INVERSE CONTEXTUAL BANDITS: LEARNING HOW BEHAVIOR EVOLVES OVER TIME

Anonymous authors
Paper under double-blind review

ABSTRACT

Understanding a decision-maker's priorities by observing their behavior is critical for transparency and accountability in decision processes—such as in healthcare. Though conventional approaches to policy learning almost invariably assume stationarity in behavior, this is hardly true in practice: Medical practice is constantly evolving as clinical professionals fine-tune their knowledge over time. For instance, as the medical community's understanding of organ transplantations has progressed over the years, a pertinent question is: How have actual organ allocation policies been evolving? To give an answer, we desire a policy learning method that provides *interpretable* representations of decision-making, in particular capturing an agent's *non-stationary* knowledge of the world, as well as operating in an *offline* manner. First, we model the evolving behavior of decision-makers in terms of contextual bandits, and formalize the problem of *Inverse Contextual Bandits* ("ICB"). Second, we propose two concrete algorithms as solutions, learning parametric and nonparametric representations of an agent's behavior. Finally, using both real and simulated data for liver transplantations, we illustrate the applicability and explainability of our method, as well as benchmarking and validating the accuracy of our algorithms.

1 INTRODUCTION

Modeling decision-making policies is a central concern in computational and behavioral science, with key applications in healthcare [1], economics [2], and cognition [3]. The business of *policy learning* is to determine an agent's decision-making policy given observations of their behavior. Typically, the objective is either to replicate the behavior of some demonstrator (cf. "imitation learning") [4, 5], or to match their performance on the basis of some reward function (cf. "apprenticeship learning") [6, 7]. However, equally important is the pursuit of "*descriptive modeling*" [8, 9]—that is, of learning interpretable parameterizations of behavior for auditing, quantifying, and understanding decision-making policies. For instance, recent work has studied representing behaviors in terms of rules [10], goals [11], intentions [12], preferences [13], subjective dynamics [14], as well as counterfactual outcomes [15].

Evolving Behaviors In this work, we ask a novel descriptive question: *How has the observed behavior changed over time?* While conventional approaches to policy learning almost invariably assume that decision-making agents are stationary, this is rarely the case in practice: In many settings, behaviors evolve constantly as decision-makers learn more about their environment and adjust their policies accordingly. In fact, disseminating new knowledge from medical research into actual clinical practice is itself a major endeavor in healthcare [16–18]. This research question is new: While capturing "variation in practice" in observed data has been studied in the context of demonstrations containing mixed policies [19, 20], multiple tasks [21, 22], and subgroup-/patient-specific preferences [8, 23], little work has attempted to capture variation in practice *over time* as an agent's knowledge of the world evolves.

Example (Organ Allocations) As our core application, consider organ allocation practices for liver transplantations: In the medical community, our understanding of organ transplantations has changed numerous times over past decades [24–26]. Thus an immediate question lies in *how* actual organ allocation practices have changed over the years. Having a data-driven, quantitative, and—importantly—interpretable description of how practices have evolved is crucial: It would enable policy-makers to evaluate if the policies they introduced/updated have had the intended impact on practice; with humans in the loop, this would in turn play a substantial role in designing better policies going forward [8, 23].

To tackle questions of this form, we desire a policy learning method that satisfies three key desiderata: It should provide an (i.) *interpretable* description of observed decision-making behavior, in particular

capturing an agent's (ii.) *non-stationary* knowledge of the world, as well as operating in an (iii.) *offline* manner—since online experimentation is impossible in high-stakes environments such as healthcare.

Inverse Contextual Bandits To accomplish this, we first identify the organ allocation problem as a *contextual bandits* problem [27–29]: Given each arriving instance of patient and organ features (i.e. the context), an agent makes an allocation decision (i.e. the action), whence some measure of feedback is perceived (i.e. internal reward). Crucially, the environment is unknown to the agent and must be *actively learned*, so the agent maintains beliefs about their environment (i.e. internal knowledge). Not only must an agent select actions that "exploit" the knowledge they have, but they must also select actions that "explore" the environment to update their knowledge. Thus the behavior of a learning agent is naturally modeled as a generalized bandit strategy—that is, how to take actions based on their knowledge, and how to update their knowledge based on the outcomes of their actions (see Figure 1).

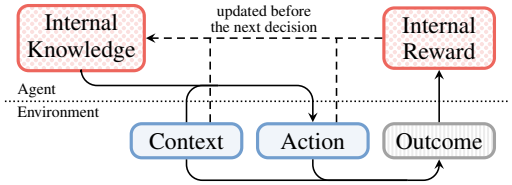


Figure 1. Bandit Strategy. The decision-maker maintains an internal belief about the environment as the basis of their decision-making policy. This knowledge is updated according to an internal reward that reflects their preferences over the outcomes of their decisions.

Now, the *forward* contextual bandits problem asks the (normative) question: Given an unknown environment, what is an effective bandit strategy that minimizes some notion of regret? By contrast, our focus here is instead on the opposite direction—that is, in formalizing and solving the problem of *Inverse Contextual Bandits* ("ICB"): We ask the (descriptive) question: Given demonstrated behavior from a decision-making agent, how has the agent's knowledge been evolving over time? Precisely, we wish to learn interpretable representations of generalized bandit strategies from the observable context-action pairs generated by those strategies—regardless of whether those strategies are effective.

Contributions Our contributions are three-fold. In the sequel, we first formalize the ICB problem, identifying it with the data-driven objective of inferring an agent's internal reward function along with their internal trajectory of beliefs about the environment (Section 2). Second, we propose two concrete learning algorithms, imposing different specifications regarding the agent's behavioral strategy: The first parameterizes the agent's knowledge in terms of Bayesian updates, whereas the second makes the milder specification that the agent's behavior evolves smoothly over time (Section 3). Third, through both simulated and real-world data for liver transplantations, we illustrate how ICB can be applied as an investigative device for recovering and explaining the evolution of organ allocation practices over the years, as well as benchmarking and validating the accuracy of our algorithms (Section 5).

2 Inverse Contextual Bandits

Preliminaries Consider a *decision-making problem* of the form $\mathbb{D} := (X, A, \mathcal{R}, \mathcal{T})$, where X indicates the state space, A the action space, $\mathcal{R} \in \Delta(\mathbb{R})^{X \times A}$ the reward dynamics, and $\mathcal{T} \in \Delta(X)^{X \times A}$ the transition dynamics. At each time $t \in \mathbb{Z}_+$, the decision-making agent is presented with some state $x_t \in X$ and decides to take an action $a_t \in A$, whence an immediate reward $r_t \sim \mathcal{R}(x_t, a_t)$ is received, and a subsequent state $x_{t+1} \sim \mathcal{T}(x_t, a_t)$ is presented. Let the *decision-making policy* employed by the agent be denoted $\pi \in \Delta(A)^X$, such that actions at any time are sampled according to $a_t \sim \pi(x_t)$.

In the forward direction, given dynamics $\mathcal{R}, \mathcal{T}$, the *reinforcement learning* problem ("RL") deals with determining the optimal policy that maximizes some notion of expected cumulative rewards [30]: $\pi^*_{\mathcal{R},\mathcal{T}} := \text{argmax}_{\pi \in \Delta(A)^X} \mathbb{E}_{\pi,\mathcal{R},\mathcal{T}} \sum_t r_t$. In the opposite direction, given some observed behavior policy π_b and transition dynamics $\mathcal{T}$, the *inverse reinforcement learning* problem ("IRL") deals with determining the reward dynamics $\mathcal{R}$ with respect to which π_b appears optimal. For instance, the classic max-margin approach seeks [31]: $\mathcal{R}^* := \text{argmin}_{\mathcal{R}}(\max_\pi \mathbb{E}_{\pi,\mathcal{R},\mathcal{T}} \sum_t r_t - \mathbb{E}_{\pi_b,\mathcal{R},\mathcal{T}} \sum_t r_t)$.

Environment Dynamics In conventional RL (and IRL), the decision-maker has unrestricted access to environment dynamics—either explicitly (i.e. $\mathcal{R}, \mathcal{T}$ are simply *known* and used in computing the optimal policy), or implicitly (i.e. the agent may *interact* freely with the environment during training). In contrast, in our setting the agent has no such luxuries—not only are the dynamics not known to the agent, but neither do they enjoy a distinct sandboxed training phase prior to live deployment. Without such access, the agent must consider both the information they may gain when taking actions (cf. "exploration") and also the expected rewards due to those actions (cf. "exploitation"). Note that this property results in much more difficult learning problems—in both the forward and inverse directions.

Parameters, Beliefs, and Updates Consider environments parameterized by *environment parameters* $\rho \in P$, such that the reward dynamics of an environment are given by $\mathcal{R}_\rho$, and the transition dynamics by $\mathcal{T}_\rho$. Let ρ^* denote the true environment parameter, such that the actual rewards received by an agent are distributed as $r_t \sim \mathcal{R}_{\rho^*}(x_t, a_t)$, and the actual states encountered by the agent are distributed as $x_{t+1} \sim \mathcal{T}_{\rho^*}(x_t, a_t)$. Since ρ^* is unknown, an agent takes actions on the basis of their beliefs about it; these beliefs are described by probability distributions $\mathcal{P}_\beta \in \Delta(P)$ parameterized by *belief parameters* $\beta \in B$. For each time t, let β_t capture the agent's belief at the beginning of that step.

Each time step, the agent first samples an environment parameter $\rho_t \sim \mathcal{P}_{\beta_t}$ according to their belief β_t, then takes an action according to $\pi_t^* := \pi^*_{\mathcal{R}_{\rho_t}, \mathcal{T}_{\rho_t}}$. Note that this ensures that any policy π_t^* is followed with probability proportional to the probability with which the agent believes it to be optimal at time step t. After receiving a reward r_t, the agent updates their current belief parameter according to a (possibly-stochastic) *belief-update function* $f \in \Delta(B)^{B \times X \times A \times \mathbb{R}}$, such that $\beta_{t+1} \sim f(\beta_t, x_t, a_t, r_t)$. Together with the initial belief parameter β_1, the agent's t-th step belief is a (possibly-stochastic) function of the history $h_t = \{\boldsymbol{x}_{1:t-1}, \boldsymbol{a}_{1:t-1}, \boldsymbol{r}_{1:t-1}\}$ defined recursively via the belief-update function f.

