RoME: A Robust Mixed-Effects Bandit Algorithm for Optimizing Mobile Health Interventions

Easton Huch[*]
Department of Statistics
University of Michigan
Ann Arbor, MI, USA
ekhuch@umich.edu

Jieru Shi[*]
Statistical Laboratory, DPMMS
University of Cambridge
Cambridge, England, UK
js2882@cam.ac.uk

Madeline R. Abbott
Department of Biostatistics
University of Michigan
Ann Arbor, MI, USA
mrabbott@umich.edu

Jessica R. Golbus
Michigan Medicine
Ann Arbor, MI, USA
jgolbus@umich.edu

Alexander Moreno
Independent Researcher[†]
alexander.f.moreno@gmail.com

Walter H. Dempsey
Department of Biostatistics
University of Michigan
Ann Arbor, MI, USA
wdem@umich.edu

Abstract

Mobile health leverages personalized and contextually tailored interventions optimized through bandit and reinforcement learning algorithms. In practice, however, challenges such as participant heterogeneity, nonstationarity, and nonlinear relationships hinder algorithm performance. We propose RoME, a **Ro**bust **M**ixed-**E**ffects contextual bandit algorithm that simultaneously addresses these challenges via (1) modeling the differential reward with user- and time-specific random effects, (2) network cohesion penalties, and (3) debiased machine learning for flexible estimation of baseline rewards. We establish a high-probability regret bound that depends solely on the dimension of the differential-reward model, enabling us to achieve robust regret bounds even when the baseline reward is highly complex. We demonstrate the superior performance of the RoME algorithm in a simulation and two off-policy evaluation studies.

1 Introduction

Mobile health (mHealth) uses smart devices to deliver digital notification interventions to users. These notifications nudge users toward healthier attitudes and behaviors. Because mHealth applications can monitor and react to users and their environment, they offer the promise of personalized, contextually tailored interventions. In practice, achieving this promise requires bandit or reinforcement learning (RL) algorithms that can accurately learn mHealth intervention effects, including how they vary by user, over time, and based on context. In this regard, contextual bandit algorithms are appealing due to strong empirical performance in other settings, their ability to customize intervention decisions based on changing contexts, and their simplicity and extensibility relative to full RL algorithms.

One unique aspect of mHealth (Greenewald et al., 2017) is the presence of a "do-nothing" action in which the mHealth app does not intervene. As mHealth rewards typically involve human decision making (e.g., whether the user chooses to exercise), the distribution of the the corresponding *baseline reward*—the reward under the do-nothing action—typically evolves in a complex, nonstationary fashion as time progresses and contexts change. In contrast, the *treatment effects*—the difference

[*]Authors Easton Huch and Jieru Shi contributed equally to this work.

[†]A portion of this work was performed while Alexander was employed at Luminous Computing.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

in the expected value of the reward relative to the do-nothing action—tend to be much more stable and can be adequately modeled using classical stationary models, such as linear models (Robinson, 1988).

Greenewald et al. (2017) introduced an action-centered contextual bandit algorithm with sublinear regret in this setting by replacing the observed reward, R_t, with a "pseudo-reward," $\tilde{R}_t$, where t indexes the time points. Denoting the binary action as $A_t \in \{0, 1\}$ with $A_t = 0$ corresponding to the do-nothing action, the pseudo-reward for A_t can be written as

$$\tilde{R}_t := (A_t - \pi_t)R_t = \pi_t(1 - \pi_t)R_t \left(\frac{A_t}{\pi_t} - \frac{1 - A_t}{1 - \pi_t} \right), \tag{1}$$

where π_t is the (known) probability of treatment assignment from the Thompson-sampling algorithm. We took an additional step to derive an equivalent expression (second equality), which reveals that $\tilde{R}_t$ is proportional to an inverse-probability-weighted (IPW) estimator. Crucially, $\tilde{R}_t$ provides an unbiased estimate of the treatment effect *regardless* of the nonstationarity of the baseline outcome.

Although Greenewald et al. (2017)'s analysis shows sublinear regret in nonstationary settings, our simulations show that other methods can achieve lower regret over finite time horizons. The reasons include its failure to address treatment effect heterogeneity, pool information across users, and model the baseline outcome, which leads to highly variable pseudo-rewards and slow learning. We propose to generalize their method to overcome these limitations by

1. imposing hierarchical structure in the treatment effect model with shared fixed effects and random effects for users and time (i.e., a mixed-effects model),
2. efficiently pooling across users and time via nearest-neighbor regularization (NNR), and
3. denoising rewards with flexible supervised learning models via the debiased machine learning (DML) framework of Chernozhukov et al. (2018).

We name the resulting method RoME to highlight that it is a **Ro**bust **M**ixed-**E**ffects algorithm. The primary contributions are (a) the RoME method, (b) a high-probability regret bound that relies solely on the dimension of the differential-reward model, which is typically much smaller than that of the complex baseline reward model, and (c) empirical comparisons demonstrating the superior performance of RoME in simulation and two real-world mHealth studies.

2 Related Work

Related work in statistics considers the estimation of treatment effects with longitudinal data. Cho et al. (2017) present a semiparametric random-effects method for estimating subject-specific *static* treatment effects. Qian et al. (2020) discusses the statistical challenges of applying linear mixed-effects models to mHealth data and shows that the resulting estimates may lack a causal interpretation.

In the bandit literature, several works address a growing pool of users. Ma et al. (2011) introduced NNR in recommender systems using homophily principles—similar nodes are more likely to be connected than dissimilar ones (McPherson et al., 2001). Cesa-Bianchi et al. (2013) adapted NNR for bandit settings, improving regret. Subsequent improvements focused on scalability and algorithm modifications for stronger regret bounds (Vaswani et al., 2017; Yang et al., 2020).

Other bandit approaches explicitly address the longitudinal setting in which treatment effects may evolve over time. The intelligentpooling method of Tomkins et al. (2021) employs a Gaussian linear mixed-effects model with both user- and time-specific random effects. Hu et al. (2021) provides a related approach based on generalized linear mixed effects models. Unlike our approach, both methods require the (generalized) linear conditional mean model to be correctly specified.

Aouali et al. (2023) and Lee et al. (2024) also propose bandit algorithms with mixed effects; methodologically, however, their use of mixed effects is dissimilar from that of the works above and our own. Aouali et al. (2023) employs random effects to relate treatments within categories (e.g., movies within a genre), and Lee et al. (2024) considers the case in which a single, discrete action leads to a multivariate reward that is affected by a user-specific random effect. In contrast, the other approaches listed above (and our own) do not assume any relationships among the actions, and they associate each discrete action with a single scalar-valued reward in an iterative fashion.

Other work develops semiparametric methods that lead to sublinear regret without requiring correct specification of the conditional mean reward. Krishnamurthy et al. (2018) and Kim & Paik (2019) propose improvements to the approach of Greenewald et al. (2017) that lead to stronger regret bounds. Kim et al. (2021, 2023) develop a doubly robust approach for both linear and generalized linear contextual bandits. However, in contrast to Tomkins et al. (2021), Hu et al. (2021), and our approach, these semiparametric methods are not immediately applicable to longitudinal settings.

