Interaction-Grounded Learning

Tengyang Xie [1] **John Langford** [2] **Paul Mineiro** [2] **Ida Momennejad** [2]

Abstract

Consider a prosthetic arm, learning to adapt to its user's control signals. We propose *Interaction-Grounded Learning* for this novel setting, in which a learner's goal is to interact with the environment with no grounding or explicit reward to optimize its policies. Such a problem evades common RL solutions which require an explicit reward. The learning agent observes a multidimensional *context vector*, takes an *action*, and then observes a multidimensional *feedback vector*. This multidimensional feedback vector has *no* explicit reward information. In order to succeed, the algorithm must learn how to evaluate the feedback vector to discover a latent reward signal, with which it can ground its policies without supervision. We show that in an Interaction-Grounded Learning setting, with certain natural assumptions, a learner can discover the latent reward and ground its policy for successful interaction. We provide theoretical guarantees and a proof-of-concept empirical evaluation to demonstrate the effectiveness of our proposed approach.

1. Introduction

We consider a novel setting. A learner's goal is to interact with an environment, and while the environment reacts to the learner's actions, its feedback does not provide an explicit reward signal. Because the learner must deduce a grounding for the feedback solely via interaction, we call this setting *Interaction-Grounded Learning* (IGL).

There are *many* examples and potential applications of Interaction-Grounded Learning. In a visual domain, a robot could learn to interact effectively with a user's personalized hand gestures. In an audio domain, a smart speaker could learn to give useful responses based upon a user's idiosyncratic ways of signaling pleasure or annoyance. In a BCI (Brain Computer Interface) setting, a computer could learn to interact with a human based upon a user's EEG signals.

These problems are not easily solved using traditional reinforcement learning (RL), inverse RL, or supervised learning because the *absence of an explicit reward* is an essential ambiguity of the setting. When solving these problems, a key issue is agreeing on a shared code between a human and a computer. For example, in neurofeedback and BCI an algorithm is often trained via supervised learning techniques to interpret brain signals, which are in turn used to interact with or train human participants (Katyal et al., 2014; Mishra & Gazzaley, 2015; deBettencourt et al., 2015; Muñoz-Moldes & Cleeremans, 2020; Akinola et al., 2020; Chiang et al., 2021). The supervised learning techniques used, however, tend to be laborious and can require chronic retraining. Another challenge of BCI solutions is that, over time, the initial placement of sensors that read user signals, as well as the interpretation of the signals, often change and require re-calibration. IGL opens up the possibility of more natural and continual self-calibration.

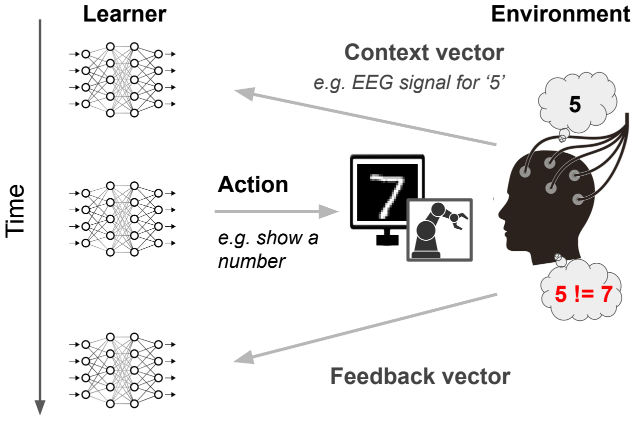
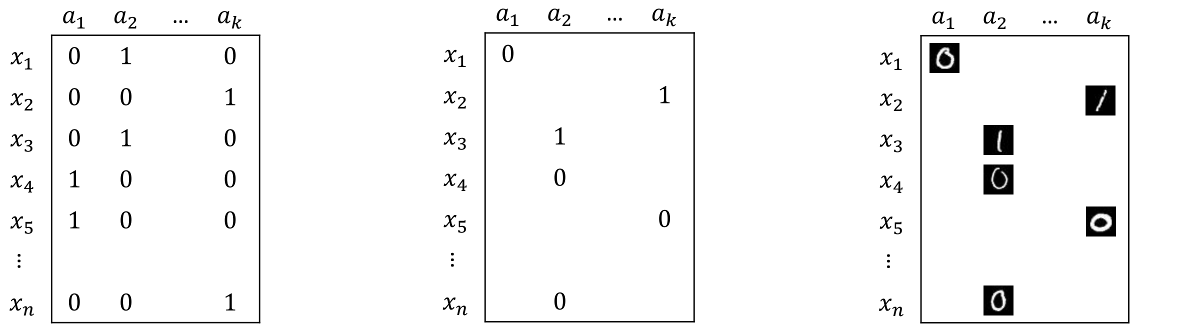


Figure 1. A schematic example of the Interaction-Grounded Learning (IGL) setting. The learner observes a context vector (e.g. EEG signal of interacting partner thinking about number 5 or intention to grab a cup), takes an action (e.g. show number 7 or move arm), and observes a feedback vector (e.g. an ungrounded multidimensional EEG signal). Importantly, there is *no explicit reward* signal. The learner assumes there is a latent reward in the feedback vector then learns a reward decoder and an optimal action policy given those assumptions.

[1] University of Illinois at Urbana-Champaign [2] Microsoft Research, New York City. Correspondence to: Tengyang Xie <tx10@illinois.edu>, John Langford <jcl@microsoft.com>, Paul Mineiro <pmineiro@microsoft.com>, Ida Momennejad <idamo@microsoft.com>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).



(a) Supervised Classification

(b) Contextual Bandits (CBs)

(c) Interaction-Grounded Learning (IGL)

Figure 2. An example of different learning approaches. **Figure 2(a):** Supervised learning assumes the full reward information is given for each context. **Figure 2(b):** Contextual bandits gives exact reward information on the selected action. **Figure 2(c):** In Interaction-Grounded Learning, a feedback vector is observed instead of a reward.

In the IGL setting, the learner observes a multidimensional *context vector*, takes an *action*, and then observes a multidimensional *feedback vector*. This feedback vector has *no* explicit reward information, but does carry information about a latent reward. In order to succeed, the learning algorithm must discover a good grounding for the feedback vector suitable for evaluating interactive policies. Interaction-Grounded Learning can then empower the agent to interpret multi-dimensional user signals in terms of latent reward, and optimize its behaviour using this inferred reward. While this may appear impossible at first, we prove that Interaction-Grounded Learning can succeed when three assumptions hold: (I) the feedback vector has information about the latent reward, (II) the feedback vector is conditionally independent of the action and context given the reward, and (III) random actions have low expected reward.

After introducing the Interaction-Grounded Learning setting in section 2 with more detail, we propose a potential algorithm to solve IGL: Explore-Exploit Ground learning, or E2G. The E2G learner takes random actions during gradually increasing exploration epoch. Later during intermittent exploitation, E2G grounds the reward in its interaction history during the exploration epochs. We prove that E2G can solve IGL under the assumptions mentioned above.

In section 8 we further discuss these assumptions, their applicability, and potential relaxation.

Our contribution

We define the Interaction-Grounded Learning setting in section 2. Given the scope of potential applications, we believe this setting may form a core area of study in the future.

Section 3 studies the feasibility of IGL in a simplest-possible batch setting, clarifying the assumptions under which it is tractable and providing a proof that it is indeed possible. The batch setting appears unnatural in most IGL applications where a more online approach is called for. Therefore, in section 4 we present E2G, an algorithm for online Interaction-Grounded Learning. We prove that E2G succeeds under similar assumptions to the simple batch setting. In section 5 we conduct proof-of-concept experiments showing IGL is possible in both the batch and online cases.

We then consider an alternatives to IGL, i.e., an unsupervised learning approach to extracting rewards. In section 6 we show a scenario, in which unsupervised learning cannot succeed without additional assumptions. The key insight is that the distribution of feedback vectors has multiple natural clusterings, which correspond to the solutions of different IGL problems. Restated, any unsupervised learning approach that can succeed on one instance of a problem must fail on another instance of the problem, while the IGL approaches we discuss here can succeed on both.

2. Problem Definition

We propose and analyze the *Interaction-Grounded Learning* setting, in which the learner uses interaction to create a grounding for evaluation and optimization of a feedback vector. Each round, the stationary environment generates an i.i.d. context $x \in \mathcal{X}$ from a distribution d_0 and reveals it to the learner, which chooses an action $a \in \mathcal{A}$ from a finite action set ($|\mathcal{A}| = K$); the environment then generates an unobserved binary reward $r \in \{0, 1\}$ and a feedback vector $y \in \mathcal{Y}$ conditional on (x, a), and reveals y to the learner. The reward can be either deterministic or stochastic, and we denote $R(x, a) := \mathbb{E}[r|x, a]$. In this setting, the spaces of both context $\mathcal{X}$ and feedback vector $\mathcal{Y}$ can be uncountably rich.

We use $\pi \in \Pi : \mathcal{X} \to \Delta(\mathcal{A})$ to denote a (stochastic) policy, and we define the expected return of policy π as $V(\pi) := \mathbb{E}_{(x,a)\sim d_0 \times \pi}[R(x, a)]$. Learning aims to achieve low regret with respect to the optimal policy in the policy class π,

$$\pi^\star := \operatorname*{argmax}_{\pi \in \Pi} V(\pi),$$

while interacting with the environment only through observations of context-action-feedback (x, a, y) triples.

The Interaction-Grounded Learning setting extends the border of interactive machine learning. Figure 2 is an example comparison of supervised learning, contextual bandits, and Interaction-Grounded Learning. In this case, an image of exact reward is provided as feedback vector y. This example does not exhaust the expressiveness of our generative model, which allows (r, y) to be drawn jointly conditional on (x, a). However, in Section 3 we impose further assumptions to make progress, and Figure 2 is consistent with those assumptions.

