly dominates the performance of agent with access to l_t until the context inference error rate approaches the maximum value considered. At a context inference error rate of 0.17, both agents using $\mathbf{p}_t$ achieve an increase of performance of more than 500 steps relative to the agents using only l_t.

To formally assess the differences between agents with access to $\mathbf{p}_t$ and l_t, we perform unpaired t-tests over the three repetitions for each context inference error rate. The results are shown in Table 4 A p-value < 0.05 indicates a statistically significant difference. The unpaired t-tests confirm that up to a context error rate of approximately 30%, access to $\mathbf{p}_t$ results in statistically significant improvements in total reward compared to access to l_t.

We provide more insight into the effect of access to context inference probabilities compared to most likely context inferences in Figure 3 The top row of plots shows the distribution of actions selected by REINFORCE when given access to context probabilities. The bottom row of plots shows the distribution of actions selected by REINFORCE when given access only to the inferred most likely context. Each plot in each row corresponds to the distribution of actions in a specific range of context inference probabilities. All results are for a context inference error rate of 17%.

As we can see, when given access to context inference probabilities, REINFORCE increasingly avoids taking the contextualized message actions 2 and 3 as the context uncertainty increases, instead preferring to take action 0. When the context inference uncertainty is low, it takes contextualized actions most of the time. By contrast, when given only the most likely inferred context as input, REINFORCE takes a larger proportion of actions 2 and 3 when the context is uncertain, resulting in a higher rate of disengagement events. Figure 1 in the supplemental material shows similar results for the DQN agent.

Finally, we further examine the effect of access to context inference probabilities compared to most likely context inferences as a function of the disengagement increment parameter ϵ_d and disengagement decay parameter δ_d. These results are presented in the supplemental material in Figure 2. These results show that context probabilities dominate most likely contexts over a wide range of disengagement dynamics. However, the performance difference tends to be larger in cases that lead to a greater chance of disengagement events occurring. This corresponds to larger values of the disengagement risk increment parameter value ϵ_d and smaller values of the disengagement risk decay parameter value δ_d.

The Effect of Partial Observability:

To study the effect of partial observability, we repeat the primary experiments presented in the previous two sections but under the scenario where the agents do not have access to the h_t and d_t state variables. Instead, the agents are given access to either the most likely context l_t and the time indicator variable i_t, or the context inference probability $\mathbf{p}_t$ and the time indicator variable i_t. We again vary the value of feature uncertainty parameter σ from 0 to 2 resulting in variation in context inference error from 0% to approximately 40%. The results when using the most likely context are given in Figure 2c The results when using context inference probabilities are given in Figure 2d.

First, we can see that the performance of the DQN method suffers drastically under partial observability. At a context inference error rate of 0, the DQN method achieves an average total reward of approximately 1500 under partial observability compared to an average total reward of 3000 with fully observed state. Further, regardless of whether most likely contexts

or context probabilities are used, the performance of the DQN agent decays similarly toward an average total reward of approximately 500 at a context inference error rate of approximately 40%.

We can see a significant contrast when comparing the DQN agent to the REINFORCE agent. The REINFORCE agent experiences a small drop in performance under the 0 context inference error condition compared to the same condition with fully observed state, thus vastly outperforming the DQN agent. Further, we can see that the REINFORCE agent maintains better performance when using context inference probabilities compared to when using most likely context under partial observability.

We again perform unpaired t-tests to formally contrast the DQN agent with the REINFORCE agent for each context inference error rate. The performance differences are highly statistically significant with large differences in mean performance across all context inference error rates. These results are presented in Table 1 in the supplemental material.

Sample Complexity of Learning:

In this experiment, we compare sample learning curves of the DQN and REINFORCE agents for scenarios C-H-D, P-H-D and P-T to illustrate their convergence properties as a function of the number of episodes of training. The results are shown in Figure 4 using a moving average window of 100 episodes. As expected, REINFORCE exhibits higher variability during learning and takes much longer to converge than the DQN agent. In general, policy gradient methods are known to be less sample efficient than value function methods, which can benefit from off-policy learning using a replay buffer. However, REINFORCE converges at a similar rate and to similar performance in both the P-H-D and P-T scenarios while the DQN method converges at a similar rate but to much worse performance under the P-T scenario.