2.1 Contextual Bandits Setting

In this work, we consider state transitions that occur independently of past states and actions. Due to this property, decisions can be made greedily without consideration of what future states may be, and yields a contextual bandits problem [27–29]. Note that this captures the organ allocation problem well: Distributions of newly arriving organs are largely independent of prior allocation decisions; in fact, transplantation and allocation policies are often modeled in bandit-like settings [23, 32, 33]. Formally:

Definition 1 (Contextual Bandits) Consider a decision-making problem $\mathbb{D} := (X, A, \mathcal{R}, \mathcal{T})$, where $\mathcal{R}, \mathcal{T}$ are unknown to the agent. Let $\mathcal{T}(x, a) = \mathcal{T}'$ for some $\mathcal{T}' \in \Delta(X)$, for all $x \in X, a \in A$, such that policies are greedy: $\text{supp}(\pi_t^*(x_t)) = \text{argmax}_{a_t \in A} \bar{\mathcal{R}}_{\rho_t}(x_t, a_t)$, where $\bar{\mathcal{R}}_\rho(x, a) := \mathbb{E}_{r \sim \mathcal{R}_\rho(x,a)}[r]$ indicates the mean reward function, and ties are broken arbitrarily. Given a space of environment parameterizations P, the *contextual bandits* problem is to design a space of belief parameterizations B, and to determine the optimal belief-update function $f^* := \text{argmax}_{f \in \Delta(B)^{B \times X \times A \times \mathbb{R}}} \sum_t \mathbb{E}_{f, \pi_t^*, \mathcal{R}, \mathcal{T}}[r_t]$.

Now, suppose that an agent follows a bandit-type policy for T time steps; this would generate an *observational dataset* of contexts and actions $\mathcal{D} := \{\boldsymbol{x}_{1:T}, \boldsymbol{a}_{1:T}\}$. (Adhering to the contextual bandits literature, we refer to states as "contexts" hereafter). But since rewards r_t and beliefs β_t are quantities *internal* to the decision-maker, we assume that they are not part of the observational dataset: The former represents the agent's preferences over outcomes after observing contexts and taking actions, which is not explicitly observable; likewise, the latter represents the agent's beliefs about what kinds of outcomes their actions result in, which is also not explicitly observable. Thus we ask the novel question:

From $\mathcal{D}$, can we infer the true environment parameter ρ^*, as well as the belief trajectory $\boldsymbol{\beta}_{1:T}$?

Definition 2 (Inverse Contextual Bandits) Consider again a contextual bandit problem $\mathbb{D} := (X, A, \mathcal{R}, \mathcal{T})$, and recall that $\mathcal{R}, \mathcal{T}$ are unknown to the agent. Given an observational dataset $\mathcal{D}$, and a family of reward parameterizations P and belief parameterizations B, the *inverse contextual bandits* problem ("ICB") is to determine the true environment parameter ρ^* and the belief parameters $\boldsymbol{\beta}_{1:T}$ (see Figure 2).

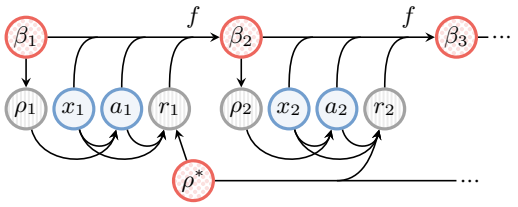


Figure 2. Graphical Model for ICB. We seek to determine the (dotted) red quantities given the (solid) blue quantities.

It is important to observe that the novelty here is *general*—that is, in posing the "inverse" question of how knowledge evolves over time. Focusing our attention on contextual bandits simply represents a specific choice of problem setting: In this first work on modeling non-stationary agents, it presents a more tractable problem for analysis, and is moreover especially suited for our motivating application in organ allocations. Note that in the bandit setting, the fact that transition dynamics are unknown to the agent is ultimately inconsequential due to greedy policies; we refer to ρ simply as "reward parameter" hereafter. We leave generalization to arbitrary transition dynamics $\mathcal{T} \in \Delta(X)^{X \times A}$ to future work.

Remark 1 (Environment vs. State Beliefs) When speaking of inferring "beliefs" here in our setting, we are referring to *environment beliefs*—that is, an agent's knowledge of the environment's rewards. It is important to distinguish this from *state beliefs* in partially-observed environments [34, 35]—that is,

an agent's estimate of which latent state they are in. State beliefs are computed by an agent using their (known) environment parameters, whereas environment beliefs concern the environment's (unknown) parameters themselves. State beliefs can be easily inferred [36]: all factors that contribute to state-belief updates (i.e. observations and actions) are observable; on the other hand, environment beliefs are more technically challenging due to latent factors (i.e. internal rewards $\boldsymbol{r}_{1:T}$) that are never observable.

Remark 2 (Subjective vs. Objective Reward) When speaking of learning what "rewards" an agent is optimizing, the novelty here that our objective refers to the full *belief trajectory* $\boldsymbol{\beta}_{1:T}$ of the agent. It is important to distinguish this from simply learning the *ground-truth parameter* ρ^* as in typical IRL. The ground-truth parameter is an objective quantity capturing the "prescriptive" notion of what the agent *ought to be* optimizing, whereas the belief trajectory is a subjective quantity capturing the "descriptive" notion of what the agent *appears to be* optimizing. Existing work in learning from non-stationary agents has focused solely on ρ^*, which is all that is required for apprenticeship [37, 38]. In contrast, for the goal of understanding behavior (and how it evolves), Definition 2 brings $\boldsymbol{\beta}_{1:T}$ to focus.

3 Bayesian Inverse Contextual Bandits

We operate in the standard contextual bandits setting [27]: Consider a d-dimensional action space A, and suppose at each time t the agent observes a k-dimensional context $x_t[a] \in \mathbb{R}^k$ for each possible action, thus the context space is given by $X \coloneqq \mathbb{R}^{d\times k}$. Let the space of reward parameters be $P \coloneqq \mathbb{R}^k$. Rewards are assumed linear with respect to contexts x_t, and normally-distributed with mean $\bar{\mathcal{R}}_\rho(x,a) = \langle \rho, x[a]\rangle$ and known variance σ^2; that is, $\mathcal{R}_\rho(x,a) \coloneqq \mathcal{N}(\langle \rho, x[a]\rangle, \sigma^2)$ with $\langle\cdot,\cdot\rangle$ denoting the inner product. (Note that these assumptions can be relaxed later in Section 3.1 below).

Belief Updates Before tackling a much more general case, we begin by modeling the agent's beliefs as Gaussian posteriors over ρ^*, and their belief-update function f as Bayesian updates. Formally, beliefs are described by the family of distributions $\mathcal{P}_\beta \coloneqq \mathcal{N}(\mu,\Sigma)$, where $\beta \coloneqq \{\mu,\Sigma\} \in \mathbb{R}^k \times \mathbb{R}^{k\times k}$, and $\mathcal{N}(\mu,\Sigma)$ is the multivariate Gaussian distribution with mean vector μ and covariance matrix Σ. At each time t, given the posterior $\beta_t = \{\mu_t,\Sigma_t\}$—such that $\rho^*|h_t \sim \mathcal{N}(\mu_t,\Sigma_t)$—and the observation (x_t,a_t,r_t), the belief-update $f(\beta_t,x_t,a_t,r_t)$ is the Dirac delta centered at $\beta_{t+1}{=}\{\mu_{t+1},\Sigma_{t+1}\}$, with:

$$\mu_{t+1} \coloneqq \Sigma_{t+1}\left(\Sigma_t^{-1}\mu_t + \frac{1}{\sigma^2} r_t x_t[a_t]\right), \quad \Sigma_{t+1} \coloneqq \left(\Sigma_t^{-1} + \frac{1}{\sigma^2} x_t[a_t] x_t[a_t]^\intercal\right)^{-1}, \tag{1}$$

Note that the initial belief parameter $\beta_1 = \{\mu_1,\Sigma_1\}$ represents the agent's (unknown) prior over ρ^*. Finally, to facilitate Bayesian analysis, we consider soft-optimal policies as usual (see e.g. [39]). Formally, instead of picking actions uniformly from $\text{argmax}_{a_t\in A}\bar{\mathcal{R}}_{\rho_t}(x_t,a_t)$, they are chosen as follows, with $\alpha \in \mathbb{R}_+$ adjusting the stochasticity of action selection ($\alpha \to \infty$ recovers the hard-optimal case):

$$\pi_t^*(x_t)[a_t] \coloneqq e^{\alpha\bar{\mathcal{R}}_{\rho_t}(x_t,a_t)} / \textstyle\sum_{a\in A} e^{\alpha\bar{\mathcal{R}}_{\rho_t}(x_t,a)}, \tag{2}$$

Bayesian Learning Under this setup, we propose an expectation-maximization (EM)-like algorithm for estimating the true reward parameter ρ^* and the initial belief parameter β_1, and samples the belief trajectory $\boldsymbol{\beta}_{2:T}$ in alternating steps. First, the joint distribution of all quantities of interest is given by:

$$\mathbb{P}(\rho^*,\beta_1,\boldsymbol{r}_{1:T},\boldsymbol{\rho}_{1:T},\mathcal{D}) = \underbrace{\mathbb{P}(\rho^*,\beta_1)}_{\text{Prior}} \textstyle\prod_{t=1}^T \underbrace{\mathbb{P}(x_t)}_{\mathcal{T}[x_t]} \underbrace{\mathbb{P}(\rho_t|\beta_t)}_{\mathcal{P}_{\beta_t}[\rho_t]} \underbrace{\mathbb{P}(a_t|x_t,\rho_t)}_{\pi_t^*(x_t)[a_t]} \underbrace{\mathbb{P}(r_t|\rho^*,x_t,a_t)}_{\mathcal{R}_{\rho^*}(x_t,a_t)[r_t]}. \tag{3}$$

Note that since each belief parameter β_t is a deterministic function of the initial belief parameter β_1 and the history h_t (cf. Equation 1), rewards $\boldsymbol{r}_{1:T-1}$ and belief parameters $\boldsymbol{\beta}_{2:T}$ are essentially interchangeable given the initial belief and context-action pairs $\mathcal{D}$ (i.e. one is completely identified by the other). Then, starting with some initial estimates $\hat{\rho}^*$ and $\hat{\beta}_1$ for the true reward parameter and the initial belief parameter, we iteratively obtain better estimates by performing the following two steps:

- *Expectation step:* Compute the expected log-likelihood of ρ^*,β_1 given the current estimates $\hat{\rho}^*,\hat{\beta}_1$:
$$\mathcal{Q}(\rho^*,\beta_1;\hat{\rho}^*,\hat{\beta}_1) \coloneqq \mathbb{E}_{\boldsymbol{r}_{1:T},\boldsymbol{\rho}_{1:T}|\hat{\rho}^*,\hat{\beta}_1,\mathcal{D}}[\log \mathbb{P}(\boldsymbol{r}_{1:T},\boldsymbol{\rho}_{1:T},\mathcal{D}|\rho^*,\beta_1)], \tag{4}$$
which we shall approximate by sampling reward values and reward parameters $\{\boldsymbol{r}_{1:T}^{(i)},\boldsymbol{\rho}_{1:T}^{(i)}\}_{i=1}^N$ from the distribution $\mathbb{P}(\boldsymbol{r}_{1:T},\boldsymbol{\rho}_{1:T}|\hat{\rho}^*,\hat{\beta}_1,\mathcal{D})$ using Markov chain Monte Carlo (MCMC) methods.
- *Maximization step:* Compute new estimates $\hat{\rho}^{*\prime},\hat{\beta}_1'$ that yield an improved expected log-posterior:
$$\mathcal{Q}(\hat{\rho}^{*\prime},\hat{\beta}_1';\hat{\rho}^*,\hat{\beta}_1) + \log\mathbb{P}(\hat{\rho}^{*\prime},\hat{\beta}_1') > \mathcal{Q}(\hat{\rho}^*,\hat{\beta}_1;\hat{\rho}^*,\hat{\beta}_1) + \log\mathbb{P}(\hat{\rho}^*,\hat{\beta}_1), \tag{5}$$
which can be achieved using gradient-based methods.

Sampling Reward Values and Reward Parameters Observe that reward values $\boldsymbol{r}_{1:T}$ are Gaussian-distributed when conditioned on reward parameters $\boldsymbol{\rho}_{1:T}$ (which means they can be sampled exactly):

Lemma 1 Let $\boldsymbol{r}_{1:T} = [r_1 \cdots r_T] \in \mathbb{R}^T$ and let $X_t = (1/\sigma^2)[x_1[a_1] \cdots x_t[a_t]\ \mathbf{0}_{k,(T-t)}] \in \mathbb{R}^{k\times T}$, where $\mathbf{0}_{i,j}$ is the zero matrix of size $i \times j$. Then we have that $\boldsymbol{r}_{1:T}|\boldsymbol{\rho}_{1:T}, \hat{\rho}^*, \hat{\beta}_1, \mathcal{D} \sim \mathcal{N}(\tilde{\mu}, \tilde{\Sigma})$, where

$$\tilde{\mu} \coloneqq \tilde{\Sigma}\left(X_T^\intercal \hat{\rho}^* + \textstyle\sum_{t=2}^T X_{t-1}^\intercal(\rho_t - \hat{\Sigma}_t\hat{\Sigma}_1^{-1}\hat{\mu}_1)\right), \tilde{\Sigma} \coloneqq \left((1/\sigma)^2 I + \textstyle\sum_{t=2}^T X_{t-1}^\intercal \hat{\Sigma}_t X_{t-1}\right)^{-1} . \quad (6)$$

Proof. All proofs are found in Appendix A. □

This motivates a Gibbs-like sampling procedure whereby samples from $\mathbb{P}(\boldsymbol{r}_{1:T}, \boldsymbol{\rho}_{1:T}|\hat{\rho}^*, \hat{\beta}_1, \mathcal{D})$ are approximated by sampling $\boldsymbol{r}_{1:T}$ and $\boldsymbol{\rho}_{1:T}$ in an alternating fashion from their respective conditional distributions $\mathbb{P}(\boldsymbol{r}_{1:T}|\boldsymbol{\rho}_{1:T}, \hat{\rho}^*, \hat{\beta}_1, \mathcal{D})$ and $\mathbb{P}(\boldsymbol{\rho}_{1:T}|\boldsymbol{r}_{1:T}, \hat{\rho}^*, \hat{\beta}_1, \mathcal{D})$ instead. However, note that sampling reward parameters $\boldsymbol{\rho}_{1:T}$ from its conditional is not as easy to perform exactly as sampling rewards values $\boldsymbol{r}_{1:T}$. We achieve this by first observing that the ρ_t's are independent from each other when conditioned on $\boldsymbol{r}_{1:T}$:

$$\begin{aligned}&\mathbb{P}(\boldsymbol{\rho}_{1:T}|\boldsymbol{r}_{1:T}, \hat{\rho}^*, \hat{\beta}_1, \mathcal{D}) \\ &\qquad \propto \textstyle\prod_{t=1}^T \mathbb{P}(\rho_t|\hat{\beta}_t)\mathbb{P}(a_t|x_t, \rho_t) .\end{aligned} \quad (7)$$

Then, we sample each ρ_t in turn by performing a single iteration of the Metropolis-Hastings algorithm using $\mathcal{P}_{\hat{\beta}_t}$ as the proposal distribution. Algorithm 1 (Bayesian Inverse Contextual Bandits) summarizes the overall learning procedure.

Algorithm 1 Bayesian ICB

Parameters: variance of rewards σ^2, stochasticity of action selections α, learning rate $\eta \in \mathbb{R}_+$
Input: dataset $\mathcal{D}$, prior $\mathbb{P}(\rho^*, \beta_1)$
Initialize $\hat{\rho}^*, \hat{\beta}_1 \sim \mathbb{P}(\rho^*, \beta_1)$
loop
  Initialize $\boldsymbol{r}_{1:T}^{(0)}, \boldsymbol{\rho}_{1:T}^{(0)}$
  for $i \in \{1, 2, \ldots, N\}$ **do**
    Sample $\boldsymbol{r}_{1:T}^{(i)} \sim \mathbb{P}(\boldsymbol{r}_{1:T}|\boldsymbol{\rho}_{1:T}^{(i-1)}, \hat{\rho}^*, \hat{\beta}_1, \mathcal{D})$ using Lemma 1
    for $t \in \{1, 2, \ldots, T\}$ **do**
      Sample $\rho', \rho'' \sim \mathcal{P}_{\beta_t[\hat{\beta}_1, \boldsymbol{x}_{1:t-1}, \boldsymbol{a}_{1:t-1}, \boldsymbol{r}_{1:t-1}^{(i)}]}$
      $p \leftarrow \min\{1, \frac{\mathbb{P}(a_t|x_t, \rho_t=\rho')}{\mathbb{P}(a_t|x_t, \rho_t=\rho'')}\}$
      $\rho_t^{(i)} \leftarrow \rho'$ w.p. p, and ρ'' otherwise
    end for
  end for
  $\bar{\mathcal{Q}}(\rho^*, \beta_1) = \frac{1}{N}\sum_{i=1}^N \log\mathbb{P}(\boldsymbol{r}_{1:T}^{(i)}, \boldsymbol{\rho}_{1:T}^{(i)}, \mathcal{D}|\rho^*, \beta_1)$
  $\{\hat{\rho}^*, \hat{\beta}_1\} \leftarrow \{\hat{\rho}^*, \hat{\beta}_1\}$
    $+\eta[\nabla_{\{\rho^*, \beta_1\}} \bar{\mathcal{Q}}(\rho^*, \beta_1)]_{\{\rho^*, \beta_1\}=\{\hat{\rho}^*, \hat{\beta}_1\}}$
    $+\eta[\nabla_{\{\rho^*, \beta_1\}} \log\mathbb{P}(\rho^*, \beta_1)]_{\{\rho^*, \beta_1\}=\{\hat{\rho}^*, \hat{\beta}_1\}}$
end loop
Output: $\hat{\rho}^*$, $\{\boldsymbol{\beta}_{1:T}^{(i)}[\hat{\beta}_1, \boldsymbol{x}_{1:T}, \boldsymbol{a}_{1:T}, \boldsymbol{r}_{1:T}^{(i)}]\}_{i=1}^N$

3.1 Nonparametric Bayesian Inverse Contextual Bandits

So far, we have modeled the learning procedure of the agent with Bayesian updates, and estimated the parameters that characterize their learning procedure (i.e. the true reward parameter ρ^* and the initial belief parameter β_1). Clearly, this approach can be similarly applied to different parameterizations of the agent's behavior (e.g. using a UCB-based model, or a general function approximator for f). In this section, instead of imposing a particular type of belief-update, we take a nonparametric approach and regard the agent's beliefs $\boldsymbol{\beta}_{1:T}$ more generally as a random process. First, we establish a prior $\mathbb{P}(\boldsymbol{\beta}_{1:T})$ over that belief process, then describe a procedure to sample from the posterior $\mathbb{P}(\boldsymbol{\beta}_{1:T}|\mathcal{D})$.

Gaussian Process Prior Different from above, now let the agent's beliefs be described by the family of distributions $\mathcal{P}_\beta \coloneqq \mathcal{N}(\beta, \Sigma_P)$ for $\beta \in \mathbb{R}^k$, where $\Sigma_P \in \mathbb{R}^{k\times k}$ controls the variability of reward parameters sampled from beliefs. Let $\boldsymbol{\beta}_{1:T} = [\beta_1 \cdots \beta_T] \in \mathbb{R}^{k\times T}$; here we consider a multivariate Gaussian process as prior:

$$\mathrm{vec}(\boldsymbol{\beta}_{1:T}) \sim \mathcal{N}(\mathbf{0}, \Sigma_T \otimes \Sigma_B) , \quad (8)$$

where $\otimes$ denotes the Kronecker product, $\Sigma_T \in \mathbb{R}^{T\times T}$ controls the covariances between different time steps, and $\Sigma_B \in \mathbb{R}^{k\times k}$ controls the covariances between different components of a given belief β_t. Although our methodology is applicable for any arbitrary Σ_T, we shall fix $(\Sigma_T)_{ij} = \min\{i, j\}$ so our prior becomes a multivariate Wiener process, and differences $\beta_t - \beta_{t-1}$ between consecutive beliefs are independent and identically distributed according to $\mathcal{N}(0, \Sigma_B)$. Intuitively, this means our prior favors belief trajectories that vary smoothly over time, where differences between consecutive beliefs are more or less bounded by Σ_B.