3 Setting

3.1 Problem Statement

Recognizing the connection between mHealth policy learning and contextual bandit methods, we now consider a contextual multi-armed bandit environment with one control arm (the do-nothing action) denoted by $a = 0$ and q non-baseline arms corresponding to different actions or treatments. Individuals are indexed by $i = 1, 2, \ldots$ and decision points are indexed by $t = 1, 2, \ldots$. For each individual i at time t, the algorithm observes a context vector $S_{i,t} \in \mathcal{S}$, chooses an action $A_{i,t} \in [q] := \{0, \ldots, q\}$, and receives a reward $R_{i,t} \in \mathbb{R}$. The *differential reward* describes the difference in expected rewards between choosing a non-baseline arm and the control arm, defined as $\Delta_{i,t}(s, a) := \mathbb{E}\left[R_{i,t}|S_{i,t} = s, A_{i,t} = a\right] - \mathbb{E}\left[R_{i,t}|S_{i,t} = s, A_{i,t} = 0\right]$.

We denote the observation history up to time t for individual i as $\mathcal{H}_{i,t} := (S_{i1}, A_{i1}, R_{i1}, \ldots, S_{it})$. Actions are selected according to stochastic policies, $\pi_{i,t} : \mathcal{H}_{i,t} \times \mathcal{S} \to \mathcal{P}([q])$. We use $\pi_{i,t}(a|s)$ to denote the probability of action $a \in [q]$ given current context $s \in \mathcal{S}$ for a fixed (implicit) history. As in Greenewald et al. (2017), we assume that the probability of the control action is bounded:

Assumption 1. *There exists* $0 < \pi_{\min}, \pi_{\max} < 1$ *s.t* $\pi_{\min} < \pi_{i,t}(0|a), 1 - \pi_{i,t}(0|a) < \pi_{\max} \forall i, t, a.$

As noted in Greenewald et al. (2017), this assignment probability controls the number of messages that participants receive. It ensures that participants receive sufficient messages that they do not disengage, but not so many that they are overwhelmed or fatigued. We now define

$$A^* = \underset{a \in [q]}{\operatorname{argmax}}\, \mathbb{E}\left[R_{i,t}|S_{i,t}, A_{i,t} = a\right], \quad \bar{A}^* = \underset{a \in [q]\setminus\{0\}}{\operatorname{argmax}}\, \mathbb{E}\left[R_{i,t}|S_{i,t}, A_{i,t} = a\right],$$

the optimal treatment arm (including control) and the alternative arm, respectively. Respecting the restrictions in Assumption 1, we can now express the optimal policy, $\pi^*_{i,t}$, in two cases:

1. $A^* = 0$: $\pi^*_{i,t}(0|S_{i,t}) = \pi_{\max}$ and $\pi^*_{i,t}(\bar{A}^*|S_{i,t}) = 1 - \pi_{\max}$.
2. $A^* > 0$: $\pi^*_{i,t}(A^*|S_{i,t}) = 1 - \pi_{\min}$ and $\pi^*_{i,t}(0|S_{i,t}) = \pi_{\min}$.

These cases represent the appropriate boundary condition given the optimal action; we maximize the probability of the do-nothing action if $A^\star = 0$ and minimize it if $A^\star \neq 0$, selecting the next-best action with the highest possible probability. In Section 5.2, we define regret relative to this optimal policy, $\pi^*_{i,t}$.

Lastly, we consider a study design that progresses in *stages*, where individuals enter the study sequentially (Friedman et al., 2010), as illustrated in Figure 1. Staged recruitment serves two purposes in our analysis: it increases the fidelity of the theoretical setting by mimicking real-world recruitment strategies, and it allows us to estimate changes in the differential reward over time. Concretely, we first observe individual $i = 1$ at time $t = 1$. Then we observe individuals $i = (1, 2)$ at times $t = (2, 1)$. Subsequent stages k involve individuals $i \leq k$, each having had $k - i + 1$ observations, respectively. Define $\mathcal{O}_k = \{(i, t) : i \leq k \,\&\, t \leq k + 1 - i\}$ as the set of all observations in stage k. A notation summary is presented in Appendix F.

3.2 Doubly Robust Differential Reward

Following Greenewald et al. (2017), we assume that the differential reward is linear:

$$\Delta_{i,t}(s, a) = x(s, a)^\top \theta_{i,t}, \tag{2}$$

where $x(s, a) \in \mathbb{R}^{p \times 1}$ is a feature vector depending on the context and action. We allow the baseline reward, $g_t(s)$, to be a nonlinear and nonstationary function of context and time. Combined, these two

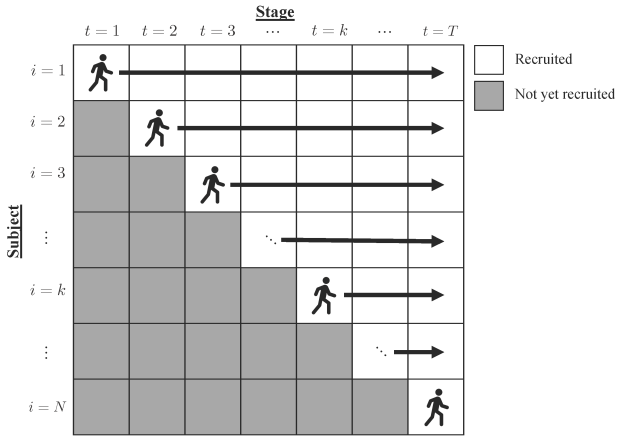

Figure 1: Illustration of the staged recruitment scheme. At each recruitment stage (each time point), a new participant is recruited and observed. At the same time, all participants who were recruited prior to the current stage are also observed again. Observations are not collected from participants who have yet to be recruited. For simplicity, we assume one participant is recruited at each stage.

assumptions imply a partially linear model for the conditional mean reward:

$$\mathbb{E}\left[R_{i,t}|S_{i,t}=s, A_{i,t}=a\right] = x(s,a)^\top \theta_{i,t}\delta_{a>0} + g_t(s), \tag{3}$$

where $\delta_{a>0}$ is an indicator of a non-baseline action. By incorporating both the linear predictor $x(s,a)^\top\theta_{i,t}\delta_{a>0}$ and the nonlinear baseline $g_t(s)$, we can maintain interpretability of the action effects while also allowing flexibility in specifying the observed rewards. The individual rewards are then realized according to the model

$$R_{i,t} = x(s,a)^\top \theta_{i,t}\delta_{a>0} + g_t(s) + \epsilon_{i,t},$$

where we make the following standard assumption (Abeille & Lazaric, 2017) regarding $\epsilon_{i,t}$:

Assumption 2. *$\epsilon_{i,t}$ is conditionally mean zero (i.e., $\mathbb{E}[\epsilon_{i,t}|\mathcal{H}_{i,t}]=0$) and sub-Gaussian with variance σ^2: $\mathbb{E}[\exp(\eta\epsilon_{i,t})|\mathcal{H}_{i,t}] \le \exp(\eta^2\sigma^2/2)$ for $\eta>0$.*