3. Learning with Conditional Independence

Direct empirical estimation of $V(\pi)$ requires access to unobserved information, frustrating the application of traditional techniques. Our algorithms instead employ a reward decoder class $\psi \in \Psi : \mathcal{Y} \to [0, 1]$. Treating $\psi(y)$ as an approximation to $\mathbb{E}[r|y]$ motivates the decoded reward $V(\pi, \psi) \doteq \mathbb{E}_{(x,a)\sim d_0 \times \pi}[\psi(y)]$. Our algorithms jointly choose ψ and π to maximize the decoded reward. The main challenge of learning the best π and ψ jointly over Π and Ψ is that the reward r is unobserved. Under what conditions does maximizing the decoded reward ensure low regret with respect to the unobserved latent reward? We leverage the following assumption to enable success.

Assumption 1 (Conditional Independence)**.** *For arbitrary (x, a, r, y) tuple where r and y are generated based on a context x and action a, we assume the feedback vector y is conditionally independent of action a and context x given latent reward r. In other words we assume that $x, a \perp\!\!\!\perp y | r$.*

Assumption 1 ensures that the feedback vector y is generated only based on the latent reward r, without further dependence on the action a or the context x. As discussed in section 8, this assumption is reasonable for some problems. While the assumption may seem unreasonable for others, it could be satisfied by existing orthogonalization practices in BCI (which are applied prior to applying machine learning) and could conceivably be relaxed in future work. Informally this assumption enables progress on the learning objective by ensuring that the mistakes of a reward predictor have a uniform effect across the policy class. This allows the decoded latent reward to be a faithful representation of the expected return of a policy. We carefully elaborate this argument below.

3.1. Proxy Learning Objective

Our target is to find a π that maximizes $V(\pi)$ and we know $V(\pi, \psi)$ can be viewed as an estimation of $V(\pi)$ using ψ so $V(\pi, \psi)$ is a natural simple objective. However, if we consider maximizing $V(\pi, \psi)$ directly over Π and Ψ, two difficulties could arise:

(i) ψ converges to the trivial wrong solution of $\psi(y) \to 1, \forall y \in \mathcal{Y}$, when maximizing $V(\pi, \psi)$ directly.

(ii) $V(\pi, \psi)$ does not necessarily correspond to the true value of $V(\pi)$. For example, if ψ always decodes the feedback vector opposite to the truth, then $V(\pi, \psi)$ decreases when $V(\pi)$ increases.

To address these difficulties, we propose to maximize the estimated value difference from a policy π_{bad} known to have low expected return. For example, a policy which chooses actions uniformly at random has an expected accuracy of $1/K$ on classification problems. Via this policy we define the learning objective $\mathcal{L}(\pi, \psi)$ and optimal policy-decoder pair as

$$\operatorname*{argmax}_{(\pi,\psi)\in \Pi\times\Psi} \mathcal{L}(\pi, \psi) := V(\pi, \psi) - V(\pi_{\mathsf{bad}}, \psi). \tag{1}$$

Expanding the objective for a fixed (π, ψ) pair reveals

$$\begin{aligned}
& V(\pi, \psi) - V(\pi_{\mathsf{bad}}, \psi) \\
&= \mathbb{E}_{(x,a)\sim d_0\times\pi}[\psi(y)] - \mathbb{E}_{(x,a)\sim d_0\times\pi_{\mathsf{bad}}}[\psi(y)] \\
&= \mathbb{E}_{(x,a)\sim d_0\times\pi}[\psi(y)\mathbb{1}(r=1) + \psi(y)\mathbb{1}(r=0)] \\
&\quad - \mathbb{E}_{(x,a)\sim d_0\times\pi_{\mathsf{bad}}}[\psi(y)\mathbb{1}(r=1) + \psi(y)\mathbb{1}(r=0)] \\
&\overset{\text{(a)}}{=} V(\pi)\mathbb{E}[\psi(y)|r=1] + (1 - V(\pi))\mathbb{E}[\psi(y)|r=0] \\
&\quad - V(\pi_{\mathsf{bad}})\mathbb{E}[\psi(y)|r=1] \\
&\quad - (1 - V(\pi_{\mathsf{bad}}))\mathbb{E}[\psi(y)|r=0] \\
&= (V(\pi) - V(\pi_{\mathsf{bad}}))\left(\mathbb{E}[\psi(y)|r=1] - \mathbb{E}[\psi(y)|r=0]\right) \\
&\doteq (V(\pi) - V(\pi_{\mathsf{bad}}))\,\Delta\psi,
\end{aligned} \tag{2}$$

where (a) leverages the conditional independence property in Assumption 1.

Equation (2) reveals our learning objective is a linear transformation of the unobservable quantity of interest, with intercept $V(\pi_{\mathsf{bad}})$ and slope $\Delta\psi \doteq (\mathbb{E}[\psi(y)|r=1] - \mathbb{E}[\psi(y)|r=0])$. Importantly, the slope $\Delta\psi$ is independent of π, implying a single reward predictor induces the correct ordering over all policies whose value exceeds that of π_{bad}. However, policies which are worse than π_{bad} can be ordered incorrectly, not just amongst themselves but also relative to policies that are insufficiently better than π_{bad}. We address this next, leading to the second assumption.

Quality of the Reward Decoder Using exact expectations, any reward decoder with $\Delta\psi > 0$ induces a correct ordering over policies whose value exceeds that of

π_{bad}. With finite sample approximations, however, a small value of $\Delta\psi$ makes this harder, resulting in increased sample complexity. Therefore we use the slope $\Delta\psi \doteq (\mathbb{E}[\psi(y)|r=1] - \mathbb{E}[\psi(y)|r=0])$ to measure the quality of the reward decoder, and we define the optimal reward decoder $\psi^\star$ via

$$\psi^\star := \operatorname*{argmax}_{\psi\in\Psi} \Delta\psi.$$

Identifiability Since $(V(\pi) - V(\pi_{\mathsf{bad}}))$ and $\Delta\psi$ are not always positive, there are potentially two extrema of equation (1), one corresponding to the best policy ($V(\pi) > V(\pi_{\mathsf{bad}})$) coupled with the optimal reward decoder ($\Delta\psi > 0$), and one corresponding to the worst policy ($V(\pi) < V(\pi_{\mathsf{bad}})$) coupled with the worst reward decoder ($\Delta\psi < 0$). To ensure the desired extrema is highest value, we make the following assumption.

Assumption 2 (identifiability)**.** *There exists a constant $\eta > 0$ such that $\pi^\star$ and $\psi^\star$ satisfy*

$$(V(\pi^\star) - V(\pi_{\mathsf{bad}}))\,\Delta\psi^\star \geq V(\pi_{\mathsf{bad}}) + \eta.$$

Remark 1. *Assumption 2 assumes that the "incorrect" optimization direction always achieves less $\mathcal{L}(\pi,\psi)$ value than the "correct" direction, which corresponds to a random action being wrong more often than not. This can be demonstrated as follows. Let $\pi^\dagger := \operatorname{argmin}_{\pi\in\Pi} V(\pi)$ and $\psi^\dagger := \operatorname{argmin}_{\psi\in\Psi} \Delta\psi$. Since $\min_{\pi\in\Pi} V(\pi) \geq 0$ and $\min_{\psi\in\Psi} \Delta\psi \geq -1$, we have $V(\pi^\dagger,\psi^\dagger) - V(\pi_{\mathsf{bad}},\psi^\dagger) = (V(\pi^\dagger) - V(\pi_{\mathsf{bad}}))\Delta\psi^\dagger \leq V(\pi_{\mathsf{bad}})$. Thus, Assumption 2 ensures $(\pi^\star,\psi^\star)$ to be the only global optima of objective Eq.(1), and the non-zero gap η allows learning with finite samples to occur.*

The requirement from Assumption 2 can be viewed as: π_{bad} must be "sufficiently bad". For example, a policy π_{bad} that chooses actions uniformly at random applied on a classification task becomes increasingly bad as K increases, but is never sufficiently bad when $K = 2$ because $V(\pi_{\mathsf{bad}}) \geq V(\pi^*)/K$ and $\Delta\psi^* \leq 1$.

3.2. Sample Complexity

We now provide the finite-sample results for batch-style optimization of the objective in equation (1) with empirical data $\mathcal{D}$. Let $\mathcal{D}$ consists of n i.i.d. (x,a,y) samples, where (x,a) is generated from distribution $d(\cdot,\cdot)$. We also define $\widehat{V}_{\mathcal{D}}(\pi,\psi) := \frac{1}{n}\sum_{i=1}^{n} \frac{\pi(a_i|x_i)}{d(a_i|x_i)}\psi(y_i)$ to be the estimated $V(\pi,\psi)$ using $\mathcal{D}$.