5 CONCLUSIONS

In this paper we have investigated the impact of context inference error and partial observability on the ability to learn intervention option selection policies for Just-In-Time adaptive interventions using RL methods. We have introduced a novel simulation environments that captures key aspects of JITAIs including habituation and disengagement risk as well as uncertainty and error in context inferences. We have investigated learning policies which rely on most likely inferred context (as is typically the case in current JITAIs), and have shown that the use of context probabilities significantly outperforms the use of most likely context inferences. We have further shown that there is a stark difference in performance between policy gradient methods and Q-learning methods under partial observability.

As noted in Section 2.3 this work has a number of important limitations. First, our primary goal is to quantify the fundamental limits of policy learnability under context inference error and uncertainty as well as partial observability using policy gradient and Q-learning methods. In doing so we have not constrained the RL methods to a realistic number of

episodes during learning. As a result, our findings should be interpreted as providing upper bounds on performance in these important and previously unexplored settings.

Going forward, more work is required to compose the findings of this paper with regard to the use of probabilistic context inference representations with prior work such as Liao et al. [2020], which focuses on sample efficiency of learning. We also note that the drastic loss of performance experienced by traditional Q-learning methods in our experiments may be addressable using state augmentation methods such as the addition of memory or the use of recurrent neural networks that have been proposed in prior work to deal with partial observability. Another potentially interesting possibility is the incorporation of probabilistic dynamic latent variable models to provide beliefs over the full state including psychological latent variables.

Finally, we note that while the simulation environment was designed to model key issues with context uncertainty and delayed effect of actions, it is limited in other aspects. Nevertheless we believe that the insights we derive have important implications for the development of RL methods that can be applied to improve the effectiveness of real-world JITAIs.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgements

This work was supported by National Institutes of Health Office of Behavior and Social Sciences, National Cancer Institute, and National Institute of Biomedical Imaging and Bioengineering through grants U01CA229445 and 1P41EB028242.

A ADDITIONAL RESULTS

A.1 ACTION SELECTION ANALYSIS FOR DQN

Figure 5 shows the distriubtion of actions taken by the DQN agent. The top row of plots shows the distribution of actions selected by DQN when given access to context probabilities. The bottom row of plots shows the distribution of actions selected by DQN when given access only to the inferred most likely context. Each plot in each row corresponds to the distribution of actions in a specific range of context inference probabilities. All results are for a context inference error rate of 17%.

A.2 STATISTICAL SIGNIFICANCE OF PERFORMANCE DIFFERENCES UNDER PARTIAL OBSERVABILITY

We perform unpaired t-tests to formally contrast the DQN agent with the REINFORCE agent for each context inference error rate under the partial observability condition. The performance differences are highly statistically significant with large differences in mean performance across all context inference error rates. These results are presented in Table 5.

Table 5:

Unpaired t-tests on performance for scenarios REINFORCE L-T vs. DQN L-T, and scenarios REINFORCE P-T vs. DQN P-T, for different inferred error rates. Effect is the difference of the average returns.

REINF. vs. DQN	Error Rate	Effect	p-value
L-T	10%	1056.15	0.00002
L-T	17%	745.50	0.00009
L-T	27%	450.42	0.00005
L-T	31%	394.75	0.00139
L-T	41%	445.21	0.01435
P-T	10%	1367.03	0.00002
P-T	17%	1032.63	0.00347
P-T	27%	927.28	0.00008
P-T	31%	681.53	0.00054
P-T	41%	329.25	0.00014

A.3 PERFORMANCE AS A FUNCTION OF DISENGAGEMENT DYNAMICS PARAMETERS.

For both agents, we study how the performance of learned policies varies as a function of the disengagement increment parameter ϵ_d and disengagement decay parameter δ_d. The presented results correspond to $\sigma = 0.6$ and habituation and disengagement observed. The results are given in Figure 6. As we can see, these results show that the use of context inference probabilities improves on using most likely context inference over a wide range of settings of these variables. However, the performance difference tends to be larger in cases that lead to a greater chance of disengagement events occurring. This corresponds to larger values of the disengagement risk increment parameter value ϵ_d and smaller values of the disengagement risk decay parameter value δ_d.