Algorithm 2 Nonparametric Bayesian ICB

Parameters: covariance matrices Σ_P and Σ_B
Input: dataset $\mathcal{D}$
Initialize $\boldsymbol{\rho}_{1:T}^{(0)}, \boldsymbol{\beta}_{1:T}^{(0)}$
for $i \in \{1, 2, \ldots, N\}$ **do**
  Sample $\boldsymbol{\beta}_{1:T}^{(i)} \sim \mathbb{P}(\boldsymbol{\beta}_{1:T}|\boldsymbol{\rho}_{1:T}^{(i-1)}, \mathcal{D})$ using Lemma 2
  for $t \in \{1, 2, \ldots, T\}$ **do**
    Sample $\rho', \rho'' \sim \mathcal{P}_{\beta_t^{(i)}}$
    $p \leftarrow \min\{1, \frac{\mathbb{P}(a_t|x_t, \rho_t=\rho')}{\mathbb{P}(a_t|x_t, \rho_t=\rho'')}\}$
    $\rho_t^{(i)} \leftarrow \rho'$ w.p. p, and ρ'' otherwise
  end for
end for
Output: $\{\boldsymbol{\beta}_{1:T}^{(i)}\}_{i=1}^N$

Sampling Beliefs from the Posterior Having established a Gaussian process prior $\mathbb{P}(\boldsymbol{\beta}_{1:T})$, observe that the posterior for $\boldsymbol{\beta}_{1:T}$ is still a Gaussian process when conditioned on the reward parameters $\boldsymbol{\rho}_{1:T}$:

Lemma 2 Let $\boldsymbol{\rho}_{1:T} = [\rho_1 \cdots \rho_T] \in \mathbb{R}^{k \times T}$. Then we have that $\text{vec}(\boldsymbol{\beta}_{1:T}) | \boldsymbol{\rho}_{1:T}, \mathcal{D} \sim \mathcal{N}(\tilde{\mu}, \tilde{\Sigma})$, with

$$\tilde{\mu} \coloneqq \tilde{\Sigma}(I \otimes \Sigma_P)^{-1} \,\text{vec}(\boldsymbol{\rho}_{1:T}), \quad \tilde{\Sigma} \coloneqq \left((\Sigma_T \otimes \Sigma_B)^{-1} + (I \otimes \Sigma_P)^{-1}\right)^{-1} . \tag{9}$$

Therefore, similar to the parametric version of our algorithm, we can approximate samples from the posterior $\mathbb{P}(\boldsymbol{\beta}_{1:T}|\mathcal{D})$ by sampling $\boldsymbol{\beta}_{1:T}$ and $\boldsymbol{\rho}_{1:T}$ in an alternating fashion from their respective conditional distributions $\mathbb{P}(\boldsymbol{\beta}_{1:T}|\boldsymbol{\rho}_{1:T}, \mathcal{D})$ and $\mathbb{P}(\boldsymbol{\rho}_{1:T}|\boldsymbol{\beta}_{1:T}, \mathcal{D})$, and discarding the samples for $\boldsymbol{\rho}_{1:T}$. Likewise, we sample each ρ_t in turn by performing a single iteration of the Metropolis-Hastings algorithm. Algorithm 2 (Nonparametric Bayesian ICB) summarizes the overall sampling procedure.

4 Related Work

Policy Learning In modeling an agent's decision-making from their observed behavior, the overarching goal is often in *imitation learning* (i.e. to replicate their actions) or in *apprenticeship learning* (i.e. to match their performance). The former directly parameterizes imitation policies as black-box functions, typically based on behavioral cloning [40–42] or state-action distribution matching [43–45]. The latter takes the indirect approach of first inferring some reward function for which the agent's demonstrations are assumed to be optimal—and on the basis of which an apprentice policy is trained; this is often done via Bayesian [39, 46, 47] or max-ent [48–50] IRL. In contrast, our overarching goal is in *descriptive modeling* (i.e. to learn interpretable parameterizations for explaining observed behavior) [8, 9]. For instance, recent work has represented policies in terms of human-understandable rules [10], goals [11], intentions [12], preferences [13], subjective dynamics [14], and counterfactuals [15].

Learning from Variation Precisely, we wish to understand how behavior has changed over time, especially relevant in healthcare as knowledge [24–26] and practices [16–18] continuously evolve. While *variation in practice* has been captured in policy learning from multi-modal [19, 20], multi-task [21, 22], and compound-task [51, 52] demonstrations, little has sought to describe *variation over time*. Some research has explored IRL from non-stationary agents, using labeled data generated by agents performing specific policy updates over time [37, 38] or ranked post-hoc by an expert [53, 54]. However, all of these simply attempt to infer the optimal reward (viz. ground-truth ρ^*) for the (prescriptive) goal of apprenticeship, revealing nothing about *how* the actual behavior has evolved (viz. trajectories $\boldsymbol{\beta}_{1:T}$) for the (descriptive) goal of understanding. The only close attempt has simply used a change-point detection intermediate step to allow inferring a sequence of optimal rewards as usual [55].

Bandits Setting To the best of our knowledge, this is the first formal attempt at learning interpretable representations of non-stationary behavior. In formulating and solving this challenge of learning $\boldsymbol{\beta}_{1:T}$, we have focused on the bandit setting [27–29]. Now, the general notion of an "inverse" bandit problem had been independently proposed by [57] and [58] as the bandit-based counterpart to apprenticeship from a learning agent [37, 38, 53, 54]. However, they both ignore the presence of contexts (viz. "non-contextual" bandits), and—more importantly—identical to the apprenticeship works above, they only attempt to infer the ground-truth reward ρ^*, without regard to the trajectory of evolution itself.

Table 1 contextualizes ICB with respect to related works in learning from decision-making behavior.

	Problem Formulation	**Axis of Variation**	**Learning Target**	**Interpretable Output**[1]	**Non-Stationary Agent**[2]	**Trajectory of Changes**[3]	**Stepwise Evolution**[4]	**Information Sharing**[5]	**Examples**
Policy Learning	Imitation Learning	-	π_b	✗	✗	-	-	-	[41]
	Apprenticeship Learning	-	ρ^*	✓	✗	-	-	-	[56]
	Interpretable Policy Learning	-	π_b	✓	✗	-	-	-	[14]
Variation in Practice	Multi-Modal Imitation	K clusters	$\{\pi_{b,k}\}$	✗	✗	-	-	-	[20]
	Multi-Task Apprenticeship	K clusters	$\{\rho_k^*\}$	✓	✗	-	-	-	[22]
	Personalized Reward Learning	N samples	$\{\rho_n^*\}$	✓	✗	-	-	-	[23]
Variation over Time	Ranked Reward Extrapolation	N samples	ρ^*	✓	✓	✗	✗	✗	[53]
	Inverse Soft Policy Improvement	M intervals	ρ^*	✓	✓	✗	✗	✗	[37]
	Change-Points Reward Learning	M intervals	$\{\rho_m^*\}$	✓	✓	✓	✗	✗	[55]
	Inverse Contextual Bandits	T time steps	$\rho^*, \{\beta_t\}$	✓	✓	✓	✓	✓	**(Ours)**

Table 1. Comparison with Related Work. We aim to provide an **[1]** *interpretable* account (i.e. reward-based or white-box) of **[2]** *non-stationary* behavior (i.e. evolving over time) that explicitly captures the **[3]** *trajectory* of changes itself (i.e. the agent's knowledge) in a **[4]** *stepwise* fashion (i.e. versus batched intervals), and **[5]** *shares information* between consecutive estimates (i.e. β_t, β_{t+1} are not independent). All works shown operate offline.

5 ILLUSTRATIVE EXAMPLES

Three aspects of our approach deserve empirical demonstration, and we shall highlight them in turn:

- *Explainability*: ICB should help us understand how medical practice has changed over the years. This is our primary motivation—in providing interpretable representations of variation over time.
- *Belief Accuracy*: ICB should recover accurate beliefs in a robust manner across a variety of learning agents—that is, it should not be sensitive to the learning procedure of the underlying agent.
- *Reward Accuracy:* ICB should recover accurate ground-truth reward parameters in a similarly robust manner—that is, it should be capable of "extrapolating beyond" suboptimal demonstrations.

Decision Environments We consider data from the Organ Procurement & Transplantation Network ("OPTN") as of December 4, 2020, which consists of patients registered for liver transplantation from 1995 to 2020 [59]. We are interested in the decision-making problem of matching organs that become available with patients who are waiting for a transplantation. For each decision, the action space A_t consists of patients who were in the waitlist at the time of an organ's arrival t, while the context $x_t[a]$ for each patient $a \in A_t$ includes the features of both the organ and the patient. We consider $k = 8$ features: $\{$ABO mismatch, age, creatinine, dialysis, INR, life support, bilirubin, weight difference$\}$.

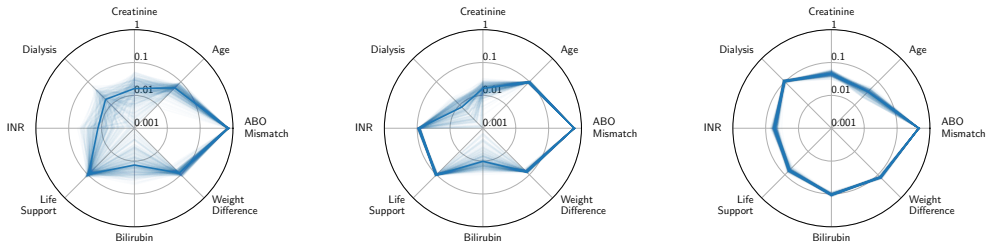
In addition to the OPTN dataset, to better validate the performance of our method in a more controlled manner, we also devise a semi-synthetic decision environment, where the features $\{x_t[a]\}_{a \in A_t}$ of potential organ-patient pairs are taken from the OPTN dataset, but the organs are matched with a final patient $a_t \in A_t$ for transplantation according to the policies of a variety of simulated learning agents. We determine the ground-truth reward parameter ρ^* of this semi-synthetic environment by performing simple linear regression over the survival time of patients who actually underwent a transplantation.