Theorem 1. *Let $(\widehat{\pi},\widehat{\psi}) := \operatorname{argmax}_{(\pi,\psi)\in\Pi\times\Psi} \widehat{V}_{\mathcal{D}}(\pi,\psi) - \widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}},\psi)$ and*

$$\varepsilon_{\mathsf{stat},n} := \sqrt{\frac{4\max_{\pi\in\Pi}\left\|\frac{\pi(a|x)}{d(a|x)}\right\|_{2,d}\log\frac{2|\Pi||\Psi|}{\delta}}{n}} + \frac{\max_{(x,a)\in\mathcal{X}\times\mathcal{A}}\frac{1}{d(a|x)}\log\frac{2|\Pi||\Psi|}{\delta}}{3n}.$$

Under Assumption 1 and 2, if n is sufficiently large such that $\varepsilon_{\mathsf{stat},n} \leq \eta/2$ where η is defined in Assumption 2, then with probability $1-\delta$:

$$V(\pi^\star) - V(\widehat{\pi}) \leq \frac{2\varepsilon_{\mathsf{stat},n}}{\Delta\psi^\star}$$
$$\Delta\psi^\star - \Delta\widehat{\psi} \leq \frac{2\varepsilon_{\mathsf{stat},n}}{V(\pi^\star) - V(\pi_{\mathsf{bad}})}.$$

Proof Sketch. We show that $\varepsilon_{\mathsf{stat},n} \leq \eta/2$ is a sufficient condition of $\Delta\widehat{\psi} > 0$ in Lemma 7 (in Appendix A). When we have $\Delta\widehat{\psi} > 0$, $\widehat{\psi}$ will not decode the opposite reward. Combining the facts of $(\widehat{\pi},\widehat{\psi}) = \operatorname{argmax}_{(\pi,\psi)\in\Pi\times\Psi} \widehat{V}_{\mathcal{D}}(\pi,\psi) - \widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}},\psi)$, $(\pi^\star,\psi^\star) = \operatorname{argmax}_{(\pi,\psi)\in\Pi\times\Psi} V(\pi,\psi) - V(\pi_{\mathsf{bad}},\psi)$, and Eq.(2) yield the bound on $V(\widehat{\pi})$ and $\Delta\widehat{\psi}$. The detailed proof of Theorem 1 is provided in Appendix A. □

Remark 2. *As we show in Theorem 1 and its proof, maximizing $\widehat{V}_{\mathcal{D}}(\pi,\psi) - \widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}},\psi)$ converges to the right direction only after we have sufficient data. This reflects the difficulty (ii) discussed at the beginning of Section 3.1 and Remark 1. The condition of $\varepsilon_{\mathsf{stat},n} \leq \eta/2$ provides a concrete sample complexity requirement to guarantee identifiability of $(\pi^\star,\psi^\star)$, according to Assumption 2.*

4. Interactive Algorithms

We now present an interactive algorithm for IGL. Similar to the epoch-greedy algorithm (Langford & Zhang, 2008) for contextual bandits, Algorithm 1 interleaves exploration and exploitation. A policy that chooses actions uniformly at random is used both for exploration and as π_{bad} (in line with Assumption 2). Throughout this section, we use μ to denote the distribution of $(x,a) \sim d_0 \times \pi_{\mathsf{bad}}$, which is the data distribution of exploration data $\mathcal{D}_i$ in Algorithm 1 at any time step i.

The key difference is the objective on line 4, which seeks the ψ,π pair that most distinguishes from uniform random action values according to ψ to form an unbiased estimate of the objective in Eq.(1).

Choice of $\{n_i\}_{i=1}^{\infty}$ As we showed in Theorem 1, some amount of "warm-up" data is needed in order to guarantee the optimization is on the correct direction (i.e., $\Delta\psi > 0$ for

Algorithm 1 E2G

```
Input: Exploration samples $\mathcal{D}_0 = \{\}$, $t = 0$, scheduling parameters $\{n_i\}_{i=1}^{\infty}$.
1: for $i = 1, 2, \ldots$ do
2:     Select an action uniformly at random and collect $\{(x_t, a_t, y_t)\}$.
3:     $\mathcal{D}_i = \mathcal{D}_{i-1} \cup \{(x_t, a_t, y_t)\}$.
4:     Compute $(\pi_i, \psi_i)$ by solving
           $\operatorname*{argmax}_{(\pi,\psi)\in\Pi\times\Psi} \mathop{\mathbb{E}}_{(x,a,y)\sim\mathcal{D}_i}[K\psi(y)\pi(a|x)] - \mathop{\mathbb{E}}_{(x,a,y)\sim\mathcal{D}_i}[\psi(y)]$.
5:     Execute $\pi_i$ for $n_i$ steps (i.e., select $a_{t'} \sim \pi_i(\cdot|x_{t'})$ for $t' = t+1, t+2, \ldots, t+n_i+1$), and set $t = t+n_i+1$.
6: end for
```

the learned ψ). Thus, the exploitation scheduling $\{n_i\}_{i=1}^{\infty}$ is chosen in the following way:

(1) If the amount of exploration data at time step i is not sufficient to guarantee $\Delta\psi_i > 0$, then explore.

(2) If we have enough exploration data to ensure $\Delta\psi_i > 0$, then explore/exploit scheduling similar to the epoch-greedy algorithm (Langford & Zhang, 2008) is used.

In the analysis of E2G, we provide the detailed definition of $\{n_i\}_{i=1}^{\infty}$, along with a discussion about when we have sufficient exploration data to guarantee $\Delta\psi_i > 0$.

4.1. Analysis of E2G

We now provide the analysis of Algorithm 1 in this section, as well as the definition and discussion of exploitation scheduling $\{n_i\}_{i=1}^{\infty}$. Over our analysis, we use ι to denote the complexity term in a sample complexity bound for standard supervised learning, which can be naively formed as $\log\left(\frac{2T|\Psi||\Pi|}{\delta}\right)$ or using some more advanced method such as covering number or Rademacher complexity (see e.g., (Mohri et al., 2018)).

We study the regret of Algorithm 1 which is defined as follows,

$$\mathrm{Regret}(T) := TV(\pi^\star) - \mathbb{E}\left[\sum_{t=1}^{T} R(x_t, a_t)\right].$$

The following theorem describes the regret guarantee of Algorithm 1, along with the precise definition of $\{n_i\}_{i=1}^{\infty}$.

Theorem 2. *If we choose $\{n_i\}_{i=1}^{\infty}$ as*

$$n_i = \begin{cases} 0, & i \leq {2K^2}/{\eta^2}; \\ \lfloor\sqrt{{i}/{K\iota}}\rfloor, & i > {2K^2}/{\eta^2}, \end{cases}$$

then with probability at least $1-\delta$, the regret of Algorithm 1 is bounded by

$$\mathrm{Regret}(T) = \widetilde{\mathcal{O}}\left(\frac{K^{1/3}T^{2/3}}{\Delta\psi_{\pi^\star}} + \frac{2K^2}{\eta^2}\right).$$

Remark 3. *Although the length of initial pure exploration stage is defined using the constant η (defined in Assumption 2), there is also a data-driven way to determine it in practice, if we can upper bound the uniform policy's performance, $V(\pi_{\mathsf{bad}})$. That is, if $\widehat{V}_{\mathcal{D}_i}(\pi_i, \psi_i) - \widehat{V}_{\mathcal{D}_i}(\pi_{\mathsf{bad}}, \psi_i) > V(\pi_{\mathsf{bad}}) + K\varepsilon_{\mathcal{D}_i}$ holds at time step i, where $K\varepsilon_{\mathcal{D}_i}$ denotes the statistical error at time step i, then we must have $\Delta\psi_i > 0$. Therefore, we can use the scheduling of $n_i = \lfloor\sqrt{{i}/{K\iota}}\rfloor$ for all the subsequent times steps. The detailed theoretical basis for this is presented in Lemma 4.*

We defer the detailed proof Theorem 2 to Appendix A. The following proof sketch describes the main technical components for proving Theorem 2.

Proof Sketch. We use $\widehat{V}_{\mathcal{D}}(\pi, \psi)$ and $\widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}}, \psi)$ to denote the empirical estimations of $V(\pi, \psi)$ and $V(\pi_{\mathsf{bad}}, \psi)$ in Algorithm 1 respectively, where

$$\widehat{V}_{\mathcal{D}}(\pi, \psi) := \mathop{\mathbb{E}}_{(x,a,y)\sim\mathcal{D}}[K\psi(y)\pi(a|s)]$$
$$\widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}}, \psi) := \mathop{\mathbb{E}}_{(x,a,y)\sim\mathcal{D}}[\psi(y)].$$

Then step 4 in Algorithm 1 can we rewritten as

$$\operatorname*{argmax}_{\pi\in\Pi} \max_{\psi\in\Psi} \quad \widehat{V}_{\mathcal{D}}(\pi, \psi) - \widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}}, \psi_\pi),$$

where $\mathcal{D}$ denotes $\mathcal{D}_i$ for specific time step i.

For simplicity, we also define $\widehat{\psi}_{\mathcal{D}}$ and $\widehat{\pi}_{\mathcal{D}}$ to be the learned policy and reward decoder given exploration data $\mathcal{D}$ (for specific time step i, also just set $\mathcal{D} = \mathcal{D}_i$),

$$(\widehat{\pi}_{\mathcal{D}}, \widehat{\psi}_{\mathcal{D}}) := \operatorname*{argmax}_{(\pi,\psi)\in\pi\times\Psi} \widehat{V}_{\mathcal{D}}(\pi, \psi) - \widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}}, \psi_\pi). \tag{3}$$

Over this section, we define $\varepsilon_{\mathcal{D}}$ as $\varepsilon_{\mathcal{D}} := \sqrt{{\iota}/{2|\mathcal{D}|}}$, and we have $|(\widehat{V}_{\mathcal{D}}(\pi, \psi) - \widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}}, \psi)) - (V(\pi, \psi) - V(\pi_{\mathsf{bad}}, \psi))| \leq K\varepsilon_{\mathcal{D}}$ for any $(\pi, \psi) \in \Pi \times \Psi$ by simply applying the standard concentration inequality.

The difficulty of identifiability discussed for the batch mode (see Remark 2), implies that Algorithm 1 should not start exploitation until it gathers enough exploration data so that better than random performance can be guaranteed after learning from the exploration data. We formalize this fact with the following lemmas.

Lemma 3. *Let $\mathcal{D}$ be the exploration data at arbitrary time step. If π satisfies $V(\pi) \leq V(\pi_{\mathsf{bad}})$, then, with probability at least $1-\delta$, we have for any $\psi \in \Psi$,*

$$\widehat{V}_{\mathcal{D}}(\pi, \psi) - \widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}}, \psi) \leq V(\pi_{\mathsf{bad}}) + K\varepsilon_{\mathcal{D}}.$$

In Lemma 3, we upper bound the value of the objective function if we enter the "opposite" optimization direction — $\psi \to \operatorname{argmin}_{\psi \in \Psi} \Delta\psi < 0$. The proof of Lemma 3 follows a similar argument as Remark 1 and is deferred to Appendix A. By using that result, the next lemma shows that if the amount of exploration data is large enough for $\widehat{V}_{\mathcal{D}}(\widehat{\pi}_{\mathcal{D}}, \widehat{\psi}_{\mathcal{D}}) - \widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}}, \widehat{\psi}_{\mathcal{D}}) > V(\pi_{\mathsf{bad}}) + K\varepsilon_{\mathcal{D}}$, then we must have $\Delta\widehat{\psi}_{\mathcal{D}} > 0$, which yields the bound on $V(\widehat{\pi}_{\mathcal{D}})$.