References

Battalio Samuel L, Conroy David E, Dempsey Walter, Liao Peng, Menictas Marianne, Murphy Susan, Nahum-Shani Inbal, Qian Tianchen, Kumar Santosh, and Spring. Bonnie Sense2stop: a micro-randomized trial using wearable sensors to optimize a just-in-time-adaptive stress management intervention for smoking relapse prevention. Contemporary Clinical Trials, 109:106534, 2021. [PubMed: 34375749]

Bruin Tim de, Kober Jens, Tuyls Karl, and Babuška. Robert The importance of experience replay database composition in deep reinforcement learning. In Deep Reinforcement Learning Workshop, Advances in Neural Information Processing Systems, 2015.

Ertin Emre, Stohs Nathan, Kumar Santosh, Raij Andrew, Al'Absi Mustafa, and Shah. Siddharth Autosense: unobtrusively wearable sensor suite for inferring the onset, causality, and consequences of stress in the field. In Proceedings of the 9th ACM conference on embedded networked sensor systems, pages 274–287, 2011.

Gönül Suat, Namlı Tuncay, Coşar Ahmet, and Toroslu İsmail Hakkı. A reinforcement learning based algorithm for personalization of digital, just-in-time, adaptive interventions. Artificial Intelligence in Medicine, 115:102062, 2021. [PubMed: 34001322]

Hardeman Wendy, Houghton Julie, Lane Kathleen, Jones Andy, and Naughton Felix. A systematic review of just-in-time adaptive interventions (jitais) to promote physical activity. International Journal of Behavioral Nutrition and Physical Activity, 16(1):1–21, 2019. [PubMed: 30606197]

Hovsepian Karen, Mustafa Al'Absi Emre Ertin, Kamarck Thomas, Nakajima Motohiro, and Kumar Santosh. cstress: towards a gold standard for continuous stress assessment in the mobile environment. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*, pages 493–504, 2015.

Kwon Sunku, Wan Neng, Burns Ryan D, Brusseau Timothy A, Kim Youngwon, Kumar Santosh, Ertin Emre, Wetter David W, Lam Cho Y, Wen Ming, et al. The validity of motionsense hrv in estimating sedentary behavior and physical activity under free-living and simulated activity settings. Sensors, 21(4):1411, 2021. [PubMed: 33670507]

Liao Peng, Greenewald Kristjan, Klasnja Predrag, and Murphy Susan. Personalized heartsteps: A reinforcement learning algorithm for optimizing physical activity. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 4(1):1–22, 2020. [PubMed: 35846237]

Mnih Volodymyr, Kavukcuoglu Koray, Silver David, Graves Alex, Antonoglou Ioannis, Wierstra Daan, and Riedmiller Martin. Playing atari with deep reinforcement learning. In NIPS Deep Learning Workshop, 2013.

Nahum-Shani Inbal, Smith Shawna N, Spring Bonnie J, Collins Linda M, Witkiewitz Katie, Tewari Ambuj, and Murphy Susan A. Just-in-time adaptive interventions (jitais) in mobile health: key components and design principles for ongoing health behavior support. Annals of Behavioral Medicine, 52(6):446–462, 2018. [PubMed: 27663578]

Paredes Pablo, Ran Gilad-Bachrach Mary Czerwinski, Roseway Asta, Rowan Kael, and Hernandez Javier. Poptherapy: Coping with stress through pop-culture. In Proceedings of the 8th international conference on pervasive computing technologies for healthcare, pages 109–117, 2014.

Perski Olga, Emily T Hébert Felix Naughton, Eric B Hekler Jamie Brown, and Businelle. Michael S Technology-mediated just-in-time adaptive interventions (JITAIs) to reduce harmful substance use: a systematic review. Addiction, 117(5):1220–1241, 2022. [PubMed: 34514668]

Rabbi Mashfiqui, Min Hane Aung Mi Zhang, and Choudhury Tanzeem. Mybehavior: automatic personalized health feedback from user behaviors and preferences using smartphones. In Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing, pages 707–718, 2015.