Assumption 2 ensures that large errors are sufficiently rare and that $\tilde{R}_t$ is proportional to an unbiased estimate of $\Delta_{i,t}(s,a)$. However, $\tilde{R}_t$ suffers from high variance, which leads to suboptimal performance in practice. Our proposed algorithm RoME, however, relies on an improved estimator, $\tilde{R}^f_{i,t}(s,\bar{a})$, that is also unbiased but achieves lower variance by denoising the rewards according to a working model, $f_{i,t}(s,a)$, for the conditional mean $r_{i,t}(s,a) := \mathbb{E}\left[R_{i,t}|S_{i,t}=s, A_{i,t}=a\right]$:

$$\tilde{R}^f_{i,t}(s,\bar{a}) := \underbrace{f_{i,t}(s,\bar{a}) - f_{i,t}(s,0)}_{\text{Model prediction}} + \underbrace{\frac{\{R_{i,t} - f_{i,t}(s,A_{i,t})\}}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}}_{\text{Debiasing term}}, \tag{4}$$

where $\bar{a}$ is a preselected non-baseline arm; i.e., we condition on $A_{i,t}\in\{0,\bar{a}\}$. The first component is a model-based prediction of the differential reward, and the second component is a debiasing term that uses IPW to correct for potential errors in the specification of f. This improved estimator is a form of pseudo-outcome, a common technique in the causal inference literature (Bang & Robins, 2005; Kennedy, 2023; Nie & Wager, 2021). We refer to it as a *Doubly Robust Differential Reward* because it produces an unbiased estimate of $\Delta_{i,t}(s,a)$ as long as either $\pi_{i,t}$ or $f_{i,t}$ is correctly specified (see proof in Appendix E.1); though, in practice, the benefit stems from (a) the robustness to misspecification of f (because $\pi_{i,t}$ is known) and (b) the variance reduction introduced by denoising the outcomes. Lemma 5 and Remark 2 in Appendix E.2 provide a proof and discussion of the variance reduction. In the following, we abbreviate $\tilde{R}^f_{i,t}(s,a)$ as $\tilde{R}^f_{i,t}$, with the context and action implied.

4 Algorithm Components

We now construct $\tilde{R}^f_{i,t}$, introduce the weighted least-squares loss function used by RoME to estimate parameters, and use nearest-neighbor regularization to pool information across users.

4.1 Debiased Machine Learning

To construct $\tilde{R}^f_{i,t}$, we fit a working model $f_{i,t}$ for the conditional mean reward via the DML framework, which requires three main conditions: (a) Neyman orthogonality, which makes the pseudo-outcome $\tilde{R}^f_{i,t}$ robust to misspecification of $f_{i,t}$, (b) sufficiently fast convergence of the working conditional mean model $f_{i,t}$, and (c) independence of $f_{i,t}$ and the observed results. (a) is ensured by the structure of the doubly robust differential reward. (b) requires us to introduce the following assumption, which states that the mean squared error (MSE) of the working conditional mean is asymptotically smaller than the inverse square root of the number of stages:

Assumption 3. *The working model, $f_{i,t}$, satisfies $\mathbb{E}_{p(s,\bar{a})}\left[\{r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a})\}^2\right] = o_P(k^{-1/2})$ and $\mathbb{E}_{p(s)}\left[\{r_{i,t}(s,0) - f_{i,t}(s,0)\}^2\right] = o_P(k^{-1/2})$. Further, $|f_{i,t}| \leq 2M$.*

Examples of supervised learning methods that converge at this rate (or faster) include parametric models of fixed dimension, local mean smoothers with small dimension (Wasserman, 2006, Chapter 5), and random forests and feedforward neural networks under certain regularity conditions (Biau, 2012; Farrell et al., 2021). Assumption 3 guarantees that RoME achieves better asymptotic variance than action-centered (AC) bandits Greenewald et al. (2017), resulting in the tighter regret bound in Section 5.2. Because our differential reward in (4) is robust to misspecification of $f_{i,t}$, our regret bound can be adapted to settings in which Assumption 3 does not hold, provided $f_{i,t}$ converges to a bounded function at the rate specified in Assumption 3; this setting would result in a weaker bound of the same asymptotic order. The final requirement of independence, (c), can be accomplished via sample-splitting techniques. Below, we present two such options.

Option 1: The first option exploits the fact that outcomes across different users are independent. Consequently, we can perform sample splitting as follows. **Step 1**: Randomly assign each user i to one of J folds. Let I_j denote the set containing user indexes for the j-th fold and let $I_j^{\complement}$ denote its complement. **Step 2**: For each fold, use a supervised learning algorithm to estimate the working model for $r_{i,t}(s,a)$ denoted $\hat{f}^{(j)}_{i,t}(s,a)$ using $I_j^{\complement}$. **Step 3**: Construct the pseudo-outcomes using (4).

Option 2: Because we do not expect an adversarial environment in mHealth, we may be willing to assume that the errors, $\epsilon_{i,t}$, are independent across time. In this case, we can adapt the procedure given above by randomly assigning data to folds at the level of (i,t). In this case, I_j, contains the (i,t) pairs assigned to fold j. This option is more powerful than option 1 in that it enables us to learn heterogeneity across users in the nuisance function, $f_{i,t}$, leading to greater variance reduction.

Lastly, we note two strategies that can reduce the computational demands of sample splitting. The first is to employ estimators that satisfy a leave-one-out stability condition, such as bagged estimators that use subsampling (Chen et al., 2022); these estimators eliminate the need to perform sample splitting. The second is to fit $f_{i,t}$ in an online fashion.

4.2 Estimation via Weighted Least Squares

Having formed $\tilde{R}^f_{i,t}(s,\bar{a})$, we now explain how to estimate the parameters, $\theta_{i,t} \in \mathbb{R}^p$, determining the differential reward, $\Delta_{i,t}(s,a)$. We assume that $\theta_{i,t}$ consists of three components:

Assumption 4. *$\theta_{i,t} = \theta^{shared} + \theta^{user}_i + \theta^{time}_t$ for all i,t.*

The parameter θ^{shared} is a fixed effect shared across all i,t. In contrast, θ^{user}_i and θ^{time}_t are random effects specific to a given user and time point, respectively. We place these parameters in a single column vector, θ, as $\text{vec}(\theta^{\text{shared}}, \theta^{\text{user}}_1, \ldots, \theta^{\text{user}}_K, \theta^{\text{time}}_1, \ldots, \theta^{\text{time}}_K)$, where K is the total number of stages. We then form a corresponding feature vector, $\phi_{i,t} = \phi(x_{i,t})$, where ϕ inserts $x_{i,t}$ into the positions corresponding to θ^{shared}, θ^{user}_i, and θ^{time}_t with zeros in all other locations; i.e., letting $p := \dim(x_{i,t})$, the feature vector $\phi(x_{i,t}) \in \mathbb{R}^{(2K+1)p}$ is defined as follows:

$$\phi(x_{i,t}) = [x_{i,t}^\top,\, 0^\top_{(i-1)p},\, x_{i,t}^\top,\, 0^\top_{(K-i+t-1)p},\, x_{i,t}^\top,\, 0^\top_{(K-t)p}]^\top. \tag{5}$$

We could then estimate $\theta_{i,t}$ via a weighted least squares (WLS) regression that minimizes the following objective function:

$$\ell_{\text{WLS}}(\theta) = \sum_{i=1}^{n} \sum_{t=1}^{K-i+1} \tilde{\sigma}^2_{i,t}\big(\tilde{R}^f_{i,t} - \phi_{i,t}^\top \theta\big)^2, \tag{6}$$

where $\tilde{\sigma}^2_{i,t} = \pi_{i,t}(0|S_{i,t}) \cdot (1 - \pi_{i,t}(0|S_{i,t}))$. However, $\ell_{\text{WLS}}(\theta)$ is over-parameterized and does not take advantage of available network information. The next section modifies $\ell_{\text{WLS}}(\theta)$ by adding regularization, producing the final loss function used in RoME.