Lemma 4. *Let $\mathcal{D}$ be the exploration data at some time step with $(\widehat{\pi}_{\mathcal{D}}, \widehat{\psi}_{\mathcal{D}})$ defined as in Eq.(3). Then, if*

$$\widehat{V}_{\mathcal{D}}(\widehat{\pi}_{\mathcal{D}}, \widehat{\psi}_{\mathcal{D}}) - \widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}}, \widehat{\psi}_{\mathcal{D}}) > V(\pi_{\mathsf{bad}}) + K\varepsilon_{\mathcal{D}}, \quad (4)$$

then we have,

$$V(\widehat{\pi}_{\mathcal{D}}) \geq V(\pi^\star) - \frac{2K\varepsilon_{\mathcal{D}}}{\Delta\psi^\star}.$$

The next lemma provides a bound on the amount of initial exploration data that we need to ensure Eq.(4).

Lemma 5. *Let $T_0 = 2K^2\iota/\eta^2$ where η is defined in Assumption 2, then with probability at least $1-\delta$, we have*

$$\widehat{V}_{\mathcal{D}}(\widehat{\pi}_{\mathcal{D}}, \widehat{\psi}_{\mathcal{D}}) - \widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}}, \widehat{\psi}_{\mathcal{D}}) > V(\pi_{\mathsf{bad}}) + K\varepsilon_{\mathcal{D}},$$

where $\mathcal{D} = \mathcal{D}_{t > T_0}$.

Combining these lemmas together with a similar argument as in (Langford & Zhang, 2008), we establish a proof for Theorem 2. The detailed proof of all the lemmas above and Theorem 2 can be found in Appendix A. □

5. Experiments

In this section, we provide empirical evaluations in simulated environments. We experiment with both batch and online IGL.

The task is as depicted in Figure 2. We evaluated our approach by comparing

(1) SUP — Supervised classification, as in Figure 2(a);

(2) CB — Contextual bandits with exact reward, as in Figure 2(b);

(3) IGL — Our proposed approach using Eq.(1) for batch mode learning and Algorithm 1 for the online setting, with an image feedback vector as in Figure 2(c).

Note that supervised learning (SUP) should be better than contextual bandit learning (CB), which in turn should do better than Interaction-Grounded Learning (IGL) since each step in that sequence makes the problem more difficult.

Supervised classification uses logistic regression with a linear representation and cross-entropy loss. The other methods use the same representation with softmax policies. During testing time, each algorithm takes the argmax of the policy. We provide the details on setting up the experiments in Appendix B, where we also discuss the practical difficulty of jointly optimizing π and ψ created by the multiple extrema of equation (1) and propose some mitigation strategies.

Experimental Results

We evaluated our approach on the MNIST environment based on the infinite MNIST simulator (Loosli et al., 2007). At each time step, the context x_t is generated uniformly from the infinite MNIST simulator. After that, the learner selects action $a_t \in \{0, ..., 9\}$ as the predicted label of x_t. The binary reward r is the correctness of the prediction label a_t. The feedback vector y_t is also generated from the infinite MNIST simulator, either an image of a one digit or an image of a zero digit depending upon r.

Over our experiments in the batch mode, we use the uniform policy π_{bad} to gather data, and the number of examples is 60000. Our results are averaged over 16 random trials.

Setting	Policy Accuracy (%)
SUP	90.62 ± 1.02
CB	85.58 ± 4.50
IGL	82.21 ± 4.33

Table 1. Results in Batch Mode of MNIST Environment.

Table 1 is the result in the batch mode of the MNIST environment. All the experiments are repeated 16 times with the average and standard deviation reported. For our IGL algorithm, we optimize the objective in line 4 of E2G on the batch data. IGL achieves comparable accuracy as CB despite the handicap of only observing feedback vectors.

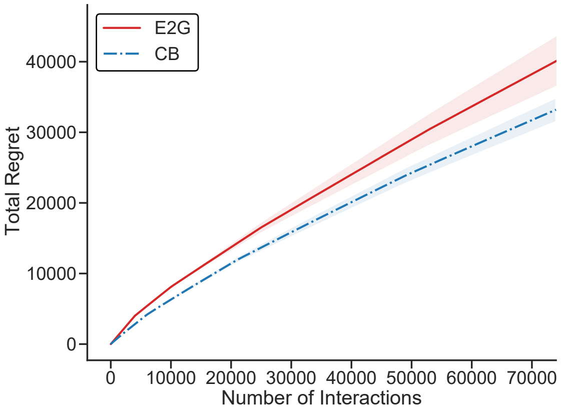


Figure 3. Comparison of E2G and competitors on the MNIST environment.

We also compare E2G with a CB algorithm in the online setting with the results shown in Figure 3 (averaged over 16 runs). In this case, E2G starts with 4000 exploration events based on the suggestion of Theorem 2. This result demonstrates the effectiveness of the online use of E2G.

6. Failure of Unsupervised Learning in IGL

Unsupervised learning provides another approach to Interaction-Grounded Learning. If unsupervised learning distinguishes the feedback vectors generated from different rewards, then learning the optimal policy for IGL is still possible. Indeed, the task from section 5 can be solved by clustering the feedback vectors. However, in this section, we show the information-theoretical hardness of using unsupervised learning in IGL in general.

To formalize our argument, suppose the agent picks an unsupervised-learning oracle and a contextual-bandits oracle. The reward is decoded from feedback vectors using the unsupervised-learning oracle, and the contextual-bandits oracle learns a policy using the decoded reward. Our result focuses on the ambiguity of unsupervised-learning-based approaches. To make it cleaner, we consider the scenario of (1) data is infinite, (2) the contextual-bandits oracle could always output the exact optimal policy. Note that, this result could be extended to the general case by capturing the statistical error and approximation error properly. The following theorem formally present our lower bound result.

Theorem 6. *There exists a set of IGL tasks and a bad behavior policy π_{bad}, such that: (i) each IGL task and π_{bad} satisfy Assumption 1 and Assumption 2, the data for each IGL task is infinite, and data is gathered by running π_{bad} (ii) the contextual bandit oracle outputs the exact optimal policy corresponding to the input reward, then for any unsupervised learning oracle called by the unsupervised-learning-based approach, there exists at least one IGL task in that class,such that the performance of output policy, $\widehat{\pi}$, is $V(\widehat{\pi}) = \Omega(1)$.*

Proof of Theorem 6. We construct the following 10 environments based on the MNIST environment introduced in Section 5. Each of theses 10 environments have the same context-action-reward setting as described in Section 5 while differing in the feedback vector generation process. The environment i ($i = 0, 1, \ldots, 9$) generates the feedback vector in the following way:

$$\begin{aligned} y|_{r=0} &= \text{a random image with label } \{\text{“0”}, \ldots, \text{“9”}\} \backslash \text{“}i\text{”}, \\ & \text{where the labels are also distributed uniformly,} \\ y|_{r=1} &= \text{a random image with label “}i\text{”}. \end{aligned}$$

If the data gathering policy, π_{bad}, is the uniform policy over 10 actions, then the distribution of y is *identical* for all 10 environments above. It implies that any unsupervised-learning-based approach is not able to distinguish among these 10 environments no matter which unsupervised-learning oracle it calls, and therefore it would decode the same reward for all these 10 environments.

We use $\psi(\cdot)$ to denote the learned reward decoder, and let $r_i := \mathbb{E}[\psi(y)|\text{image with label “}i\text{”}], \forall i \in \{0, 1, \ldots, 9\}$, where $r_i \in [0, 1]$ are the expected decoded reward for images with label “i”.

Without loss of generality, let $\operatorname{argmin}_{i \in \{0,1,\ldots,9\}} r_i = r_0$. Since the contextual bandits oracle output the exact optimal policy with the corresponding decoded reward, we can obtain that the output policy for the environment 0, $\widehat{\pi}_0$, has $V(\widehat{\pi}_0) = 0$. That is because $\mathbb{E}[\psi(y)|r = 1] = r_0 < \mathbb{E}[\psi(y)|r = 0] = \frac{1}{9}(r_1 + r_2 + \ldots + r_9)$. This completes the proof. □

7. Related Work

The problem of partial monitoring (Mertens, 1990; Piccolboni & Schindelhauer, 2001; Mannor & Shimkin, 2003; Cesa-Bianchi et al., 2006; Bartók et al., 2014; Lattimore & Szepesvári, 2019; 2020) also provides a framework for the decision-making problems with imperfect feedback. Most of the papers on the partial monitoring problem consider the case where the feedback is a known function of the actual cost, resulting in algorithms that compose the known function with more standard online learning strategies. Notable exceptions are Hanawal et al. (2017); Verma et al. (2019; 2020) which study an unsupervised sequential bandit setting where feedback information does not directly identify the underlying arm reward. They identify a condition where pairwise disagreement among the binary components of the feedback vector is sufficient to order the arms correctly. The IGL setting includes these prior works as special cases.

Another related setting is latent (contextual) bandits (Maillard & Mannor, 2014; Zhou & Brunskill, 2016; Hong et al., 2020). In the problem of latent (contextual) bandits, the reward is not only observed, but is drawn from a known distribution conditioned on a latent state. The primary goal of this problem is to identify that latent state, such that acting optimally is straightforward, and learning proceeds more rapidly than naive utilization of the observed reward. IGL distinguishes from this setting as the reward is still observed in the latent (contextual) bandits, whereas IGL must infer the latent reward from interaction.

Inverse reinforcement learning (Ng et al., 2000) learns a reward which explains the behavior of expert demonstrations for the purpose of learning a control policy. In the IGL setting, there are no expert demonstrations; instead there is joint learning of a reward decoder and policy from interaction data.