Saleheen Nazir, Amin Ahsan Ali Syed Monowar Hossain, Sarker Hillol, Chatterjee Soujanya, Marlin Benjamin, Ertin Emre, Al'Absi Mustafa, and Kumar. Santosh puffmarker: a multi-sensor approach for pinpointing the timing of first lapse in smoking cessation. In Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pages 999–1010, 2015.

Schaul Tom, Quan John, Antonoglou Ioannis, and Silver David. Prioritized experience replay. In Proceedings of the International Conference on Learning Representations, 2016.

Spruijt-Metz Donna, Marlin Benjamin M, Pavel Misha, Rivera Daniel E, Hekler Eric, Torre Steven De La, Mistiri Mohamed El, Golaszweski Natalie M, Cynthia Li, Braganca Rebecca Braga De, et al. Advancing behavioral intervention and theory development for mobile health: the heartsteps ii protocol. International journal of environmental research and public health, 19(4):2267, 2022. [PubMed: 35206455]

Sutton Richard S. and Barto Andrew G.. Reinforcement Learning: An Introduction. MIT press, Cambridge, MA, 1998.

Sutton Richard S., David McAllester Satinder Singh, and Mansour Yishay. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems*, 1999.

Tewari Ambuj and Murphy. Susan A From ads to interventions: Contextual bandits in mobile health. Mobile Health: Sensors, Analytic Methods, and Applications, pages 495–517, 2017.

Wang Ziyu, Schaul Tom, Hessel Hado van Hasselt Matteo, Lanctot Marc, and Freitas. Nando de Dueling network architectures for deep reinforcement learning. In Proceedings of The 33rd International Conference on Machine Learning, 2016.

Williams Ronald J.. Simple statistical gradient-following algorithms for connectionist reinforcement learning. In *Machine Learning*, pages 8:229–256, 1992.

Yang Min-Jeong, Sutton Steven K, Hernandez Laura M, Jones Sarah R, Wetter David W, Kumar Santosh, and Vinci. Christine A just-in-time adaptive intervention (jitai) for smoking cessation: Feasibility and acceptability findings. Addictive Behaviors, 136:107467, 2023. [PubMed: 36037610]

Elad Yom-Tov Guy Feraru, Kozdoba Mark, Mannor Shie, Tennenholtz Moshe, and Hochberg Irit. Encouraging physical activity in patients with diabetes: intervention using a reinforcement learning system. Journal of medical Internet research, 19(10): e338, 2017. [PubMed: 29017988]

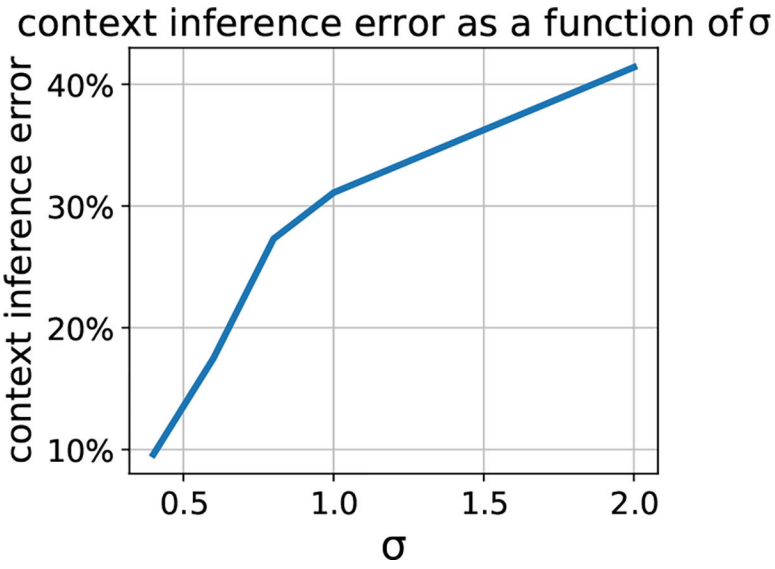

Figure 1:
Context inference error as a function of σ.