Learning Agents For the semi-synthetic experiments, we generate observational datasets by employing the following simulated agents, which includes a stationary agent as well as six learning agents:

- *Stationary*: The agent knows the ground-truth ρ^* and takes actions accordingly. Formally, $\rho_t = \rho^*$ for all t.
- *Sampling*: The agent employs the posterior sampling-based bandit strategy proposed in [27] (cf. Equation 1).
- *Optimistic*: The agent forms the same posteriors as *Sampling* but selects actions based on an optimistic estimation of their rewards: $a_t = \text{argmax}_{a \in A} \mu_t^\intercal x_t[a] + x_t[a]^\intercal \Sigma_t x_t[a]$. This is essentially LinUCB in [28].
- *Greedy*: The agent forms the same posteriors as *Sampling* but acts based on the mean reward parameters (i.e. $\rho_t = \mathbb{E}_{\rho \sim \beta_t}[\rho]$). In doing so, the agent only exploits their knowledge greedily and ignores exploration.
- *Stepping*: The agent first explores the environment using uniform preferences until some time step t^*, whence they learn the ground-truth reward parameter ρ^* and immediately begin taking actions accordingly. Formally, we set $\rho_t = -\mathbf{1}/k$ for $t \in \{1, \ldots, t^*\}$ and $\rho_t = \rho^*$ for $t \in \{t^* + 1, \ldots, T\}$. In healthcare settings, stepping behavior like this may occur when new guidelines are introduced and adopted.
- *Linear*: The agent learns the true reward parameter ρ^* in a linear fashion: $\rho_t = {}^t/_T \cdot \rho^* + (1 - {}^t/_T) \cdot (-\mathbf{1}/k)$.
- *Regressing*: The agent first acquires the ground-truth reward parameter ρ^* gradually until some time step t^*, at which point they begin to regress while retaining some amount of knowledge. Formally, we set $\rho_t = {}^t/_{t^*} \cdot \rho^* + (1 - {}^t/_{t^*}) \cdot \rho^0$ for $t \in \{1, \ldots, t^*\}$ and $\rho_t = {}^{(t-t^*)}/_{(T-t^*)} \cdot \rho^\gamma + (1 - {}^{(t-t^*)}/_{(T-t^*)}) \cdot \rho^*$ for $t \in \{t^* + 1, \ldots, T\}$, where $\rho^0 = -\mathbf{1}/k$ and $\rho^\gamma = \gamma\rho^* + (1 - \gamma)\rho^0$. Here, $\gamma \in [0, 1]$ controls the amount of knowledge that is retained. Note that this is a particularly challenging setting because the regressing agent's behavior does not improve monotonically (unlike the other agents considered here).

Benchmark Algorithms In addition to our two proposed algorithms Bayesian ICB (**B-ICB**) and Nonparametric Bayesian ICB (**NB-ICB**), we consider all the applicable algorithms in the literature:

- **B-IRL** [39]: This is the classic Bayesian inverse reinforcement learning algorithm that has been widely applied to a variety of problem settings [46, 60, 61]. As usual, it assumes $\rho_t = \rho^*$ for all $t \in \{1, \ldots, T\}$.
- M-**fold-IRL**: This runs a copy of Bayesian IRL for each of M equal-sized intervals in the dataset. Formally, this assumes $\rho_t = \rho^{(j)}$ for $t \in \{1 + \lfloor (j-1)T/M \rfloor, \ldots, \lfloor jT/M \rfloor\}$ and $j \in \{1, \ldots, M\}$. Note that setting $M = T$ is equivalent to assuming that beliefs over time are completely independent from each other.
- **CP-IRL** [55]: This recently-proposed method is similar to M-fold IRL in that it assumes $\rho_t = \rho^{(j)}$ for $t \in (t^{(j-1)}, t^{(j)})$ and $j \in \{1, ...M\}$, where $t^{(0)} = 0$ and $t^{(M)} = T$. However unlike M-fold IRL, it employs a change-point detection algorithm to learn the $t^{(j)}$'s together with the $\rho^{(j)}$'s (hence referred to as "CP"-IRL).
- **I-SPI** [37]: This recently-proposed method assumes the agent updates their behavioral policy via "soft policy improvements", which can be "inverted" to recover ρ^* (hence referred to as inverse "SPI"). In the contex-



(a) Feature Importances in 2000 **(b) Feature Importances in 2010** **(c) Feature Importances in 2020**

Figure 3. Relative Feature Importances in 2000, 2010, and 2020. INR gains significant importance in 2010 despite being the least important feature in 2000, which is consistent with the introduction of MELD scores in 2002.

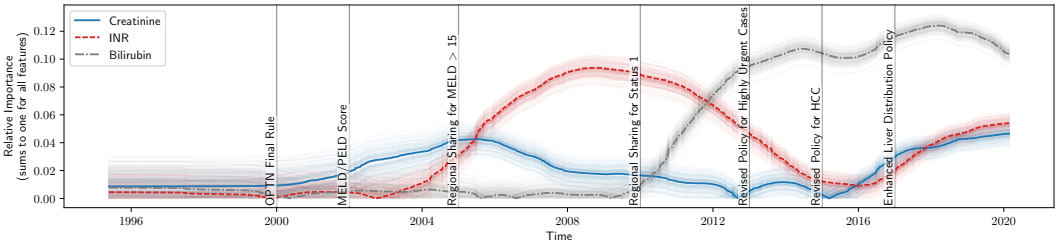


Figure 4. Relative Feature Importances over Time for Creatinine, INR, and Bilirubin. Significant changes in behavior have generally coincided with the important events surrounding guidance on liver allocation policies [62].

tual bandit setting, this is equivalent to assuming the agent is less stochastic over time (i.e. larger values of α). However, I-SPI only recovers the ground-truth ρ^*, and does not provide any estimate for belief trajectories.

- **T-REX** [53]: This recently-proposed method learns ρ^* using rankings between context-action pairs; absent explicit rankings, it assumes the agent learns monotonically, such that pairs encountered later in time are preferred over earlier ones (i.e. $(x_t, a_t) \prec (x_{t'}, a_{t'})$ if $t < t'$). Like I-SPI, T-REX only infers ρ^*, not beliefs.

In addition to these benchmark algorithms, as a baseline we also report the performance of simply estimating all preferences to be uniform—that is, $\hat{\rho}^* = \hat{\rho}_t = -\mathbf{1}/k$ (**Baseline**). Details regarding the implementation of both the learning agents and the benchmark algorithms can be found in Appendix B.

Explainability First, we direct attention to the potential utility of ICB as an *investigative device* for auditing and quantifying behaviors as they evolve. We use NB-ICB to estimate belief parameters $\{\beta_t = \mathbb{E}[\rho_t]\}_{t=1}^T$ for liver transplantations in the OPTN dataset. Since the agent's rewards are linear combinations of features weighted per their belief parameters, we may naturally interpret the normalized belief parameters $|\beta_t(i)| / \sum_{j=1}^k |\beta_t(j)|$ as the *relative importance* of each feature $i \in \{1, \ldots, k\}$.

Figure 3 shows the relative importances of all eight features at three different points in time: 2000, 2010, and 2020, and Figure 4 shows the importance of creatinine, INR, and bilirubin—components considered in the MELD score ("Model for End-Stage Liver Disease"), a scoring system for assessing the severity of chronic liver disease, [63]. Empirically, three observations immediately stand out: First, INR and creatinine appear to have gained significant importance over the 2000s, despite being the least important features in 2000. Second, their importances appear to have subsequently decreased towards the end of the decade. Third, since 2015 their importances appear to have steadily increased once again.

Interestingly, we can actually verify that these findings are perfectly consistent with the medical environments of their respective time periods. First, the MELD *scoring system* itself was introduced in 2002, which—using INR and creatinine as their most heavily weighted components—explains the rise in importance of those features in the 2000s. Second, over time there was an increase in the usage of MELD *exception points* (i.e. patients getting prioritized for special conditions like hepatocellular carcinoma, which are not directly reflected in their laboratory MELD scores), which explains the decrease in relative importance for such MELD components. Third, 2015 saw the introduction of an *official cap* on the use of MELD exception points (e.g. limited at 34 for hepatocellular carcinoma), which is consistent with the subsequent increase in relative importance of those features once again.

Figure 4 also plots important historical events that happened regarding liver allocation policies [62]. Of course, ICB has no knowledge of these events during training, so any apparent changes in behavior in the figure are discovered solely on the basis of organ-patient matching data in the OPTN dataset. Intriguingly, the importance of bilirubin appears to have not increased until 2008, instead of earlier when the MELD score was first introduced. Now, there are possible clinical explanations for this: For instance, bilirubin is not weighted as heavily as other features when computing MELD scores, so

Table 2. Mean Error of Belief Estimates. B-ICB and NB-ICB are best across all types of agents. Note that I-SPI and T-REX are not applicable as they do not estimate belief trajectories. Results show mean $\pm$ std. error (5 runs).

	Learning Agent						
Algorithm	Stationary	Sampling	Optimistic	Greedy	Stepping	Linear	Regressing
Baseline	0.477 ± 0.000	0.395 ± 0.030	0.391 ± 0.030	0.385 ± 0.020	0.238 ± 0.000	0.238 ± 0.000	0.238 ± 0.000
B-IRL	0.252 ± 0.248	0.206 ± 0.099	0.241 ± 0.195	0.248 ± 0.204	0.262 ± 0.027	0.165 ± 0.019	0.154 ± 0.018
5-fold IRL	0.294 ± 0.041	0.369 ± 0.075	0.313 ± 0.043	0.310 ± 0.033	0.333 ± 0.040	0.289 ± 0.031	0.292 ± 0.070
10-fold IRL	0.424 ± 0.017	0.407 ± 0.062	0.401 ± 0.061	0.399 ± 0.57	0.398 ± 0.027	0.386 ± 0.017	0.388 ± 0.043
T-fold IRL	1.168 ± 0.012	1.134 ± 0.021	1.140 ± 0.011	1.135 ± 0.011	1.117 ± 0.010	1.103 ± 0.011	1.105 ± 0.010
5-fold CP-IRL	0.272 ± 0.040	0.334 ± 0.039	0.280 ± 0.029	0.277 ± 0.029	0.294 ± 0.046	0.282 ± 0.029	0.278 ± 0.010
10-fold CP-IRL	0.409 ± 0.036	0.462 ± 0.075	0.388 ± 0.057	0.384 ± 0.055	0.374 ± 0.047	0.376 ± 0.049	0.401 ± 0.035
B-ICB	$\mathbf{0.120 \pm 0.032}$	$\mathbf{0.140 \pm 0.014}$	$\mathbf{0.121 \pm 0.016}$	$\mathbf{0.120 \pm 0.018}$	0.234 ± 0.016	0.153 ± 0.010	0.147 ± 0.026
NB-ICB	0.201 ± 0.027	0.178 ± 0.017	0.152 ± 0.042	0.149 ± 0.036	$\mathbf{0.150 \pm 0.012}$	$\mathbf{0.134 \pm 0.015}$	$\mathbf{0.140 \pm 0.017}$

Table 3. Mean Error of Ground-Truth Reward Estimates. B-ICB is best across all types of agents. M-fold IRL, CP-IRL, and NB-ICB are not applicable as they do not estimate ρ^*. Results show mean $\pm$ std. error (5 runs).