Other authors have investigated alternatives to rewards for agent behavior declaration such as via convex constraints (Miryoosefi et al., 2019) or to satisfy multiple objectives (Agarwal et al., 2014). The specification and nature of the feedback in IGL is less structured than in these settings.

Language games attempt to model the emergence of grounded communication between multiple agents cooperating to succeed in a task where each agent has partial information (Nowak et al., 1999; Bouchacourt & Baroni, 2018). In language games, reward is observed and grounding is required to communicate context information; whereas in IGL the reward is unobserved and the context is fully observed.

8. Discussion

We have proposed a novel setting, Interaction-Grounded Learning, in which a learner learns to interact with the environment in the absence of any explicit reward signals. The learner observes a context vector from the environment, takes an action, and observes a feedback vector. Without having a grounding for the feedback, the agent makes the assumption that there is latent reward signal in the feedback. With a conditional independence assumption on this latent reward, the agent uses an algorithm to discover this latent reward in order to ground its policies. We have proposed the E2G algorithm and proven that when the assumptions are met, it can solve IGL.

This work leverages the assumption of *conditional independence* of the feedback vector from the context and action given the reward. Of the required assumptions, this assumption may appear the most restrictive. This assumption is reasonable for some problems (e.g., a smart speaker reacting to idiosyncratic user vocabulary indicating approval or disapproval) and unreasonable for others (e.g., in a BCI application, EEG signals exhibit autocorrelation). However, in many signal detection application areas including EEG-based neurofeedback, which is a motivating application of this setting, it is common practice to postprocess the data to approximately satisfy this assumption by regressing out conditions and analyzing residuals.

Relaxing the assumption of conditional independence is a direction for future work. For problems where the feedback vector is influenced by the context and action, it may be possible to synthesize a conditionally independent signal, e.g., via variational approximations to mutual information (Belghazi et al., 2018) or via regression residuals (Shah et al., 2020). Indeed semi-parametric regression approaches are used pervasively in functional neuroimaging studies. Neural signals corresponding to successive conditions are commonly orthogonalized using a General Linear Model (Momennejad & Haynes, 2013) or Finite Impulse Response (Momennejad & Haynes, 2012). Regression and residual approaches thus orthogonalize the signal prior to further analysis with other machine learning methods. In future work we hope to develop an approach to sanitize the signal to ensure conditional independence is satisfied.

Interaction-grounded learning can be applied to many interesting domains. Among them are applications to BCI, prosthetics, and neurofeedback. BCI and Neurofeedback have been applied to improve memory (Fukuda & Woodman, 2015), train attention (Mishra & Gazzaley, 2015; deBettencourt et al., 2015), and facilitate learning and memory during sleep (Antony et al., 2018). For example, in functional neuroimaging studies of closed-loop neurofeedback, the content on screen reacts to the neural signals of the human participant in order to help the participant gain control of a given neural state, e.g. in training attention (Mishra & Gazzaley, 2015; Muñoz-Moldes & Cleeremans, 2020). To do so, a classifier is often extensively trained on numerous labeled samples of neural data (e.g. real-time fMRI signals of attentional state) before it is applied to reading and reacting to brain data (deBettencourt et al., 2015). With IGL, successful feedback (e.g. neurofeedback) may be achievable without supervised training.

Broadly, in all previous BCI work the feedback was grounded in extensive training. This is not ideal in many scenarios, such as the case of locked-in patients who may be conscious but lack the ability to ground their neural signals in other responses. IGL may be specifically helpful for such cases. Moreover, in neurofeedback applications (Muñoz-Moldes & Cleeremans, 2020) typically a human participant learns to calibrate their neural response to the feedback of an algorithm. Conversely, Interaction-Grounded Learning is a setting where the learner algorithm is required to calibrate itself to interpret un-grounded feedback from the environment, e.g. human EEG signals. IGL opens up possibilities for these and broader future directions, both in terms of research and application. Further examples follow.

Applications of IGL in more standard human-computer interface problems are potentially powerful. An example problem here is interpreting human gestures—people learn and use many gestures in a personalized way when working with each other. Could a robot naturally learn to interpret the gestures of a human using IGL techniques? Could the operating system of a laptop computer use IGL techniques to improve the interpretation of mouse, touchscreen, and/or viewed gestures? Realizing the benefits of IGL here requires a paradigm shift away from designed interfaces towards learned interfaces. The most extreme example of a designed interface is perhaps a keyboard. The keyboard has been very successful, yet the shift to small form factor compute devices has made keyboards significantly more awkward necessitating the development of new kinds of interfacing which inherently suffer from more ambiguity. Ambiguities in interpreting touch-screen gestures, handwriting, speech, or gestures and body language are areas where interaction-grounded learning may be valuable.

Finally, it is worth noting that in this work we assume a stationary environment. A two-agent IGL scenario is equivalent to one agent being oblivious, while the other agent might try to adjust the feedback vector to help grounding succeed. This beneficial learning variant of IGL could overlap with the language game literature, if the reward is considered privileged information for one of the agents. The language game variant of IGL is a fascinating topic for future theoretical and empirical investigations.

References

Agarwal, A., Badanidiyuru, A., Dudík, M., Schapire, R. E., and Slivkins, A. Robust multi-objective learning with mentor feedback. In Balcan, M., Feldman, V., and Szepesvári, C. (eds.), *Proceedings of The 27th Conference on Learning Theory, COLT 2014, Barcelona, Spain, June 13-15, 2014*, volume 35 of *JMLR Workshop and Conference Proceedings*, pp. 726–741. JMLR.org, 2014. URL `http://proceedings.mlr.press/v35/agarwal14b.html`.

Akinola, I., Wang, Z., Shi, J., He, X., Lapborisuth, P., Xu, J., Watkins-Valls, D., Sajda, P., and Allen, P. K. Accelerated robot learning via human brain signals. In *2020 IEEE International Conference on Robotics and Automation, ICRA 2020, Paris, France, May 31 - August 31, 2020*, pp. 3799–3805. IEEE, 2020. doi: 10.1109/ICRA40945.2020.9196566. URL `https://doi.org/10.1109/ICRA40945.2020.9196566`.

Antony, J. W., Piloto, L., Wang, M., Pacheco, P., Norman, K. A., and Paller, K. A. Sleep Spindle Refractoriness Segregates Periods of Memory Reactivation. *Curr Biol*, 28(11):1736–1743, 06 2018.

Bartók, G., Foster, D. P., Pál, D., Rakhlin, A., and Szepesvári, C. Partial monitoring—classification, regret bounds, and algorithms. *Mathematics of Operations Research*, 39(4):967–997, 2014.

Belghazi, M. I., Baratin, A., Rajeswar, S., Ozair, S., Bengio, Y., Courville, A., and Hjelm, R. D. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.

Bouchacourt, D. and Baroni, M. How agents see things: On visual representations in an emergent language game. *arXiv preprint arXiv:1808.10696*, 2018.

Cesa-Bianchi, N., Lugosi, G., and Stoltz, G. Regret minimization under partial monitoring. *Mathematics of Operations Research*, 31(3):562–580, 2006.

Chiang, K.-J., Emmanouilidou, D., Gamper, H., Johnston, D., Jalobeanu, M., Cutrell, E., Wilson, A., An., W. W., and Tashev, I. A closed-loop adaptive brain-computer interface framework. In *Proceedings of 10th International IEEE EMBS Conference On Neural Engineering (NER'21) being held virtually May 4-6, 2021*, 2021.

deBettencourt, M. T., Cohen, J. D., Lee, R. F., Norman, K. A., and Turk-Browne, N. B. Closed-loop training of attention with real-time brain imaging. *Nat Neurosci*, 18 (3):470–475, Mar 2015.

Fukuda, K. and Woodman, G. F. Predicting and Improving Recognition Memory Using Multiple Electrophysiological Signals in Real Time. *Psychol Sci*, 26(7):1026–1037, Jul 2015.

Hanawal, M., Szepesvari, C., and Saligrama, V. Unsupervised sequential sensor acquisition. In *Artificial Intelligence and Statistics*, pp. 803–811. PMLR, 2017.

Hong, J., Kveton, B., Zaheer, M., Chow, Y., Ahmed, A., and Boutilier, C. Latent bandits revisited. *Advances in Neural Information Processing Systems*, 33, 2020.

Katyal, K. D., Johannes, M. S., Kellis, S. S., Aflalo, T., Klaes, C., McGee, T. G., Para, M. P., Shi, Y., Lee, B. C., Pejsa, K., Liu, C., Wester, B. A., Tenore, F., Beaty, J. D., Ravitz, A. D., Andersen, R. A., and McLoughlin, M. P. A collaborative BCI approach to autonomous control of a prosthetic limb system. In *2014 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2014, San Diego, CA, USA, October 5-8, 2014*, pp. 1479–1482. IEEE, 2014. doi: 10.1109/SMC.2014.6974124. URL `https://doi.org/10.1109/SMC.2014.6974124`.

Langford, J. and Zhang, T. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in neural information processing systems*, pp. 817–824, 2008.

Lattimore, T. and Szepesvári, C. An information-theoretic approach to minimax regret in partial monitoring. In *Conference on Learning Theory*, pp. 2111–2139. PMLR, 2019.

Lattimore, T. and Szepesvári, C. *Bandit algorithms*. Cambridge University Press, 2020.

Loosli, G., Canu, S., and Bottou, L. Training invariant support vector machines using selective sampling. *Large scale kernel machines*, 2, 2007.

Maillard, O.-A. and Mannor, S. Latent bandits. In *International Conference on Machine Learning*, pp. 136–144. PMLR, 2014.

Mannor, S. and Shimkin, N. On-line learning with imperfect monitoring. In *Learning Theory and Kernel Machines*, pp. 552–566. Springer, 2003.

Mei, J., Xiao, C., Dai, B., Li, L., Szepesvári, C., and Schuurmans, D. Escaping the gravitational pull of softmax. *Advances in Neural Information Processing Systems*, 33, 2020.

Mertens, J.-F. Repeated games. In *Game Theory and Applications*, pp. 77–130. Elsevier, 1990.