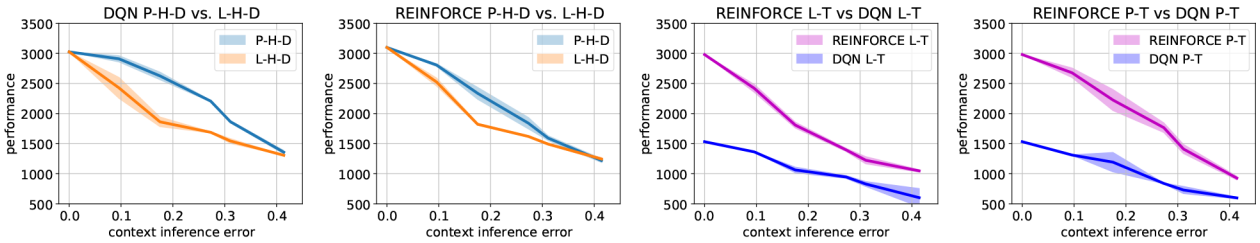

Figure 2:

(a) Effect of learning with most likely context and context probabilities for DQN. (b) Effect of learning with most likely context and context probabilities for REINFORCE. (c) Effect of learning with most likely contexts and partial observability for REINFORCE and DQN. (d) Effect of learning with context probabilities and partial observability for REINFORCE and DQN.

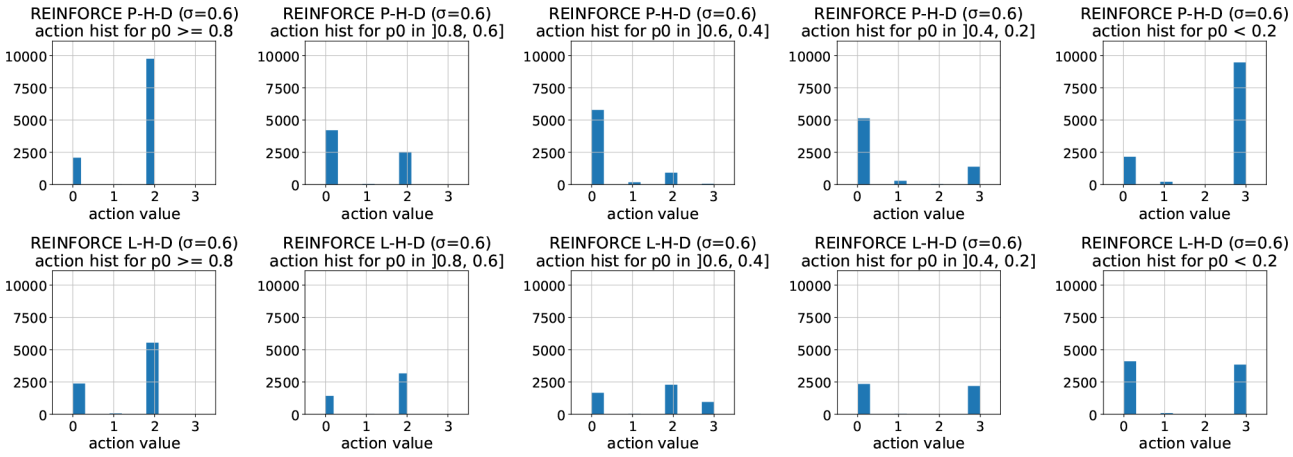

Figure 3:

The top row of plots shows the distribution of actions selected by REINFORCE when given access to context probabilities. The bottom row of plots shows the distribution of actions selected by REINFORCE when given access only to the inferred most likely context.