4.3 Nearest-Neighbor Regularization

We assume access to network information relating neighboring users and time points. These networks are defined by graphs $G_{\text{user}} = (V_{\text{user}}, E_{\text{user}})$ and $G_{\text{time}} = (V_{\text{time}}, E_{\text{time}})$, where V denotes vertices and E, edges. In practice, these networks can be formed via known clusters (e.g., shared schools, companies, geographic locations) or similar covariates. In the case of G_{time}, a reasonable choice is to construct a graph of neighboring sequential time points; i.e., $t = 1 \leftrightarrow t = 2 \leftrightarrow t = 3 \ldots$. We assume the following bounds on the random effects:

Assumption 5. *There exists $D_{user}, D_{time}, B_{user}, B_{time}, B_{shared} \in \mathbb{R}^+$ such that $\|\theta^{user}_i - \theta^{user}_j\|_2^2 \leq D_{user}$, $\|\theta^{time}_i - \theta^{time}_j\|_2^2 \leq D_{time}$, $\|\theta^{user}_i\| \leq B_{user}$, and $\|\theta^{time}_i\| \leq B_{time}$ and $\|\theta^{shared}\| \leq B_{shared}$ for all i, j.*

The assumption 5 suggests that we may be able to improve our estimates of θ by regularizing neighboring values of θ^{user}_i and θ^{time}_i toward each other. This can be accomplished by using an L^2 Laplacian penalty. We illustrate the idea using G_{user}; the application to G_{time} is similar.

First, we form the incidence matrix, $Q \in \mathbb{R}^{K \times K}$, with the entry $Q_{v,e}$ corresponding to the v-th vertex (user) and the e-th edge. Denote the vertices of e as v_i and v_j with $i > j$. $Q_{v,e}$ is then equal to 1 if $v = v_i$, -1 if $v = v_j$, and 0 otherwise. The Laplacian matrix is then defined as $L = QQ^\top$. Similar to Yang et al. (2020), we can then form a *network cohesion* penalty across users as follows:

$$\text{tr}(\Theta^\top_{\text{user}} L_{\text{user}} \Theta_{\text{user}}) = \sum_{(i,j) \in E_{\text{user}}} \|\theta^{\text{user}}_i - \theta^{\text{user}}_j\|_2^2,$$

where $\Theta_{\text{user}} := (\theta^{\text{user}}_1, \ldots, \theta^{\text{user}}_K)^\top \in \mathbb{R}^{K \times p}$ and L_{user} is the Laplacian matrix for users. The penalty is small when θ^{user}_i and θ^{user}_j are close for connected users. We employ a shared regularization hyperparameter λ, for G_{user} and G_{time}, and include a standard L^2 penalty on θ with hyperparameter γ. The full penalization matrix $V_0 \in \mathbb{R}^{(2K+1)p \times (2K+1)p}$ is

$$V_0 = \text{diag}(\gamma I_p, \lambda L^{\text{user}}_{\otimes} + \gamma I_{Kp}, \lambda L^{\text{time}}_{\otimes} + \gamma I_{Kp}), \tag{7}$$

where $L^{\text{user}}_{\otimes} := L_{\text{user}} \otimes I_p$, the Kronecker product of L_{user} with I_p. We then adapt $\ell_{\text{WLS}}(\theta)$ to include a corresponding penalty term:

$$\ell(\theta) = \sum_{i=1}^{n} \sum_{t=1}^{K-i+1} \tilde{\sigma}^2_{i,t} \big(\tilde{R}^f_{i,t} - \phi^\top_{i,t}\theta\big)^2 + \theta^\top V_0 \theta. \tag{8}$$

In Section 5, we show that (8) leads to a Thompson sampling algorithm for action selection.

5 Thompson Sampling Algorithm

We now describe the Thompson sampling procedure and provide a high-probability regret bound.

5.1 Algorithm Description

The minimizer of the weighted regularized least squares loss in (8) can be expressed as $\hat{\theta} = V^{-1} b$, where

$$V = V_0 + \sum_{i=1}^{n} \sum_{t=1}^{K-i+1} \tilde{\sigma}^2_{i,t} \phi_{i,t} \phi^\top_{i,t}, \quad b = \sum_{i=1}^{n} \sum_{t=1}^{K-i+1} \tilde{R}^f_{i,t} \tilde{\sigma}^2_{i,t} \phi_{i,t}.$$

In analogy to Bayesian linear regression, the penalized and weighted Gram matrix, V, plays the role of a (scaled) posterior precision matrix under a multivariate Gaussian prior for θ. This motivates an algorithm in which we randomly perturb the point estimate, $\hat{\theta}$, scaling by an inverse square-root matrix, $V^{-1/2}$, times a mean-zero random deviate, η. The perturbed estimate, $\tilde{\theta}$, can be written as

$\tilde{\theta} = \hat{\theta} + \beta(\delta)V^{-1/2}\eta$, where letting $W > 0$ be the max number of neighbors a time point or user has, $\beta(\delta)$ is defined as

$$\beta(\delta) = v\left[2\log\left(\frac{\det(V)^{1/2}}{\det(V_0)^{1/2}\delta/2}\right)\right]^{1/2} + B, \tag{9}$$

$$B = \sqrt{\gamma}B_{\text{shared}} + \sqrt{\lambda WK}(\sqrt{D_{\text{user}}} + \sqrt{D_{\text{time}}}) + \sqrt{\gamma WK}(B_{\text{user}} + B_{\text{time}}). \tag{10}$$

The value v is defined in Corollary 4 and can be set to a sufficiently large constant. The hyperparameter δ determines the probability with which the regret bound holds. The random deviate, η, is drawn from a distribution, $\mathcal{D}^{TS}$, satisfying the technical conditions given in Definition 1 in Appendix D. In practice, we recommend setting $\mathcal{D}^{TS}$ to a multivariate Gaussian distribution or t-distribution to simplify calculations. Given $\tilde{\theta}$, we then compute $\bar{A}_{i,t} = \text{argmax}_{a\in[q]\setminus\{0\}}\, \phi_{i,t}^\top\tilde{\theta}$ and randomly select $A_{i,t} = 0$ with probability

$$\pi_{i,t}(0|\bar{A}_{i,t}) := \max\left[\pi_{\min}, \min\left\{\pi_{\max}, \Pr(\phi_{i,t}^\top\tilde{\theta} > 0)\right\}\right], \tag{11}$$

or $A_{i,t} = \bar{A}_{i,t}$ with probability $1 - \pi_{i,t}(0|\bar{A}_{i,t})$. The action selection is summarized in Algorithm 1.

Algorithm 1 Action selection

input V, b, $\pi_{\min}$, $\pi_{\max}$, $\phi_{i,t}$, $\beta(\delta)$
  $\hat{\theta} = V^{-1}b$
  Sample $\eta \sim \mathcal{D}^{TS}$
  $\tilde{\theta} = \hat{\theta} + \beta(\delta)V^{-1/2}\eta$, with $\beta(\delta)$ given in (9)
  $\bar{A}_{i,t} = \text{argmax}_{a\in[q]\setminus\{0\}}\, \phi_{i,t}^\top\tilde{\theta}$
  Calculate $\pi_{i,t}(0|\bar{A}_{i,t})$ according to (11)
  With probability $\pi_{i,t}(0|\bar{A}_{i,t})$, play action 0; otherwise, play action $\bar{A}_{i,t}$

Having detailed the action selection algorithm, we now summarize RoME in Algorithm 2. The algorithm includes an outer loop that iterates through the stages from $k = 1$ to $k = K$. Within each stage, we handle users sequentially, iteratively selecting actions according to Algorithm 1, observing rewards, and updating V and b accordingly.