	Learning Agent						
Algorithm	Stationary	Sampling	Optimistic	Greedy	Stepping	Linear	Regressing
Baseline	0.477 ± 0.000	0.477 ± 0.000	0.477 ± 0.000	0.477 ± 0.000	0.477 ± 0.000	0.477 ± 0.000	0.477 ± 0.000
B-IRL	0.252 ± 0.248	0.224 ± 0.119	0.258 ± 0.168	0.266 ± 0.182	0.237 ± 0.031	0.266 ± 0.022	0.251 ± 0.020
I-SPI	0.231 ± 0.212	0.206 ± 0.092	0.276 ± 0.208	0.263 ± 0.183	0.237 ± 0.031	0.266 ± 0.021	0.250 ± 0.020
T-REX	1.482 ± 0.062	1.463 ± 0.067	1.480 ± 0.059	1.479 ± 0.060	1.447 ± 0.086	1.428 ± 0.089	1.438 ± 0.983
B-ICB	$\mathbf{0.121 \pm 0.032}$	$\mathbf{0.149 \pm 0.050}$	$\mathbf{0.148 \pm 0.036}$	$\mathbf{0.141 \pm 0.041}$	$\mathbf{0.161 \pm 0.027}$	$\mathbf{0.225 \pm 0.026}$	$\mathbf{0.246 \pm 0.029}$

their importance may not have been apparent until the later years, when patients generally became much sicker (with higher MELD scores overall). In any case, however, the point here is precisely that ICB is an *investigative device* that allows introspectively describing how policies have changed in this manner—such that notable phenomena may be duly investigated with a data-driven starting point.

Belief Accuracy Since it is not possible to compare the belief parameters of different algorithms directly, we define $\|\mathbb{E}_{\rho\sim\mathcal{P}_{\beta_t}}[\rho] - \mathbb{E}_{\rho\sim\mathcal{P}_{\hat{\beta}_t}}[\rho]\|_1$ as the error of belief estimate $\hat{\beta}_t$ at time t. Then, Table 2 shows the mean error of belief estimates learned by various algorithms in our semi-synthetic environment. As we would expect, B-ICB performs the best for the Sampling, Optimistic, and Greedy agents—which learn via Bayesian updates—while NB-ICB—which is more flexible in terms how it models belief trajectories—performs the best for the Stepping, Linear, and Regressing agents. Interestingly, we also observe that both M-fold IRL and CP-IRL perform worse than vanilla IRL. This could be due to the fact that—in both algorithms—the estimates for each interval are trained with fewer data points and independently from the other intervals; that is, there is no *information sharing* and potential similarities between adjacent intervals are disregarded. In contrast, both B-ICB and NB-ICB formalize some relationship between beliefs at different time steps (viz. Equations 1 and 8); under either formulation, beliefs at different time steps are never independent from each other.

Reward Accuracy Defining $\|\rho^* - \hat{\rho}^*\|_1$ as the error in estimating the ground-truth parameter ρ^*, Table 3 shows the mean error for various algorithms. B-ICB performs the best; this shows that not only can it recover (subjective) descriptors of how the agent *appears to be* behaving over time, but can also infer the (objective) prescriptor of how the agent *ought to be* behaving ideally. That is, B-ICB can also "extrapolate beyond" suboptimal demonstrations to infer the ground-truth reward. Interestingly, T-REX completely fails: This is because in the contextual bandits setting, it effectively assumes later context-action pairs (i.e. generated by a better policy) on average earn larger rewards than earlier ones (i.e. generated by a worse policy)—but this ignores the stochastic arrival of contexts themselves, which (in contextual bandits) is independent of the policy! For instance, sicker patients at a later time may always yield worse outcomes, no matter how much better the allocation policy has become.

6 Conclusion

In this paper, we motivated the importance of learning interpretable representations of non-stationary behavior, formalized the problem of inferring belief trajectories from data, and proposed concrete algorithms with demonstrated advantages in explainability and accuracy over any existing methods. Two points deserve brief comment in closing. First, we focused on contextual bandits; while this encapsulates a large class of applications, future work may investigate generalizing this approach to environments with arbitrary transition dynamics $\mathcal{T} \in \Delta(X)^{X\times A}$. Second, it is crucial to keep in mind that ICB does not claim to identify the real intentions of an agent: humans are complex, and rationality is bounded. What it does do, is to provide an interpretable explanation of how an agent is effectively behaving, offering a quantitative yardstick by which to investigate hypotheses and understand behavior.

REFERENCES

[1] A. Li, S. Jin, L. Zhang, and Y. Jia, "A sequential decision-theoretic model for medical diagnostic system," *Technol. Healthcare*, vol. 23, no. 1, pp. 37–42, 2015.

[2] J. A. Clithero, "Response times in economics: Looking through the lens of sequential sampling models," *J. Econ. Psychol.*, vol. 69, pp. 61–86, 2018.

[3] J. Drugowitsch, R. Moreno-Bote, and A. Pouget, "Relation between belief and performance in perceptual decision making," *PLOS ONE*, vol. 9, no. 5, 2014.

[4] M. Bain and C. Sammut, "A framework for behavioural cloning," *Mach. Intell.*, vol. 15, pp. 103–129, 1996.

[5] J. Ho and S. Ermon, "Generative adversarial imitation learning," in *Proc. 30th Conf. Neural Inf. Process. Syst.*, 2016.

[6] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proc. 21st Int. Conf. Mach. Learn.*, 2004.

[7] D. S. Brown, R. Coleman, R. Srinivasan, and S. Niekum, "Safe imitation learning via fast Bayesian reward inference from preferences," in *Proc. 37th Int. Conf. Mach. Learn.)*, 2020.

[8] D. Jarrett, A. Hüyük, and M. Van Der Schaar, "Inverse decision modeling: Learning interpretable representations of behavior," in *Proc. 38th Int. Conf. Mach. Learn.*, 2021.

[9] T. Bewley, "Am I building a white box agent or interpreting a black box agent?" *arXiv preprint arXiv:2007.01187*, 2020.

[10] T. Bewley, J. Lawry, and A. Richards, "Modelling agent policies with interpretable imitation learning," in *TAILOR Workshop at ECAI*, 2020.

[11] T. Zhi-Xuan, J. L. Mann, T. Silver, J. B. Tenenbaum, and V. K. Mansinghka, "Online Bayesian goal inference for boundedly-rational planning agents," in *Proc. 24th Conf. Neural Inf. Process. Syst.*, 2020.

[12] H. Yau, C. Russell, and S. Hadfield, "What did you think would happen? Explaining agent behaviour through intended outcomes," in *ICML Workshop on Extending Explainable AI*, 2020.

[13] D. Jarrett and M. van der Schaar, "Inverse active sensing: Modeling and understanding timely decision-making," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020.

[14] A. Hüyük, D. Jarrett, C. Tekin, and M. van der Schaar, "Explaining by imitating: Understanding decisions by interpretable policy learning," *Proc. 9th Int. Conf. Learn. Representations*, 2021.

[15] I. Bica, D. Jarrett, A. Hüyük, and M. van der Schaar, "Learning what-if explanations for sequential decision-making," *Proc. 9th Int. Conf. Learn. Representations*, 2021.

[16] J. M. Grimshaw and I. T. Russell, "Achieving health gain through clinical guidelines II: Ensuring guidelines change medical practice," *Qual. Health Care*, vol. 3, no. 1, pp. 45–52, 1994.

[17] R. Foy, G. MacLennan, J. Grimshaw, G. Penney, M. Campbell, and R. Grol, "Attributes of clinical recommendations that influence change in practice following audit and feedback," *J. Clin. Epidemiology*, vol. 55, no. 7, pp. 717–722, 2002.

[18] E. H. Bradley, M. Schlesinger, T. R. Webster, D. Baker, and S. K. Inouye, "Translating research into clinical practice: Making change happen," *J. Amer. Geriatrics Soc.*, vol. 52, no. 11, pp. 1875–1882, 2004.

[19] Z. Wang, J. Merel, S. Reed, G. Wayne, N. de Freitas, and N. Heess, "Robust imitation of diverse behaviors," in *Proc. 31st Conf. Neural Inf. Process. Syst.*, 2017.

[20] F.-I. Hsiao, J.-H. Kuo, and M. Sun, "Learning a multi-modal policy via imitating demonstrations with mixed behaviors," in *Proc. 32nd Conf. Neural Inf. Process. Syst.*, 2018.

[21] M. Babes, V. Marivate, and M. L. Littman, "Apprenticeship learning about multiple intentions," in *Proc. 28th Int. Conf. Mach. Learn.*, 2011.

[22] G. Ramponi, A. Likmeta, A. M. Metelli, A. Tirinzoni, and M. Restelli, "Truly batch model-free inverse reinforcement learning about multiple intentions," in *Proc. 23rd Int. Conf. Artif. Intell. Statist.*, 2020.

[23] Y. Qin, F. Imrie, A. Hüyük, D. Jarrett, A. E. Gimson, and M. van der Schaar, "Closing the loop in medical decision support by understanding clinical decision-making: A case study on organ transplantation," in *Proc. 35th Conf. Neural Inf. Process. Syst.*, 2021.