Miryoosefi, S., Brantley, K., III, H. D., Dudík, M., and Schapire, R. E. Reinforcement learning with convex constraints. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 14070–14079, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/873be0705c80679f2c71fbf4d872df59-Abstract.html.

Mishra, J. and Gazzaley, A. Closed-loop cognition: the next frontier arrives. *Trends Cogn Sci*, 19(5):242–243, May 2015.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of machine learning*. MIT press, 2018.

Momennejad, I. and Haynes, J. D. Human anterior prefrontal cortex encodes the 'what' and 'when' of future intentions. *Neuroimage*, 61(1):139–148, May 2012.

Momennejad, I. and Haynes, J. D. Encoding of prospective tasks in the human prefrontal cortex under varying task loads. *J Neurosci*, 33(44):17342–17349, Oct 2013.

Muñoz-Moldes, S. and Cleeremans, A. Delineating implicit and explicit processes in neurofeedback learning. *Neurosci Biobehav Rev*, 118:681–688, 11 2020.

Ng, A. Y., Russell, S. J., et al. Algorithms for inverse reinforcement learning. In *Icml*, volume 1, pp. 2, 2000.

Nowak, M. A., Plotkin, J. B., and Krakauer, D. C. The evolutionary language game. *Journal of theoretical biology*, 200(2):147–162, 1999.

Piccolboni, A. and Schindelhauer, C. Discrete prediction games with arbitrary feedback and loss. In *International Conference on Computational Learning Theory*, pp. 208–223. Springer, 2001.

Shah, R. D., Peters, J., et al. The hardness of conditional independence testing and the generalised covariance measure. *Annals of Statistics*, 48(3):1514–1538, 2020.

Verma, A., Hanawal, M., Szepesvari, C., and Saligrama, V. Online algorithm for unsupervised sensor selection. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 3168–3176. PMLR, 2019.

Verma, A., Hanawal, M. K., Szepesvári, C., and Saligrama, V. Online algorithm for unsupervised sequential selection with contextual information. *arXiv preprint arXiv:2010.12353*, 2020.

Zhou, L. and Brunskill, E. Latent contextual bandits and their application to personalized recommendations for new users. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pp. 3646–3653, 2016.

Appendix

A. Detailed Proofs

A.1. Proofs for Section 3

Before providing the proof of Theorem 1, we first present the following lemma to show that $\varepsilon_{\mathsf{stat},n} < \eta/2$ is a sufficient condition of $\Delta\widehat{\psi} > 0$.

Lemma 7. *Let $(\widehat{\pi}, \widehat{\psi})$ and $\varepsilon_{\mathsf{stat},n}$ are defined same as Theorem 1, and $n = |\mathcal{D}|$. If n is large enough such that $\varepsilon_{\mathsf{stat},n} < \eta/2$, we have $\Delta\widehat{\psi} > 0$.*

Proof of Lemma 7. By the optimality of $(\widehat{\pi}, \widehat{\psi})$, we have

$$\begin{aligned} & \widehat{V}_{\mathcal{D}}(\widehat{\pi}, \widehat{\psi}) - \widehat{V}_{\mathcal{D}}(\pi_u, \widehat{\psi}) \geq \widehat{V}_{\mathcal{D}}(\pi^\star, \psi^\star) - \widehat{V}_{\mathcal{D}}(\pi_u, \psi^\star) \\ \Longrightarrow\ & V(\widehat{\pi}, \widehat{\psi}) - V(\pi_u, \widehat{\psi}) \geq V(\pi^\star, \psi^\star) - V(\pi_u, \psi^\star) - 2\varepsilon_{\mathsf{stat},n}. \end{aligned}$$

Plugging in the definition of $V(\widehat{\pi}, \widehat{\psi})$ and combining with Eq.(2), we obtain

$$\begin{aligned} \left(V(\widehat{\pi}) - V(\pi_u)\right)\Delta\widehat{\psi} \geq &\ \left(V(\pi^\star) - V(\pi_u)\right)\Delta\psi^\star - 2\varepsilon_{\mathsf{stat},n} \\ \overset{\text{(a)}}{\geq} &\ V(\pi_u) + \eta - 2\varepsilon_{\mathsf{stat},n} \\ > &\ V(\pi_u) && (\varepsilon_{\mathsf{stat},n} < \eta/2) \\ \geq &\ (0 - V(\pi_u)) \cdot (-1) \\ \overset{\text{(b)}}{\geq} &\ \big(V(\pi^\dagger) - V(\pi_u)\big)\Delta\psi^\dagger, && (5) \end{aligned}$$

where (a) follows from Assumption 2, i.e.,$\left(V(\pi^\star) - V(\pi_u)\right)\Delta\psi^\star \geq V(\pi_u) + \eta$, and (b) follows from that fact of $V(\pi^\dagger) \coloneqq \min_{\pi\in\Pi} V(\pi) \geq 0$ and $\Delta\psi^\dagger \coloneqq \min_{\psi\in\Psi} \Delta\psi \geq -1$.

If $\Delta\widehat{\psi} < 0$, we must have $|\Delta\widehat{\psi}| \leq |\Delta\psi^\dagger|$ by the definition of $\psi^\dagger$, and then $\left(V(\pi) - V(\pi_u)\right)\Delta\widehat{\psi}$ will be no greater than the RHS of Eq.(5) for any $\pi \in \Pi$. Therefore, $\Delta\widehat{\psi} < 0$ contradicts the results of Eq.(5), and so we must have $\Delta\widehat{\psi} > 0$. □

Proof of Theorem 1. For any π and ψ, we have the following results with probability $1 - \delta$,

$$\begin{aligned} & \left|\left(\widehat{V}_{\mathcal{D}}(\pi, \psi) - \widehat{V}_{\mathcal{D}}(\pi_u, \psi)\right) - \left(V(\pi, \psi) - V(\pi_u, \psi)\right)\right| \\ \leq & \left|\frac{1}{n}\sum_{i=1}^{n} \frac{\pi(a_i|x_i) - \frac{1}{K}}{d(a_i|x_i)}\psi(y_i) - \mathop{\mathbb{E}}_{(x,a)\sim d(\cdot,\cdot)}\left[\frac{1}{n}\sum_{i=1}^{n} \frac{\pi(a|x) - \frac{1}{K}}{d(a|x)}\psi(y)\right]\right| \\ \leq & \sqrt{\frac{2\mathrm{Var}_d\left[\frac{\pi(a|x) - \frac{1}{K}}{d(a|x)}\psi(y)\right]\log\frac{2|\Pi||\Psi|}{\delta}}{n}} + \frac{\max_{(x,a)\in\mathcal{X}\times\mathcal{A}}\left|\frac{\pi(a|x) - \frac{1}{K}}{d(a|x)}\psi(y)\right|\log\frac{2|\Pi||\Psi|}{\delta}}{3n} && \text{(Bernstein's inequality)} \\ \leq & \sqrt{\frac{4\max_{\pi\in\Pi}\left\|\frac{\pi(a|x)}{d(a|x)}\right\|_{2,d}\log\frac{2|\Pi||\Psi|}{\delta}}{n}} + \frac{\max_{(x,a)\in\mathcal{X}\times\mathcal{A}}\frac{1}{d(a|x)}\log\frac{2|\Pi||\Psi|}{\delta}}{3n} \eqqcolon \varepsilon_{\mathsf{stat},n}. && (\psi(\cdot) \in [0,1],\ \forall\psi \in \Psi) \end{aligned}$$

By the optimality of $(\widehat{\pi}, \widehat{\psi})$, we have

$$\begin{aligned} \widehat{V}_{\mathcal{D}}(\widehat{\pi}, \widehat{\psi}) - \widehat{V}_{\mathcal{D}}(\pi_u, \widehat{\psi}) \geq &\ \widehat{V}_{\mathcal{D}}(\pi^\star, \psi^\star) - \widehat{V}_{\mathcal{D}}(\pi_u, \psi^\star) \\ \geq &\ V(\pi^\star, \psi^\star) - V(\pi_u, \psi^\star) - \varepsilon_{\mathsf{stat},n} \\ = &\ \left(V(\pi^\star) - V(\pi_u)\right)\Delta\psi^\star - \varepsilon_{\mathsf{stat},n}. && (6) \end{aligned}$$

On the other hand,

$$\begin{aligned}\widehat{V}_{\mathcal{D}}(\widehat{\pi}, \widehat{\psi}) - \widehat{V}_{\mathcal{D}}(\pi_u, \widehat{\psi}) &\le V(\widehat{\pi}, \widehat{\psi}) - V(\pi_u, \widehat{\psi}) + \varepsilon_{\mathsf{stat},n} \\ &= \Big(V(\widehat{\psi}) - V(\pi_u)\Big)\, \Delta\widehat{\psi} + \varepsilon_{\mathsf{stat},n} \qquad (7)\end{aligned}$$

Therefore, combining Eq.(6) and Eq.(7), we obtain for $\widehat{\pi}$

$$\begin{aligned}\left(V(\widehat{\pi}) - V(\pi_u)\right) \Delta\widehat{\psi} + \varepsilon_{\mathsf{stat},n} &\ge \left(V(\pi^\star) - V(\pi_u)\right) \Delta\psi^\star \\ \Longrightarrow \left(V(\widehat{\pi}) - V(\pi_u)\right) \Delta\widehat{\psi} &\ge \left(V(\pi^\star) - V(\pi_u)\right) \Delta\psi^\star - 2\varepsilon_{\mathsf{stat},n} \\ \overset{\text{(a)}}{\Longrightarrow} \left(V(\widehat{\pi}) - V(\pi_u)\right) \Delta\psi^\star &\ge \left(V(\pi^\star) - V(\pi_u)\right) \Delta\psi^\star - 2\varepsilon_{\mathsf{stat},n} \\ \Longrightarrow V(\widehat{\pi}) &\ge V(\pi^\star) - \frac{2\varepsilon_{\mathsf{stat},n}}{\Delta\psi^\star},\end{aligned}$$

where (a) requires $\Delta\widehat{\psi} > 0$ which is following Lemma 7.