Algorithm 2 RoME

  Initialize V_0 according to (7)
  $V \leftarrow V_0$
  $b \leftarrow 0$
  for $k = 1, \ldots, K$ **do**
    for $(i,t) = (1,k), (2,k-1), \ldots, (k,1)$ **do**
      Choose an arm according to Algorithm 1 using V and b
      Observe rewards, $R_{i,t}$
      Construct pseudo-rewards, $\tilde{R}_{i,t}^{f^{(m)}}$, as explained in Section 4.1
      Set $\phi = \phi(x_{i,t})$ as described in (5)
      Update weighted Gram matrix: $V \leftarrow V + \tilde{\sigma}_{i,t}^2\phi_{i,t}\phi_{i,t}^\top$
      Update feature-outcome products: $b \leftarrow b + \tilde{\sigma}_{i,t}^2\tilde{R}_{i,t}^f\phi_{i,t}$
    end for
  end for

5.2 Regret bound

We first define regret, $\mathbf{Regret}_K$, in terms of the following average across stages:

$$\sum_{k=1}^{K}\frac{1}{k}\sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}}\left\{\pi_{i,t}^\star(\bar{A}_{i,t}^*|S_{i,t})\cdot x(S_{i,t},\bar{A}_{i,t}^*)^\top\theta_{i,t}^* - \pi_{i,t}(\bar{A}_{i,t}|S_{i,t})\cdot x(S_{i,t},\bar{A}_{i,t})^\top\theta_{i,t}^*\right\}.$$

The inner sum calculates regret (the difference between the expected reward of the optimal action and the chosen action) weighted by the probability of the action, following Greenewald et al. (2017), for all individuals at stage k. We then average these values to account for the growing number of pairs (i, t) as the stages progress. The cumulative regret is the sum of these averaged values over all stages.

The rewards associated with the baseline action are excluded since $\Delta_{i,t}(S_{i,t}, 0) = 0$. $\mathbf{Regret}_K$ is a form of pseudo-regret because the randomness attributed to $\{\epsilon_{i,t}\}_{t=1}^K$ has been eliminated (Audibert et al., 2003). Before presenting the regret bound, we introduce one more assumption:

Assumption 6. *$\|x(s, a)\| \leq 1$ for all s and a. Also, there exists a known value $M \in \mathbb{R}^+$ bounding both $\|\theta_{i,t}\|$ and $|g_t(s)|$ for all i, t, and s. Further, $|f_{i,t}| \leq 2M$.*

Theorem 1. *Under Assumptions 1–6, with probability at least $1 - \delta$, $\mathbf{Regret}_K$ satisfies*

$$O\left(\left[\beta_K(\delta') + \gamma_K(\delta')\left\{1 + \frac{4}{p}\right\}\right] \cdot \log(K)\sqrt{cKp}\right)$$

where $\delta' = \delta/4K$ and $\min\{\pi(0|s), 1 - \pi(0|s)\} > 1/c$. The exact regret bound is given and both β_K and γ_K are defined in Appendix E.

Theorem 1's proof is in Appendix E. It resembles Abeille & Lazaric (2017), albeit with adjustments to address (a) the nonlinear baseline and (b) the increasing pool of users and time points. The nonlinear baseline requires deriving the new sub-Gaussian variance factor for the centered pseudo-reward and adapting classic contextual bandit theory (Abbasi-Yadkori et al., 2011) to our weighted least squares setting. The increasing pool of users and time points requires modifying the Abeille & Lazaric (2017) proof with a careful use of stages and the Cauchy–Schwarz inequality to decouple the weighted feature norms from inverse stage scaling, which allows us to apply standard techniques to the sum of norms and bound the sum of inverse stages by the harmonic number.

DML is particularly advantageous for $\beta_K(\delta')$ and $\gamma_K(\delta')$, which rely on the convergence rate of the model f towards the true mean differential reward r, as discussed further in Appendix E.2. Another interesting aspect of the regret bound is its dependence solely on the dimension, p, of the differential-reward model, rather than the dimension of the baseline reward. This characteristic allows us to achieve strong regret bounds even in cases where the baseline reward is highly complex.

6 Experiments

This section presents results from applying RoME in a simulated mHealth study with staggered recruitment and an off-policy comparison study for cardiac rehabilitation mHealth interventions. The simulations were implemented using Python and the results were generated using individual compute nodes with two 3.0 GHz Intel Xeon Gold 6154 processors and 180 GB of RAM. Case studies were implemented using R 4.2.2 and results were generated on a cluster composed of individual compute nodes with 2.10 GHz Intel Xeon Gold 6230 processors and 192 GB of RAM.

6.1 Competitor Comparison Simulation

In this section, we compare RoME to four competing methods in simulation. We implemented RoME using **Option 2** with a bagged ensemble of stochastic gradient trees (Gouk et al., 2019; Mastelini et al., 2021) trained online via the River library (Montiel et al., 2021). We programmed the linear algebra operations using SuiteSparse (Davis & Hu, 2011) to take advantage of the sparse nature of the Laplacian matrices and $\phi_{i,t}$ from Algorithm 2.

Our competing baselines are (a) Standard: Standard Thompson sampling for linear contextual bandits, (b) AC: the **A**ction-**C**entered contextual bandit algorithm (Greenewald et al., 2017), (c) IntelPooling: The intelligentpooling method of Tomkins et al. (2021) fixing the variance parameters close to their true values, and (d) Neural-Linear: a method that uses a pre-trained neural network to transform the feature space for the baseline reward (similar to the Neural Linear method of Riquelme et al. (2018)).

We compare these methods under three settings: Homogeneous Users, Heterogeneous Users, and Nonlinear. The first two involve a linear baseline model and time-homogeneous parameters, with the second setting having distinct user parameters. The third setting, designed to mirror the challenges of mHealth studies, includes a nonlinear baseline and both user- and time-specific parameters. For each

setting, we simulate 200 stages following the staged recruitment regime depicted in Figure 1, and we repeat the full 200-stage simulation 50 times. Appendix A.1 provides details on the setup and a link to our implementation.

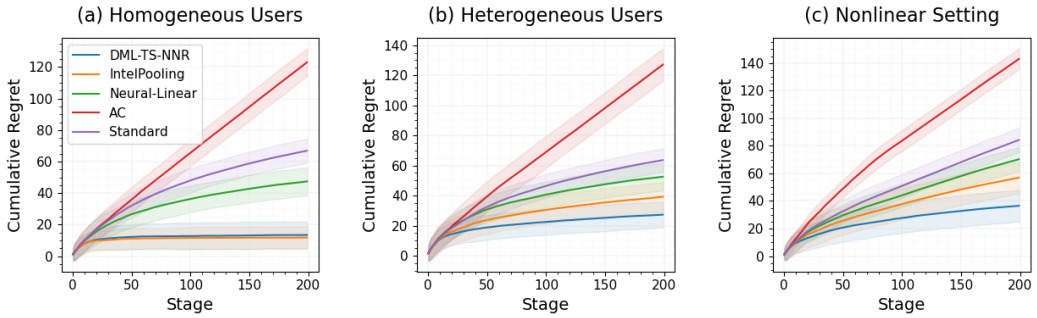

Figure 2: Cumulative regret in the (a) Homogeneous Users, (b) Heterogeneous Users, and (c) Nonlinear settings. RoME performs competitively in the first setting (the simplest), and it substantially outperforms the next-best method (IntelPooling) in the others.

Figure 2 displays the average cumulative regret for each method. RoME and IntelPooling have similar performance in the Homogeneous Users setting, but RoME excels in the Heterogeneous Users and Nonlinear settings, with significantly lower cumulative regret by stage 200. This is expected as RoME can utilize network information and model nonlinear baselines in these settings. However, in the Homogeneous Users setting, where no network information is available and a linear baseline is used, RoME does not outperform IntelPooling.