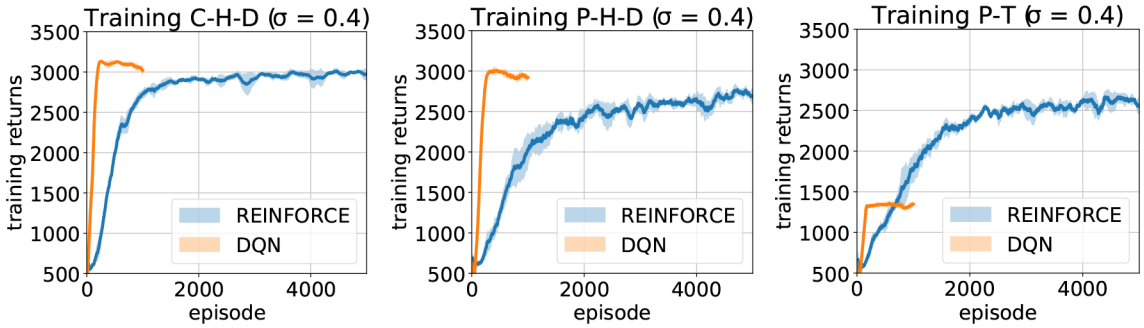

Figure 4:
Learning curves of DQN and REINFORCE.

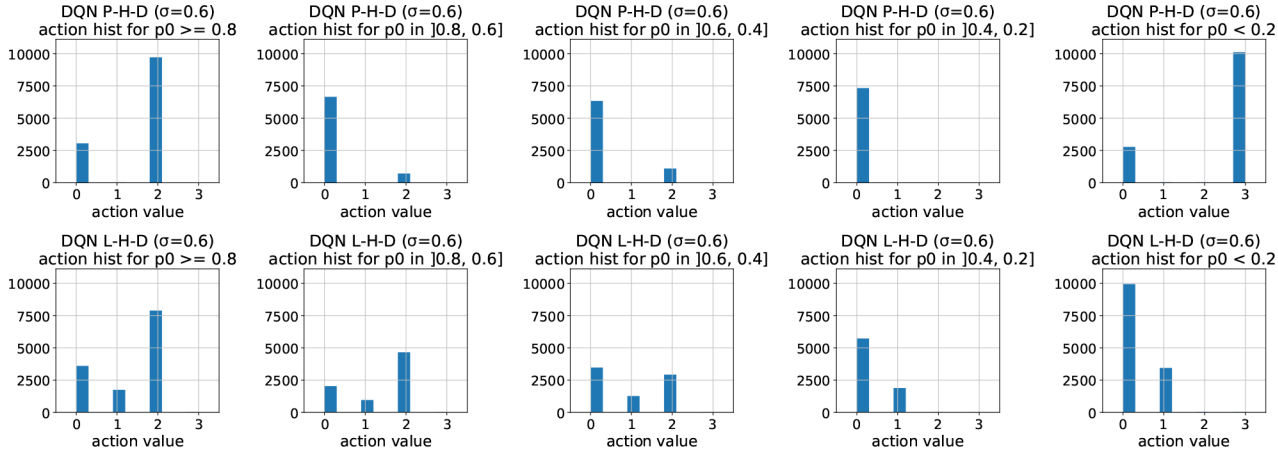

Figure 5:

The top row of plots shows the distribution of actions selected by DQN when given access to context probabilities. The bottom row of plots shows the distribution of actions selected by DQN when given access only to the inferred most likely context.