[24] T. E. Starzl, S. Iwatsuki, D. H. van Thiel, J. C. Gartner, B. J. Zitelli, J. J. Malatack, R. R. Schade, B. W. Shaw Jr., T. R. Hakala, J. T. Rosenthal, and K. A. Porter, "Evolution of liver transplantation," *Hepatology*, vol. 2, no. 5, pp. 614S–636S, 1982.

[25] R. Adam, P. McMaster, J. G. O'Grady, D. Castaing, J. L. Klempnauer, N. Jamieson, P. Neuhaus, J. Lerut, M. Salizzoni, S. Pollard, and F. Muhlbacher, "Evolution of liver transplantation in Europe: Report of the European Liver Transplant Registry," *Liver Transplantation*, vol. 9, no. 12, pp. 1231–1243, 2003.

[26] R. Adam, V. Karam, V. Delvart, J. O'Grady, D. Mirza, J. Klempnauer, D. Castaing, P. Neuhaus, N. Jamieson, M. Salizzoni, and S. Pollard, "Evolution of indications and results of liver transplantation in Europe. A report from the European Liver Transplant Registery (eltr)," *J. Hepatology*, vol. 57, no. 3, pp. 675–688, 2012.

[27] S. Agrawal and N. Goyal, "Thompson sampling for contextual bandits with linear payoffs," in *Proc. 30th Int. Conf. Mach. Learn.*, 2013, pp. 127–135.

[28] W. Chu, L. Li, L. Reyzin, and R. Schapire, "Contextual bandits with linear payoff functions," in *Proc. 14th Conf. Artif. Intell. Statis.*, 2011.

[29] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proc. 19th Int. Conf. World Wide Web*, 2010.

[30] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT press, 2018.

[31] A. Y. Ng, S. J. Russell *et al.*, "Algorithms for inverse reinforcement learning," *Proc. 17th Proc. Int. Conf. Mach. Learn.*, 2000.

[32] Y. S. Tang, A. A. Scheller-Wolf, and S. R. Tayur, "Generalized bandits with learning and queueing in split liver transplantation," *Social Sci. Res. Netw.*, 2021.

[33] J. Berrevoets, J. Jordon, I. Bica, A. Gimson, and M. van der Schaar, "OrganITE: Optimal transplant donor organ offering using an individual treatment effect," in *Proc. 24th Conf. Neural Inf. Process. Syst.*, 2020.

[34] J. Pineau, G. Gordon, S. Thrun *et al.*, "Point-based value iteration: An anytime algorithm for pomdps," in *Proc. 18th Int. Joint Conf. Artif. Intell.*, 2003.

[35] H. Kurniawati, D. Hsu, and W. S. Lee, "Sarsop: Efficient point-based pomdp planning by approximating optimally reachable belief spaces," *Robot.: Sci. Syst.*, 2008.

[36] J. Choi and K.-E. Kim, "Inverse reinforcement learning in partially observable environments," *J. Mach. Learn. Res.*, vol. 12, pp. 691–730, 2011.

[37] A. Jacq, M. Geist, A. Paiva, and O. Pietquin, "Learning from a learner," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019.

[38] G. Ramponi, G. Drappo, and M. Restelli, "Inverse reinforcement learning from a gradient-based learner," *arXiv preprint arXiv:2007.07812*, 2020.

[39] D. Ramachandran and E. Amir, "Bayesian inverse reinforcement learning," in *Proc. 20th Int. Joint Conf. Artif. Intell.*, 2007.

[40] D. A. Pomerleau, "Efficient training of artificial neural networks for autonomous navigation," *Neural Comput.*, vol. 3, no. 1, pp. 88–97, 1991.

[41] B. Piot, M. Geist, and O. Pietquin, "Boosted and reward-regularized classification for apprenticeship learning," in *Proc. 13th Int. Conf. Auton. Agents Multi-agent Syst.*, 2014.

[42] D. Jarrett, I. Bica, and M. van der Schaar, "Strictly batch imitation learning by energy-based distribution matching," in *Proc. 34th Conf. Neural Inf. Process. Syst.*, 2020.

[43] N. Baram, O. Anschel, and S. Mannor, "Model-based adversarial imitation learning," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017.

[44] W. Jeon, S. Seo, and K.-E. Kim, "A Bayesian approach to generative adversarial imitation learning," in *Proc. 32nd Conf. Neural Inf. Process. Syst.*, 2018.

[45] X. Zhang, Y. Li, Z. Zhang, and Z.-L. Zhang, "f-gail: Learning f-divergence for generative adversarial imitation learning," in *Proc. 34th Conf. Neural Inf. Process. Syst.*, 2020.

[46] J. Choi and K.-E. Kim, "MAP inference for Bayesian inverse reinforcement learning," in *Proc. 25th Conf. Neural Inf. Process. Syst.*, 2011.

[47] A. Balakrishna, B. Thananjeyan, J. Lee, F. Li, A. Zahed, J. E. Gonzalez, and K. Goldberg, "On-policy robot imitation learning from a converging supervisor," in *Conf. Robot. Learn.*, 2020.

[48] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning," in *Proc. 23rd AAAI Conf. Artif. Intell.*, 2008.

[49] J. Fu, K. Luo, and S. Levine, "Learning robust rewards with adversarial inverse reinforcement learning," *Proc. 6th Int. Conf. Learn. Representations*, 2018.

[50] A. H. Qureshi, B. Boots, and M. C. Yip, "Adversarial imitation via variational inverse reinforcement learning," *Proc. 7th Int. Conf. Learn. Representations*, 2019.

[51] M. Sharma, A. Sharma, N. Rhinehart, and K. M. Kitani, "Directed-info GAIL: learning hierarchical policies from unsegmented demonstrations using directed information," in *Proc. 7th Int. Conf. Learn. Representations*, 2019.

[52] S.-H. Lee and S.-W. Seo, "Learning compound tasks without task-specific knowledge via imitation and self-supervised learning," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020.

[53] D. S. Brown, W. Goo, P. Nagarajan, and S. Niekum, "Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019.

[54] D. S. Brown and S. Niekum, "Deep Bayesian reward learning from preferences," in *NeurIPS Workshop on Safety and Robustness in Decision-Making*, 2019.

[55] A. Likmeta, A. M. Metelli, G. Ramponi, A. Tirinzoni, M. Giuliani, and M. Restelli, "Dealing with multiple experts and non-stationarity in inverse reinforcement learning: An application to real-life problems," *Mach. Learn.*, vol. 110, p. 2541–2576, 2021.

[56] D. Lee, S. Srinivasan, and F. Doshi-Velez, "Truly batch apprenticeship learning with deep successor features," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, 2019.

[57] L. Chan, D. Hadfield-Menell, S. Srinivasa, and A. Dragan, "The assistive multi-armed bandit," in *Proc. 14th Annu. ACM/IEEE Int. Conf. Human-Robot Interact.*, 2019.

[58] W. Guo, K. K. Agrawal, A. Grover, V. Muthukumar, and A. Pananjady, "Learning from an exploring demonstrator: Optimal reward estimation for bandits," *arXiv preprint arXiv:2106.14866*, 2021.

[59] Organ Procurement and Transplantation Network and the Scientific Registry of Transplant Recipients. *Department of Health and Human Services, Health Resources and Services Administration, Healthcare Systems Bureau, Division of Transplantation, Rockville, MD.* 2020.

[60] C. A. Rothkopf and C. Dimitrakakis, "Preference elicitation and inverse reinforcement learning," in *Proc. 11th Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, 2011.

[61] S. Balakrishnan, Q. P. Nguyen, B. K. H. Low, and H. Soh, "Efficient exploration of reward functions in inverse reinforcement learning via Bayesian optimization," *Proc. 34th Conf. Neural Inf. Process. Syst.*, 2020.

[62] Organ procurement and transplantation network: Timeline of evolution of liver allocation and distribution policy. Accessed on: Sep. 9, 2021. [Online]. Available: https://optn.transplant.hrsa.gov/governance/key-initiatives/liver-timeline/.

[63] M. Bernardi, S. Gitto, and M. Biselli, "The MELD score in patients awaiting liver transplant: Strengths and weaknesses," *Frontiers Liver Transplantation*, vol. 54, no. 6, pp. 1297–1306, 2010.

A PROOFS OF LEMMAS

Proof of Lemma 1 First note that $\mu_t = \Sigma_t(\Sigma_1^{-1}\mu_1 + X_{t-1}\boldsymbol{r}_{1:T})$ by definition. Then, we have