Appendices A.2–A.5 offer more results, including additional method comparisons, a rectangular data array simulation, hyperparameter sensitivity analyses, and pairwise statistical comparisons. The latter show that RoME outperformed each competitor in at least 48 of 50 repetitions in the Nonlinear setting, indicating even higher statistical confidence than Figure 2 suggests. RoME substantially outperforms the competing algorithms in the additional comparisons and is several orders of magnitude faster than would be required in a standard mHealth study, producing over 20,000 decisions in as little as 45 seconds (see Table 2 in Appendix A.2).

6.2 Valentine Study Analysis Results

In this section, we compare RoME to the above algorithms via off-policy evaluation using data from the Valentine Study (Jeganathan et al., 2022), a prospective, randomized-controlled, remotely administered trial designed to evaluate an mHealth intervention to supplement cardiac rehabilitation for low- and moderate-risk patients. In the analyzed dataset, participants were randomized to receive or not receive contextually tailored notifications promoting low-level physical activity and exercise throughout the day. The left of Figure 3 shows the estimated improvement in the average reward over the original constant randomization, averaged over stages (K = 120) and participants (N=108). We see that RoME achieved the highest average reward; its average performance (across bootstrap replications) is higher than the 75th percentile of all other methods.

To further analyze the improvement of RoME relative to the other methods, we test whether the proposed algorithm significantly improves cumulative rewards using paired t-tests with one-sided alternative hypotheses. The null hypothesis (H_0) for these tests is that the two algorithms being compared achieve the same average reward. The alternative hypothesis (H_1) is that the algorithm listed in the column achieves higher average rewards than the algorithm listed in the row. The right of Figure 3 displays the p-values obtained from these pairwise t-tests. The dark shade of the last column indicates that the proposed RoME algorithm achieves significantly higher rewards than the other five competing algorithms ($p \leq 0.01$ for all pairwise comparisons). The results imply that RoME would have achieved a 3.5% increase in step count relative to the constant randomization policy that was actually implemented in this study. This effect translates to an increase of 140 steps per day, given a baseline of 1,000 steps per hour and four one-hour measurement windows. See Appendix B for additional implementation details.

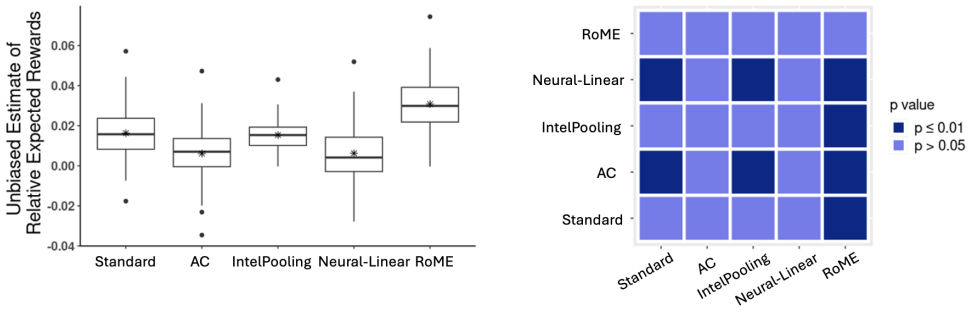

Figure 3: **(left)** Boxplot of unbiased estimates of the average per-trial reward for all five competing algorithms, relative to the reward obtained under the pre-specified Valentine randomization policy across 100 bootstrap samples. Within each box, the asterisk ($*$) indicates the mean value, while the mid-bar represents the median. **(right)** Heatmap of p-values from the pairwise paired t-tests.

We performed an additional off-policy comparison using data from the Intern Health Study (IHS) (NeCamp et al., 2020), as shown in Figure 11 in Appendix C.2. The results further demonstrate the competitive performance of the RoME algorithm, with the AC algorithm also showing comparable performance in this dataset. Further details on the analysis can be found in Appendix C.

7 Discussion

This paper introduces RoME, a robust mixed-effects contextual bandit algorithm for mHealth studies. RoME adapts DML and network cohesion penalties to dynamic settings, enabling researchers to efficiently pool information across users and over time. In addition to the methodological contribution, we also prove a high-probability regret bound in a challenging asymptotic regime that involves a growing pool of users and time points. Our implementation of RoME is several orders of magnitude faster than would be required in practice and could easily be adapted for use in future mHealth studies.

We see several promising directions for improving RoME in future work. One such direction might consider data-adaptive strategies for setting the hyperparameters or constructing a network in settings where the network is not known *a priori*. Another interesting direction is to address the long-term impacts of mHealth treatments, such as accumulating treatment fatigue. Future work could also consider computational improvements that would enable large-scale deployment of RoME, which would be especially relevant in nonclinical settings.

Acknowledgements

The authors gratefully acknowledge the contributions of the researchers, administrators, and participants involved in the Valentine (`https://clinicaltrials.gov/study/NCT04587882`) and Intern Health (`https://clinicaltrials.gov/study/NCT03972293`) studies. This work was supported in part by grant P50 DA054039 provided through the NIH and NIDDK and by grant R01 GM152549 through the NIH and NIGMS. Dr. Golbus receives funding from the NIH (L30HL143700, 1K23HL168220). Initially while working on this paper, Dr. Moreno was supported by Luminous Computing, Inc.

References

Abbasi-Yadkori, Y., Pál, D., and Szepesvári, C. Improved algorithms for linear stochastic bandits. *Advances in Neural Information Processing Systems*, 24, 2011.

Abeille, M. and Lazaric, A. Linear Thompson sampling revisited. *Electronic Journal of Statistics*, 11(2):5165 – 5197, 2017. doi: 10.1214/17-EJS1341SI. URL `https://doi.org/10.1214/17-EJS1341SI`.

Aouali, I., Kveton, B., and Katariya, S. Mixed-effect Thompson sampling. In *International Conference on Artificial Intelligence and Statistics*, pp. 2087–2115. PMLR, 2023.

Audibert, J.-Y., Biermann, A., and Long, P. Reinforcement learning with immediate rewards and linear hypotheses. *Algorithmica*, 37(4):263–296, 2003.

Bang, H. and Robins, J. M. Doubly robust estimation in missing data and causal inference models. *Biometrics*, 61(4):962–973, 2005.

Biau, G. Analysis of a random forests model. *The Journal of Machine Learning Research*, 13(1): 1063–1095, 2012.

Cesa-Bianchi, N., Gentile, C., and Zappella, G. A gang of bandits. *Advances in Neural Information Processing Systems*, 26, 2013.

Chen, Q., Syrgkanis, V., and Austern, M. Debiased machine learning without sample-splitting for stable estimators. *Advances in Neural Information Processing Systems*, 35:3096–3109, 2022.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., and Robins, J. Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, 21(1):C1–C68, 01 2018. ISSN 1368-4221. doi: 10.1111/ectj.12097. URL `https://doi.org/10.1111/ectj.12097`.

Cho, H., Wang, P., and Qu, A. Personalize treatment for longitudinal data using unspecified random-effects model. *Statistica Sinica*, pp. 187–206, 2017.

Davis, T. A. and Hu, Y. The University of Florida sparse matrix collection. *ACM Transactions on Mathematical Software (TOMS)*, 38(1):1–25, 2011.

Farrell, M. H., Liang, T., and Misra, S. Deep neural networks for estimation and inference. *Econometrica*, 89(1):181–213, 2021.

Friedman, L. M., Furberg, C. D., and DeMetsR., D. L. (eds.). *Machine Learning: An Artificial Intelligence Approach, Vol. I*. Springer, New York, NY, 4 edition, 2010.

Golbus, J. R., Gupta, K., Stevens, R., Jeganathan, V. S. E., Luff, E., Shi, J., Dempsey, W., Boyden, T., Mukherjee, B., Kohnstamm, S., et al. A randomized trial of a mobile health intervention to augment cardiac rehabilitation. *NPJ Digital Medicine*, 6(1):173, 2023.