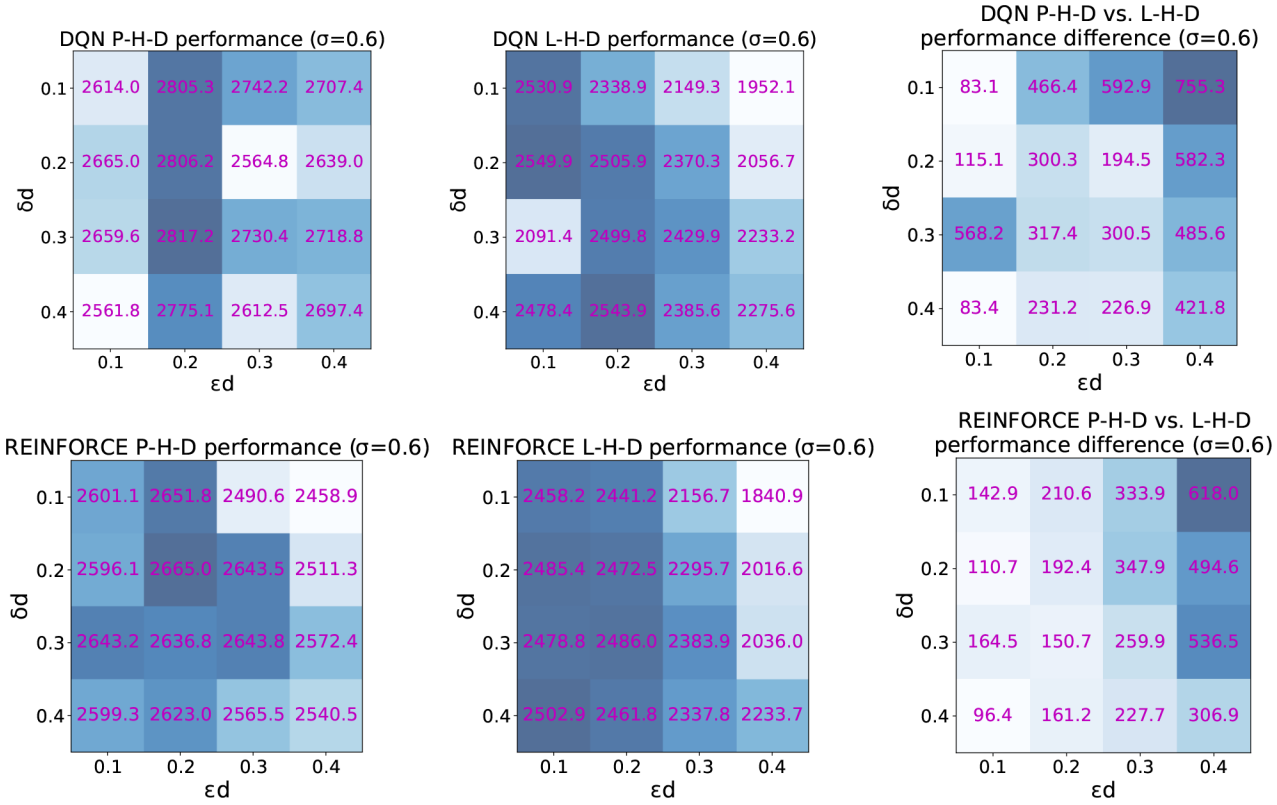

Figure 6:

Performance as a function of the disengagement increment ϵ_d and decay parameters δ_d, for DQN (top row) and REINFORCE (bottom row).

Table 1:

Actions Values

Action Value	Description
$a = 0$	do not send a message
$a = 1$	send a non tailored message
$a = 2$	send a message tailored to context 0
$a = 3$	send a message tailored to context 1

Table 2:

Simulation Variables

Variable	Description	Values
c_t	true context	{0,1}
$\mathbf{p}_t$	context probabilities	Δ^1
l_t	most likely context	{0,1}
d_t	disengagement risk level	[0,1]
h_t	habituation level	[0,1]
s_t	number of steps	$\mathbb{N}$

Table 3:

Environment Parameter Settings.

Parameter	Description	Value
δ_h	habituation decay	0.1
ϵ_h	habituation increment	0.05
δ_d	disengagement decay	0.1
ϵ_d	disengagement increment	0.4
ρ_1	$a_t = 1$ base reward	50.
ρ_2	$a_t = c_t + 2$ base reward	200.
σ	feature uncertainty	{0.4,..., 2}

Table 4:

Unpaired t-tests on performance for scenarios P-H-D vs. L-H-D, for different error rates, for both agents. Effect is the difference of the average returns.

P-H-D vs. L-H-D	Error Rate	Effect	p-value
DQN	10%	483.02	0.01930
DQN	17%	763.36	0.00043
DQN	27%	518.49	0.00000
DQN	31%	320.38	0.00045
DQN	41%	51.42	0.06041
REINFORCE	10%	282.42	0.00442
REINFORCE	17%	514.57	0.00183
REINFORCE	27%	220.28	0.03019
REINFORCE	31%	99.38	0.02693
REINFORCE	41%	−26.18	0.37660