For $\widehat{\psi}$, we have

$$\begin{aligned}\left(V(\widehat{\pi}) - V(\pi_u)\right) \Delta\widehat{\psi} + \varepsilon_{\mathsf{stat},n} &\ge \left(V(\pi^\star) - V(\pi_u)\right) \Delta\psi^\star \\ \Longrightarrow \left(V(\widehat{\pi}) - V(\pi_u)\right) \Delta\widehat{\psi} &\ge \left(V(\pi^\star) - V(\pi_u)\right) \Delta\psi^\star - 2\varepsilon_{\mathsf{stat},n} \\ \overset{\text{(b)}}{\Longrightarrow} \left(V(\pi^\star) - V(\pi_u)\right) \Delta\widehat{\psi} &\ge \left(V(\pi^\star) - V(\pi_u)\right) \Delta\psi^\star - 2\varepsilon_{\mathsf{stat},n} \\ \Longrightarrow \Delta\widehat{\psi} &\ge \Delta\psi^\star - \frac{2\varepsilon_{\mathsf{stat},n}}{V(\pi^\star) - V(\pi_u)},\end{aligned}$$

where (b) requires $V(\widehat{\pi}) - V(\pi_u) > 0$ which can be obtained directly following $\Delta\widehat{\psi} > 0$. This completes the proof. □

A.2. Proofs for Section 4

Proof of Lemma 3. By the Hoeffding's equality, with probability at least $1 - \delta$, we have the following inequalities for any $\pi \in \Pi$ with $V(\pi) \le V(\pi_u)$ and $\psi \in \Psi$

$$\begin{aligned}\widehat{V}_{\mathcal{D}}(\pi, \psi) - \widehat{V}_{\mathcal{D}}(\pi_u, \psi) &= \mathop{\mathbb{E}}_{(x,a,y)\sim\mathcal{D}} \left[K\psi(y)\pi(a|x) - \psi(y)\right] \\ &\le \mathop{\mathbb{E}}_{(x,a,y)\sim\mu} \left[K\psi(y)\pi(a|x) - \psi(y)\right] + K\varepsilon_{\mathcal{D}} \\ &\le \left(V(\pi) - V(\pi_u)\right)(\psi_1 - \psi_0) + K\varepsilon_{\mathcal{D}} \\ &\le V(\pi_u) + K\varepsilon_{\mathcal{D}}.\end{aligned}$$

□

Proof of Lemma 4. Let $\widehat{\psi}_{\widehat{\pi}_{\mathcal{D}},\mathcal{D},1} \coloneqq \mathbb{E}\left[\widehat{\psi}_{\mathcal{D}}(y)\middle| r = 1\right]$ and $\widehat{\psi}_{\widehat{\pi}_{\mathcal{D}},\mathcal{D},0} \coloneqq \mathbb{E}\left[\widehat{\psi}_{\mathcal{D}}(y)\middle| r = 0\right]$, then, on one hand we have.

$$\begin{aligned}\widehat{V}_{\mathcal{D}}(\widehat{\pi}_{\mathcal{D}}, \widehat{\psi}_{\mathcal{D}}) - \widehat{V}_{\mathcal{D}}(\pi_u, \widehat{\psi}_{\mathcal{D}}) &\ge \widehat{V}_{\mathcal{D}}(\pi^\star, \psi_{\pi^\star}) - \widehat{V}_{\mathcal{D}}(\pi_u, \psi_{\pi^\star}) \\ &\ge V(\pi^\star, \psi^\star) - V(\pi_u, \psi^\star) - K\varepsilon_{\mathcal{D}} \\ &= (\psi_1^\star - \psi_0^\star)\left(V(\pi^\star) - V(\pi_u)\right) - K\varepsilon_{\mathcal{D}}. \\ &= \Delta\psi^\star \left(V(\pi^\star) - V(\pi_u)\right) - K\varepsilon_{\mathcal{D}}. \qquad (8)\end{aligned}$$

On the other hand,

$$\begin{aligned}\widehat{V}_{\mathcal{D}}(\widehat{\pi}_{\mathcal{D}}, \widehat{\psi}_{\mathcal{D}}) - \widehat{V}_{\mathcal{D}}(\pi_u, \widehat{\psi}_{\mathcal{D}}) &\le V(\widehat{\pi}_{\mathcal{D}}, \widehat{\psi}_{\mathcal{D}}) - V(\pi_u, \widehat{\psi}_{\mathcal{D}}) + K\varepsilon_{\mathcal{D}} \\ &= (\widehat{\psi}_{\widehat{\pi}_{\mathcal{D}},\mathcal{D},1} - \widehat{\psi}_{\widehat{\pi}_{\mathcal{D}},\mathcal{D},0})\left(V(\widehat{\pi}_{\mathcal{D}}) - V(\pi_u)\right) + K\varepsilon_{\mathcal{D}}. \\ &= \Delta\widehat{\psi}_{\mathcal{D}}\left(V(\widehat{\pi}_{\mathcal{D}}) - V(\pi_u)\right) + K\varepsilon_{\mathcal{D}}. \qquad (9)\end{aligned}$$

By assuming Eq.(4), we know

$$\begin{aligned}
&\widehat{V}_{\mathcal{D}}(\widehat{\pi}_{\mathcal{D}}, \widehat{\psi}_{\mathcal{D}}) - \widehat{V}_{\mathcal{D}}(\pi_u, \widehat{\psi}_{\mathcal{D}}) > V(\pi_u) + K\varepsilon_{\mathcal{D}} \\
\Longrightarrow\ & V(\widehat{\pi}_{\mathcal{D}}, \widehat{\psi}_{\mathcal{D}}) - V(\pi_u, \widehat{\psi}_{\mathcal{D}}) + \varepsilon_{\mathcal{D}} > V(\pi_u) + K\varepsilon_{\mathcal{D}} \\
\Longrightarrow\ & V(\widehat{\pi}_{\mathcal{D}}, \widehat{\psi}_{\mathcal{D}}) - V(\pi_u, \widehat{\psi}_{\mathcal{D}}) > V(\pi_u) \\
\Longrightarrow\ & \Delta\widehat{\psi}_{\mathcal{D}} \left(V(\widehat{\pi}_{\mathcal{D}}) - V(\pi_u)\right) > V(\pi_u) \qquad (10) \\
\Longrightarrow\ & \Delta\widehat{\psi}_{\mathcal{D}} > 0, \qquad (11)
\end{aligned}$$

where the last inequality follows from the fact that $V(\widehat{\pi}_{\mathcal{D}}) - V(\pi_u) \geq -V(\pi_u)$ and $\Delta\widehat{\psi}_{\mathcal{D}} \geq -1$, and therefore, the LHS of Eq.(10) is at most $1/K$ if $\Delta\widehat{\psi}_{\mathcal{D}} < 0$.

Combining Eq.(8), Eq.(9), and Eq.(11), we obtain

$$\begin{aligned}
\Delta\widehat{\psi}_{\mathcal{D}} \left(V(\widehat{\pi}_{\mathcal{D}}) - V(\pi_u)\right) + K\varepsilon_{\mathcal{D}} &\geq \Delta\psi^\star \left(V(\pi^\star) - V(\pi_u)\right) - K\varepsilon_{\mathcal{D}} \\
\Longrightarrow V(\widehat{\pi}_{\mathcal{D}}) &\geq \frac{\Delta\psi^\star \left(V(\pi^\star) - V(\pi_u)\right) - 2K\varepsilon_{\mathcal{D}}}{\Delta\widehat{\psi}_{\mathcal{D}}} \\
&\geq \frac{\Delta\psi^\star \left(V(\pi^\star) - V(\pi_u)\right) - 2K\varepsilon_{\mathcal{D}}}{\Delta\psi^\star} \\
&= V(\pi^\star) - \frac{2K\varepsilon_{\mathcal{D}}}{\Delta\psi^\star}.
\end{aligned}$$

This completes the proof. □

Proof of Lemma 5. Let $\mathcal{D}$ is the $\mathcal{D}_i$ with $t > T_0 = 2K^2\iota/\eta^2$, then we have $2K\varepsilon_{\mathcal{D}} \leq \eta$ with probability at least $1-\delta$, by following the definition of $\varepsilon_{\mathcal{D}}$ ($\varepsilon_{\mathcal{D}} := \sqrt{\iota/2|\mathcal{D}|}$). Therefore, by Assumption 2, we obtain

$$\begin{aligned}
&\left(V(\pi^\star) - V(\pi_u)\right)\Delta\psi_{\pi^\star} > V(\pi_u) + 2K\varepsilon_{\mathcal{D}} \\
\Longrightarrow\ & V(\pi^\star, \psi_{\pi^\star}) - V(\pi_u, \psi_{\pi^\star}) > V(\pi_u) + 2K\varepsilon_{\mathcal{D}} \\
\Longrightarrow\ & \widehat{V}_{\mathcal{D}}(\pi^\star, \psi_{\pi^\star}) - \widehat{V}_{\mathcal{D}}(\pi_u, \psi_{\pi^\star}) > V(\pi_u) + K\varepsilon_{\mathcal{D}} \\
& \widehat{V}_{\mathcal{D}}(\widehat{\pi}_{\mathcal{D}}, \widehat{\psi}_{\mathcal{D}}) - \widehat{V}_{\mathcal{D}}(\pi_u, \widehat{\psi}_{\mathcal{D}}) > V(\pi_u) + K\varepsilon_{\mathcal{D}}.
\end{aligned}$$

This completes the proof. □

Proof of Theorem 2. By Lemma 4, we can bound the one-step regret of executing any $\widehat{\pi}_{\mathcal{D}_i}$ by $\frac{2K\varepsilon_{\mathcal{D}}}{\Delta\psi_{\pi^\star}}$, as long as $t > T_0 = 2K^2\iota/\eta^2$. Therefore, the scheduling of $\{n_t\}_{t=1}^{\infty}$ can be chosen using a similar strategy as (Langford & Zhang, 2008) as long as $t > {}^{2K^2}/\eta^2$, i.e.,

$$n_t = \begin{cases} 0, & t \leq {}^{2K^2}/\eta^2; \\ \lfloor\sqrt{t/K\iota}\rfloor, & t > {}^{2K^2}/\eta^2. \end{cases}$$