Gouk, H., Pfahringer, B., and Frank, E. Stochastic gradient trees. In *Asian Conference on Machine Learning*, pp. 1094–1109. PMLR, 2019.

Greenewald, K., Tewari, A., Klasnja, P., and Murphy, S. Action centered contextual bandits. *Advances in Neural Information Processing Systems*, 30, 2017.

Hu, X., Qian, M., Cheng, B., and Cheung, Y. K. Personalized policy learning using longitudinal mobile health data. *Journal of the American Statistical Association*, 116(533):410–420, 2021.

Jeganathan, V. S., Golbus, J. R., Gupta, K., Luff, E., Dempsey, W., Boyden, T., Rubenfire, M., Mukherjee, B., Klasnja, P., Kheterpal, S., et al. Virtual application-supported environment to increase exercise (VALENTINE) during cardiac rehabilitation study: Rationale and design. *American Heart Journal*, 248:53–62, 2022.

Kennedy, E. H. Towards optimal doubly robust estimation of heterogeneous causal effects. *Electronic Journal of Statistics*, 17(2):3008–3049, 2023.

Kim, G.-S. and Paik, M. C. Contextual multi-armed bandit algorithm for semiparametric reward model. In *International Conference on Machine Learning*, pp. 3389–3397. PMLR, 2019.

Kim, W., Kim, G.-S., and Paik, M. C. Doubly robust Thompson sampling with linear payoffs. *Advances in Neural Information Processing Systems*, 34:15830–15840, 2021.

Kim, W., Lee, K., and Paik, M. C. Double doubly robust Thompson sampling for generalized linear contextual bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 8300–8307, 2023.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Krishnamurthy, A., Wu, Z. S., and Syrgkanis, V. Semiparametric contextual bandits. In *International Conference on Machine Learning*, pp. 2776–2785. PMLR, 2018.

Lee, K., Paik, M. C., Oh, M.-h., and Kim, G.-S. Mixed-effects contextual bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 13409–13417, 2024.

Li, L., Chu, W., Langford, J., and Schapire, R. A contextual-bandit approach to personalized news article recommendation. *WWW*, pp. 661–670, 2010.

Ma, H., Zhou, D., Liu, C., Lyu, M. R., and King, I. Recommender systems with social regularization. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, pp. 287–296, 2011.

Mastelini, S. M., de Leon Ferreira, A. C. P., et al. Using dynamical quantization to perform split attempts in online tree regressors. *Pattern Recognition Letters*, 145:37–42, 2021.

McPherson, M., Smith-Lovin, L., and Cook, J. M. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001.

Montiel, J., Halford, M., Mastelini, S. M., Bolmier, G., Sourty, R., Vaysse, R., Zouitine, A., Gomes, H. M., Read, J., Abdessalem, T., et al. River: Machine learning for streaming data in Python. 2021.

Nair, V. and Hinton, G. E. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning*, pp. 807–814, 2010.

NeCamp, T., Sen, S., Frank, E., Walton, M. A., Ionides, E. L., Fang, Y., Tewari, A., and Wu, Z. Assessing real-time moderation for developing adaptive mobile health interventions for medical interns: Micro-randomized trial. *Journal of Medical Internet Research*, 22(3):e15033, 2020.

Nie, X. and Wager, S. Quasi-oracle estimation of heterogeneous treatment effects. *Biometrika*, 108 (2):299–319, 2021.

Qian, T., Klasnja, P., and Murphy, S. A. Linear mixed models with endogenous covariates: Modeling sequential treatment effects with application to a mobile health study. *Statistical Science*, 35(3): 375, 2020.

Riquelme, C., Tucker, G., and Snoek, J. Deep Bayesian bandits showdown: An empirical comparison of bayesian deep networks for Thompson sampling. *arXiv preprint arXiv:1802.09127*, 2018.

Robinson, P. M. Root-n-consistent semiparametric regression. *Econometrica: Journal of the Econometric Society*, pp. 931–954, 1988.

Snoek, J., Rippel, O., Swersky, K., Kiros, R., Satish, N., Sundaram, N., Patwary, M., Prabhat, M., and Adams, R. Scalable bayesian optimization using deep neural networks. In *International Conference on Machine Learning*, pp. 2171–2180. PMLR, 2015.

Tomkins, S., Liao, P., Klasnja, P., and Murphy, S. Intelligentpooling: Practical Thompson sampling for mhealth. *Machine Learning*, 110(9):2685–2727, 2021.

Vaswani, S., Schmidt, M., and Lakshmanan, L. Horde of bandits using Gaussian Markov random fields. In *Artificial Intelligence and Statistics*, pp. 690–699. PMLR, 2017.

Wasserman, L. *All of Nonparametric Statistics*. Springer Science & Business Media, 2006.

Yang, K., Toni, L., and Dong, X. Laplacian-regularized graph bandits: Algorithms and theoretical analysis. In *International Conference on Artificial Intelligence and Statistics*, pp. 3133–3143. PMLR, 2020.

A Additional Details for Simulation Study

This appendix provides details regarding the setup of our simulation study, additional results not presented in the main paper, and results from additional simulations employing different competitors and/or simulation designs. We summarize the differences between the competing algorithms in Table 1.

	User-specific Parameters	Time-specific parameters	Pools Across Users	Enforces User Network Cohesion	Enforces Time Network Cohesion	Allows Nonlinear Baseline	Nonlinear Baseline Model
RoME	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$
RoME-BLM	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\times$
RoME-SU	$\times$	$\surd$	$\surd$	N/A	$\surd$	$\surd$	$\surd$
NNR-Linear	$\surd$	$\times$	$\surd$	$\surd$	N/A	$\times$	$\times$
IntelPooling	$\surd$	$\surd$	$\surd$	$\surd$	$\times$	$\times$	$\times$
Neural-Linear	$\surd$	$\times$	$\times$	N/A	N/A	$\surd$	$\surd$
Standard	$\surd$	$\times$	$\times$	N/A	N/A	$\times$	$\times$
AC	$\surd$	$\times$	$\times$	N/A	N/A	$\surd$	$\times$

Table 1: This table compares the methods tested in the simulation study based on the design components listed in the columns, with $\surd$ indicating that the method includes the component listed in the column, $\times$ indicating that it does not, and N/A indicating that the component is not applicable. This table can be used to investigate the effect of certain model components on cumulative regret; for example, comparing RoME and RoME-BLM helps us understand the impact of modeling the nonlinear baseline—the only column that differs between these two methods.

A.1 Setup Details

The code for the simulation study is fully containerized and publicly available at `https://github.com/eastonhuch/RoME`. The simulation study involves a generative model of the following form for user i at time t:

$$R_{it} = g(S_{it}) + x(S_{it}, A_{it})^\top \theta_{it} + \epsilon_{it}, \quad \epsilon_{it} \sim \mathcal{N}(0, 1)$$

Here $S_{it} = (s_1, s_2) \in \mathbb{R}^2$ is a context vector, with both dimensions $\overset{iid}{\sim} U(-1, 1)$. We set $x(s, a) = a\,(1, s_1, s_2)$. For simplicity, we set g to a time-homogeneous function. The specific nature of the function varies across the following three settings mentioned in Section 6.1:

- Homogeneous Users: Standard contextual bandit assumptions with a linear baseline and no user- or time-specific parameters. The linear baseline is $g(S_{it}) = 2 - 2s_1 + 3s_2$, and the causal parameter is $\theta_{it} = (1, 0.5, -4)$ such that the optimal action varies across the context space.
- Heterogeneous Users: Same as the above but each user's causal parameter has iid $\mathcal{N}(0, 1)$ noise added to it.
- Nonlinear: The general setting discussed in the paper with a nonlinear baseline, user-specific parameters, and time-specific parameters. The base causal parameter and user-specific parameters are the same as in the previous two settings. The nonlinear baseline and time-specific parameter are shown in Figure 4.