$$
\begin{aligned}
&\mathbb{P}(\boldsymbol{r}_{1:T}|\boldsymbol{\rho}_{1:T}, \hat{\rho}^*, \hat{\beta}_1, \mathcal{D}) \\
&\quad\propto \mathbb{P}(\boldsymbol{r}_{1:T}, \boldsymbol{\rho}_{1:T}, \hat{\rho}^*, \hat{\beta}_1, \mathcal{D}) \\
&\quad\propto \prod_{t=1}^{T} \mathbb{P}(r_t|\hat{\rho}^*, x_t, a_t) \times \prod_{t=2}^{T} \mathbb{P}(\rho_t|\hat{\beta}_t) \\
&\quad\propto \prod_{t=1}^{T} \mathcal{R}_{\hat{\rho}^*}(x_t, a_t)[r_t] \times \prod_{t=2}^{T} \mathcal{P}_{\hat{\beta}_t}[\rho_t] \\
&\quad\propto \mathcal{N}(\sigma^2 X_T^\intercal \hat{\rho}^*, \sigma^2 I)[\boldsymbol{r}_{1:T}] \times \prod_{t=2}^{T} \mathcal{N}(\hat{\mu}_t, \hat{\Sigma}_t)[\rho_t] \\
&\quad\propto \mathcal{N}(\sigma^2 X_T^\intercal \hat{\rho}^*, \sigma^2 I)[\boldsymbol{r}_{1:T}] \times \prod_{t=2}^{T} \exp\Big(-\frac{1}{2} \cdot (\rho_t - \hat{\mu}_t)^\intercal \hat{\Sigma}_t^{-1} (\rho_t - \hat{\mu}_t) \Big) \\
&\quad\propto \mathcal{N}(\sigma^2 X_T^\intercal \hat{\rho}^*, \sigma^2 I)[\boldsymbol{r}_{1:T}] \\
&\qquad\times \prod_{t=2}^{T} \exp\Big(-\frac{1}{2} \cdot (\rho_t - \hat{\Sigma}_t\hat{\Sigma}_1^{-1}\hat{\mu}_1 - \hat{\Sigma}_t X_{t-1}\boldsymbol{r}_{1:T})^\intercal \cdot \hat{\Sigma}_t^{-1} \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \cdot (\rho_t - \hat{\Sigma}_t\hat{\Sigma}_1^{-1}\hat{\mu}_1 - \hat{\Sigma}_t X_{t-1}\boldsymbol{r}_{1:T}) \Big) \\
&\quad\propto \mathcal{N}(\sigma^2 X_T^\intercal \hat{\rho}^*, \sigma^2 I)[\boldsymbol{r}_{1:T}] \\
&\qquad\times \prod_{t=2}^{T} \exp\Big(-\frac{1}{2} \cdot \Big(\boldsymbol{r}_{1:T}^\intercal X_{t-1}^\intercal \hat{\Sigma}_t X_{t-1} \boldsymbol{r}_{1:T} - 2(\rho_t - \hat{\Sigma}_t\hat{\Sigma}_1^{-1}\hat{\mu}_1)^\intercal X_{t-1}\boldsymbol{r}_{1:T}) \Big) \Big) \\
&\quad\propto \mathcal{N}(\sigma^2 X_T^\intercal \hat{\rho}^*, \sigma^2 I)[\boldsymbol{r}_{1:T}] \\
&\qquad\times \prod_{t=2}^{T} \exp\Big(-\frac{1}{2} \cdot \Big(\boldsymbol{r}_{1:T} - (X_{t-1}^\intercal \hat{\Sigma}_t X_{t-1})^{-1} X_{t-1}^\intercal (\rho_t - \hat{\Sigma}_t\hat{\Sigma}_1^{-1}\hat{\mu}_1) \Big)^\intercal \\
&\qquad\qquad\qquad\qquad \cdot X_{t-1}^\intercal \hat{\Sigma}_t X_{t-1} \cdot \Big(\boldsymbol{r}_{1:T} - (X_{t-1}^\intercal \hat{\Sigma}_t X_{t-1})^{-1} X_{t-1}^\intercal (\rho_t - \hat{\Sigma}_t\hat{\Sigma}_1^{-1}\hat{\mu}_1) \Big) \Big) \\
&\quad\propto \mathcal{N}(\sigma^2 X_T^\intercal \hat{\rho}^*, \sigma^2 I)[\boldsymbol{r}_{1:T}] \\
&\qquad\times \prod_{t=2}^{T} \mathcal{N}\Big((X_{t-1}^\intercal \hat{\Sigma}_t X_{t-1})^{-1} X_{t-1}^\intercal (\rho_t - \hat{\Sigma}_t\hat{\Sigma}_1^{-1}\hat{\mu}_1), (X_{t-1}^\intercal \hat{\Sigma}_t X_{t-1})^{-1} \Big)[\boldsymbol{r}_{1:T}] \\
&\quad\propto \mathcal{N}(\sigma^2 X_T^\intercal \hat{\rho}^*, \sigma^2 I)[\boldsymbol{r}_{1:T}] \\
&\qquad\times \mathcal{N}\Bigg(\Big(\sum_{t=2}^{T} X_{t-1}^\intercal \hat{\Sigma}_t X_{t-1} \Big)^{-1} \Big(\sum_{t=2}^{T} X_{t-1}^\intercal (\rho_t - \hat{\Sigma}_t\hat{\Sigma}_1^{-1}\hat{\mu}_1) \Big), \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \Big(\sum_{t=2}^{T} X_{t-1}^\intercal \hat{\Sigma}_t X_{t-1} \Big)^{-1} \Bigg)[\boldsymbol{r}_{1:T}] \\
&\quad\propto \mathcal{N}\Bigg(\Big(\frac{1}{\sigma^2} I + \sum_{t=2}^{T} X_{t-1}^\intercal \hat{\Sigma}_t X_{t-1} \Big)^{-1} \Big(X_T^\intercal \hat{\rho}^* + \sum_{t=2}^{T} X_{t-1}^\intercal (\rho_t - \hat{\Sigma}_t\hat{\Sigma}_1^{-1}\hat{\mu}_1) \Big), \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \Big(\frac{1}{\sigma^2} I + \sum_{t=2}^{T} X_{t-1}^\intercal \hat{\Sigma}_t X_{t-1} \Big)^{-1} \Bigg)[\boldsymbol{r}_{1:T}].
\end{aligned}
$$

Proof of Lemma 2 We have

$$
\begin{aligned}
&\mathbb{P}(\boldsymbol{\beta}_{1:T}|\boldsymbol{\rho}_{1:T}, \mathcal{D}) \\
&\quad\propto \mathbb{P}(\boldsymbol{\beta}_{1:T}, \boldsymbol{\rho}_{1:T}, \mathcal{D}) \\
&\quad\propto \mathbb{P}(\boldsymbol{\beta}_{1:T}) \cdot \mathbb{P}(\boldsymbol{\rho}_{1:T}|\boldsymbol{\beta}_{1:T}, \mathcal{D})
\end{aligned}
$$

$$
\begin{aligned}
&\propto \mathcal{N}(\mathbf{0}, \Sigma_T \otimes \Sigma_B)[\mathrm{vec}(\boldsymbol{\beta}_{1:T})] \cdot \mathcal{N}(\mathrm{vec}(\boldsymbol{\beta}_{1:T}), I \otimes \Sigma_P)[\mathrm{vec}(\boldsymbol{\rho}_{1:T})] \\
&\propto \mathcal{N}(\mathbf{0}, \Sigma_T \otimes \Sigma_B)[\mathrm{vec}(\boldsymbol{\beta}_{1:T})] \cdot \mathcal{N}(\mathrm{vec}(\boldsymbol{\rho}_{1:T}), I \otimes \Sigma_P)[\mathrm{vec}(\boldsymbol{\beta}_{1:T})] \\
&\propto \mathcal{N}\Big(\big((\Sigma_T \otimes \Sigma_B)^{-1} + (I \otimes \Sigma_P)^{-1}\big)^{-1} (I \otimes \Sigma_P)^{-1} \mathrm{vec}(\boldsymbol{\rho}_{1:T}), \\
&\qquad\qquad \big((\Sigma_T \otimes \Sigma_B)^{-1} + (I \otimes \Sigma_P)^{-1}\big)^{-1} \Big) [\mathrm{vec}(\boldsymbol{\beta}_{1:T})] \,.
\end{aligned}
$$

B Experimental Details

Decision Environments There are 308,912 patients in the OPTN dataset who either were waiting for a liver transplantation or underwent a liver transplantation. Among these patients, we have filtered out the ones who never underwent a transplantation, the ones who were under the age of 18 or had a donor who was under the age of 18, and the ones who had a missing value for ABO Mismatch, Creatinine, Dialysis, INR, Life Support, Bilirubin, or Weight Difference. This filtering has left us with 31,059 patients, each corresponding to a different donor arrival. For the real-data experiments, we have sampled 2500 donor arrivals uniformly at random, and for each arrival, in addition to the patient who received the donor organ, sampled two more patients who were in the waitlist for a transplantation at the time of donor's arrival in order to form an action space. For the simulated experiments, we have sampled 500 donor arrivals and two patients for each arrival uniformly at random.

Learning Agents All learning agents select actions stochastically as described in (3) with $\alpha = 20$. Moreover, we have set $\sigma = 0.10$ for the sampling agent, $t^* = T/2$ for the stepping agent, and $t^* = T/2$ and $\gamma = 0$ for the regressing agent.

Benchmark Algorithms

- *B-IRL*: We have run the Metropolis-Hastings algorithm for 10,000 iterations to obtain 1,000 samples from $\mathbb{P}(\rho^*|\mathcal{D})$ with intervals of 10 iterations between each sample after 10,000 burn-in iterations. At each iteration, new candidate samples are generated by adding Gaussian noise with covariance matrix $0.005^2 I$ to the last sample. The final estimate $\hat{\rho}^*$ is formed by averaging all samples.
- *CP-IRL*: We have partitioned the trajectory $\{1, \ldots, T\}$ into sub-trajectories of length 10 when detecting change points and used B-IRL as our IRL solver.
- *I-SPI*: We have assumed that the agent performs soft policy improvement after each time step (starting with a uniformly random policy). When computing soft policy improvements, the transition probabilities $\mathcal{T}(x, a)[x'] = \mathcal{T}[x']$ were estimated using the empirical distribution of contexts in dataset $\mathcal{D}$—that is expectations of the form $\mathbb{E}_{x' \sim \mathcal{T}'}[\cdot]$ were approximated by computing $1/T \sum_{x' \in \mathcal{D}}[\cdot]$ instead. Similar to B-IRL, we have run the Metropolis-Hastings algorithm for 10,000 iterations to obtain 1,000 samples from $\mathbb{P}(\rho^*|\mathcal{D})$ with intervals of 10 iterations between each sample after 10,000 burn-in iterations. The same proposal distribution was used, and again, the final estimate $\hat{\rho}^*$ was formed by by averaging all samples.
- *T-REX*: We have maximized the likelihood given in [53] using the Adam optimizer with a learning rate of 0.001, $\beta_1 = 0.9$ and $\beta_2 = 0.999$ until convergence, that is when the likelihood stopped improving for 100 consecutive iterations.
- *B-ICB*: We have set $\sigma = 0.10$, $\alpha = 20$, and $N = 1000$ (with an additional 1000 samples as burn-in). When taking gradient steps, we have used the RMSprop optimizer with a learning rate of 0.001 and a discount factor of 0.9. We have run our algorithm for 100 iterations.
- *NB-ICB*: We have set $\Sigma_P = 5 \cdot 10^{-4} \cdot I$ and $\Sigma_B = 5 \cdot 10^{-5} \cdot I$. We have taken 1,000 samples from $\mathbb{P}(\boldsymbol{\beta}_{1:T}|\mathcal{D})$ with an interval of 10 iterations between each sample after 10,000 burn-in iterations (i.e. $N = 20{,}000$).

For the simulated experiments, the error bars are obtained by repeating each experiment five times.