By the similar argument as (Langford & Zhang, 2008), we obtain the following regret bound,

$$\mathrm{Regret}(T) = \mathcal{O}\left(\frac{K^{1/3}T^{2/3}\iota^{1/3}}{\Delta\psi_{\pi^\star}} + \frac{2K^2\iota}{\eta^2}\right).$$ □

B. Details of Experiments

B.1. Mitigation Strategies of the Issues in Optimizing π and ψ Jointly

In the practical implementation of E2G, we observe two critical issues that could cause poor performance. i) When using the gradient-based method to optimize the softmax policy class, it exhibits sensitivity to parameter initialization[1]. ii) For the

[1] This is a well known difficulty of using gradient-based methods to optimize the softmax policy (Mei et al., 2020).

objective function Eq.(1), $(\pi^\dagger, \psi^\dagger)$ is also a local minimum of it, where $\pi^\dagger := \mathrm{argmin}_{\pi \in \Pi} V(\pi)$ and $\psi^\dagger := \mathrm{argmin}_{\psi \in \Psi} \Delta\psi$. It is observed in practice that, optimizing Eq.(1) could easily suffer from that local minimum and converge to the opposite optimization direction, $(\pi^\dagger, \psi^\dagger)$.

To address these two issues, we use the following two important components in our implementation:

Adaptive Restart Procedure Although there is no access to the explicit reward in the interaction-grounded learning, Eq.(2) shows that the value of $V(\pi, \psi) - V(\pi_{\mathsf{bad}}, \psi)$ can be used to measure the quality of π and ψ, where π_{bad} is the uniform policy. Therefore, we use the following indicator to measure the performance of our learned policy π empirically for both E2G,

$$\frac{1}{|\mathcal{D}|} \sum_{(x,a,y) \in \mathcal{D}} K\pi(a|s)\psi(y) - \frac{1}{|\mathcal{D}|} \sum_{(x,a,y) \in \mathcal{D}} \psi(y), \tag{12}$$

This indicator can be viewed as the importance weighted estimator of $V(\pi, \psi) - V(\pi_{\mathsf{bad}}, \psi)$ as we use the uniform policy as the data-gathering policy over our experiments. By Lemma 3, we know that the local minimum $(\pi^\dagger, \psi^\dagger)$ cannot have a large value on indicator Eq.(12) when we have enough data. In addition, we can use the value of Eq.(12) to detect if E2G achieves a near-optimal solution by the following lemma.

Lemma 8. *Let π_{bad} be the uniform policy with $\mathcal{D}$ obtained by π_{bad}. We define $\widehat{V}_{\mathcal{D}}(\pi, \psi) := K\, \mathbb{E}_{(x,a,y)\sim\mathcal{D}}\left[\psi(y)\pi(a|s)\right]$ and $\widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}}, \psi) := \mathbb{E}_{(x,a,y)\sim\mathcal{D}}\left[\psi(y)\right]$ for any (π, ψ). Then, if $\Delta\psi > 0$[2], we have*

$$V(\pi) \geq \frac{\widehat{V}_{\mathcal{D}}(\pi, \psi) - \widehat{V}_{\mathcal{D}}(\pi_{\mathsf{bad}}, \psi) - K\varepsilon_{\mathcal{D}}}{\Delta\psi^\star} + V(\pi_{\mathsf{bad}}).$$

Proof of Lemma 8.

$$\begin{aligned}
\widehat{V}_{\mathcal{D}}(\pi, \psi) - \widehat{V}_{\mathcal{D}}(\pi_u, \psi) &\leq V(\pi, \psi) - V(\pi_u, \psi) + K\varepsilon_{\mathcal{D}} \\
&= (V(\pi) - V(\pi_u))\Delta\psi + K\varepsilon_{\mathcal{D}} && \text{(by Eq.(2))} \\
&\leq (V(\pi) - V(\pi_u))\Delta\psi^\star + K\varepsilon_{\mathcal{D}} && (\Delta\psi > 0) \\
\Longrightarrow V(\pi) &\geq \frac{\widehat{V}_{\mathcal{D}}(\pi, \psi) - \widehat{V}_{\mathcal{D}}(\pi_u, \psi) - K\varepsilon_{\mathcal{D}}}{\Delta\psi^\star} + V(\pi_u).
\end{aligned}$$

This completes the proof. □

A direct consequence of Lemma 8 is that, if we have enough data, and the value of Eq.(12) is greater than some threshold close to $V(\pi_{\mathsf{bad}})$ (which is $1/K$ in our experiments), then the value of Eq.(12) actually controls the expected return of π.

Therefore, we propose an adaptive restart procedure that works in the following way: During the early training stage of E2G and CB, we set a threshold for CB and E2G. If the value of the above indicator does not go beyond that threshold, we restart the training of the corresponding algorithm until that threshold is surpassed.

Data-Driven Corrector for Reward Decoder Lemma 3 and Lemma 8 suggest that the adaptive restart procedure based on the value of Eq.(12) is able to not only address the issue of converging to a bad local minimum (when having enough data) but also overcomes the sensitivity to parameter initialization. However, the amount of data sufficient to avoid bad local minimum by applying the adaptive restart procedure is usually too large to use in the online setting.

We notice that if we define the opposite reward decoder $\widetilde{\psi} := 1 - \psi, \forall \psi \in \Psi$, then $\widetilde{\psi}^\dagger := 1 - \mathrm{argmin}_{\psi \in \Psi} \Delta\psi = \mathrm{argmax}_{\psi \in 1 - \Psi} \Delta\psi$, where $1 - \Psi := \{1 - \psi : \psi \in \Psi\}$. Since the Ψ class we used in our experiment is the linear classifier with sigmoid activation function, $1 - \psi$ and ψ will only differ in the sign before the parameters. Thus, we have $\Psi = 1 - \Psi$ by setting the parameter space to be $\mathbb{R}^d$, and $\widetilde{\psi}^\dagger = \mathrm{argmax}_{\psi \in \Psi} \Delta\psi = \psi^\star$.

Since optimizing Eq.(1) will either maximize $\Delta\psi$ or minimize $\Delta\psi$, if we can determine the learned ψ is actually converging to $\psi^\dagger = \mathrm{argmin}\, \Delta\psi$, choosing its opposite decoder $\widetilde{\psi}^\dagger = 1 - \psi^\dagger$ could still provide us the desired decoder of

[2] One sufficient condition of $\Delta\psi > 0$ is Eq.(4). In our experiments, we introduce a data-driven corrector for the reward decoder (stated as below). Together with the adaptive restart procedure, we could ensure $\Delta\psi > 0$ with much less data.

$\operatorname{argmax}_{\psi \in \Psi} \Delta\psi$. By following this fact, we add an additional layer at the output of each ψ as a data-driven reward decoder corrector, which works as follows:

(1) Calculate $\mathsf{sign} \coloneqq \frac{1}{|\mathcal{D}|} \sum_{y \in \mathcal{D}} \mathbb{1}(\psi(y) > 0.5)$.

(2) If $\mathsf{sign} \leq 0.5$, output the original prediction $\psi(y), \forall y$.

(3) If $\mathsf{sign} > 0.5$, output the corrected prediction $1 - \psi(y), \forall y$.

Note that, the corrector above is designed based on data obtained from the uniform policy π_{bad}. In this case, the number of reaction signals generated from $r = 0$ should be more than that of $r = 1$. Otherwise, that data-driven reward decoder corrector should be adjusted according to the data-gathering policy by using an importance weighted estimation.

B.2. Additional Experimental Details

We now provide some addition details in our implentation. The experiments were conducted using Google Colab CPU instance, which was based on Intel Xeon CPU (2.30GHz) and 12 GB memory. No GPU was used. The prototype codes were built over `Python` and `PyTorch`. With the single process in the setup above, a single trial of either batch experiment or iterative experiment took less than 15 minutes to finish.

Environment The infinite MNIST environment used in Section 5 is built based on the `infinite MNIST dataset` at `https://leon.bottou.org/projects/infimnist` and a `Python` building of the infinite MNIST dataset generator at `https://github.com/albietz/infimnist_py`. To increase the speed of the experiment, we pre-generate a context set with $\sim$ 500000 samples, and two sets of reaction vector, $\sim$ 100000 samples of both image "0" and image "1". At each time step, the context is randomly selected from that pre-generated context set, and the reaction vector is randomly selected from the two sets of reaction vectors according to the actual reward.

The environment in Section 6 is built similarly, and only differs in the way of generating the reaction vector sets. In ENV (1), we generated two reaction vector sets with size $\sim$ 100000 for both, one is consist of the mixture of images of "'0" and "1", with the ratio of $8:1$. Another reaction vector set only includes images of "2". Therefore, if the actual reward is 0, a random image from the first set is selected as the reaction vector, otherwise, a random image from the second set will be selected. Those of ENV (2) only exchange the positions of images of "1" and images of "2", and others are set similarly.

Implementation of Π and Ψ Both of the policy class Π and reward decoder Ψ are using linear classifiers. The policy Π is implemented using the regular softmax policy with the temperature of 1, and the reward decoder class Ψ uses the sigmoid prediction with temperature of 0.1.

Details in the Iterative Algorithm The total number of round is 10000 in our experiment (the "round" denotes "i" in step 1 of Algorithm 1, and each "round" may contain more than one interactions due to Algorithm 1's suggestion). We set a number of 4000 samples as the "warm-up" data for $\mathsf{E2G}$, which is suggested by Theorem 2. After that, we use the uniform policy for exploration and the exploitation scheduling is set based on the suggestion in Theorem 2. That is, at each round i (i starts with 4001), we act one-step uniform exploration and $\sqrt{i/100K}$ steps exploitation. To accelerate the training process, we update the parameter every 100 rounds.