We assume that the data are observed via a staged recruitment scheme, as illustrated in Figure 1. For computational convenience, we update parameters and select actions in batches. If, for instance, we observe twenty users at a given stage, we update our estimates of the relevant causal parameters and select actions for all twenty users simultaneously. This strategy offers a slight computational advantage with limited implications in terms of statistical performance.

For simplicity, we assume that the nearest-neighbor network is known and set the relevant hyperparameters accordingly. We took care to set the hyperparameters such that RoME performs a similar

amount of shrinkage compared to other methods, especially IntelPooling, which effectively uses a separate penalty matrix for users and time. To accomplish this, we used the same penalty matrices for RoME and IntelPooling, effectively generalizing the scalar γ penalty discussed in the main paper. For the purposes of calculating the constant B given in Algorithm 2, we used a separate value of γ for both users and time and set them equal to the maximum eigenvalue of the corresponding penalty matrix. We use 5 neighbors within the DML methods and set the other hyperparameters as follows: $\lambda = 1$, and $\delta = 0.01$, and $v_k = 1$ (i.e., v_k is constant).

For the Neural-Linear method, we generate a 200×200 array of baseline rewards to train the neural network prior to running the bandit algorithm. Consequently, the results shown in the paper for Neural-Linear are better than would be observed in practice because we allowed the Neural-Linear method to leverage data that we did not make available to the other methods. This setup offers the computational benefit of not needing to update the neural network within bandit replications, which substantially reduces the necessary computation time.

Aside from the input features used, the Neural-Linear method has the same implementation as the Standard method. The Neural-Linear method uses the output from the last hidden layer of a neural network to model the baseline reward. However, we use the original features (the state vectors) to model the advantage function because the true advantage function is, in fact, linear in these features.

Our neural networks consisted of four hidden layers with 10, 20, 20, and 10 nodes, respectively. The first two employ the ReLU activation function (Nair & Hinton, 2010) while the latter two employ the hyperbolic tangent. We chose to use the hyperbolic tangent for the last two layers because Snoek et al. (2015) found that smooth activation functions such as the hyperbolic tangent were advantageous in their neural bandit algorithm. The loss function was the mean squared error between the neural networks' output and the baseline reward on a simulated data set. We trained our networks using the Adam optimizer (Kingma & Ba, 2014) with batch sizes of 200 for between 20 and 50 epochs. We simulated a separate validation data set to check that our model had converged and was generating accurate predictions.

Figure 5 compares the true baseline reward function in the nonlinear setting (left) to that estimated by the neural network (right). We see that the neural network produced an accurate approximation of the baseline reward, which helps explain the good performance of the Neural-Linear method relative to other baseline approaches, such as Standard.

We include the Neural-Linear method primarily to demonstrate that correctly modeling the baseline is not sufficient to ensure good performance. In the mHealth, algorithms should also be able to (1) efficiently pool data across users and time and (2) leverage network information. The Neural-Linear method satisfies neither of these criteria. Note that a neural network could be used to model the baseline rewards as part of our algorithm. Future work could consider allowing the differential rewards

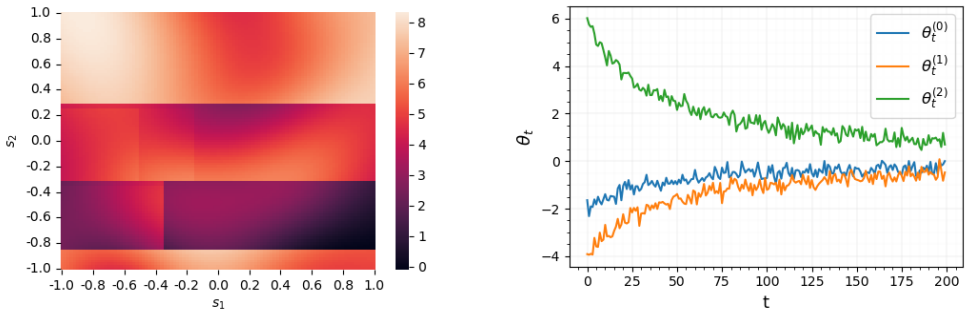

Figure 4: (left) The baseline reward function $g(S_{it})$ used in the simulation study. The proposed method allows this function to be a nonlinear function of the context vectors. The baseline was generated using a combination of recursive partitioning and by summing scaled, shifted, and rotated Gaussian densities. (right) The time-specific parameters used in the simulation study. These parameters cause the advantage function to vary over time. We set them such that the advantage function changes quickly at the beginning of the study then stabilizes.

themselves to also be complex nonlinear functions, which could be accomplished by combining our method with Neural-Linear. We leave the details to future work.

A.2 Additional Method Comparisons

Our results in the main paper indicate that RoME outperforms existing methods, especially in the complex longitudinal setting that we are targeting. These positive results beg the question: What aspects of RoME contribute most to its superior performance?

In this section, we describe an additional simulation study designed to answer this question. In it, we compare RoME (as implemented in the main paper) to three additional comparison methods: RoME-BLM, RoME-SU, and NNR-Linear. Each comparison isolates particular aspects of our algorithm to understand its contribution to RoME's performance.

RoME-**BLM** is a version of RoME that employs **B**agged **L**inear **M**odels as the baseline supervised learning algorithm. Unlike the method implemented in the main paper (which uses bagged stochastic gradient trees), this baseline model is misspecified in the nonlinear setting; i.e., it cannot accurately model the nonlinear baseline reward function. Consequently, this comparison will allow us to gauge the impact of the supervised learning method used in our method. In particular, it should allow us to validate whether the algorithm is robust to baseline misspecification as the theory suggests.

RoME-**SU** is a **S**ingle-**U**ser version of our algorithm. Rather than having a separate θ_i for each user, this method estimates a single advantage function that is shared across all users. It does, however, employ DML, time effects, and nearest-neighbor regularization (for the time effects). Consequently, this comparison will help us understand the impact of the user-specific parameters in our simulation.

NNR-Linear is similar to the existing network-cohesion bandit algorithms described in the main paper. It includes a distinct θ_i for each user and regularizes them toward each other using a Laplacian penalty. It does not, however, use DML or time effects. As a result, this comparison isolates the impact of these two factors and should provide evidence that RoME is more broadly applicable than existing network-cohesion bandit algorithms.

Figure 6 displays the cumulative regret as a function of the stage for these methods across the three settings used in the main paper. We see that all methods perform similarly in panel (a), the setting with homogeneous users. However, in panel (b) we see that the regret achieved by RoME-SU begins to resemble a straight upward-sloping line because it cannot appropriately model the user-specific advantage functions.

In panel (c), we observe a much wider spread of performance, with RoME and RoME-BLM performing similarly and substantially outperforming the other methods. The fact that RoME and RoME-BLM perform similarly indicates that our algorithm is, in fact, robust to misspecification as expected. The comparison to NNR-Linear reveals that the DML and time effects—the new components of our method compared to existing network-cohesion bandit algorithms—do contribute meaningfully to RoME's superior performance. We found in additional simulations studies not reported here that the gap between NNR-Linear and our method widens as the magnitude of the time effects increases.

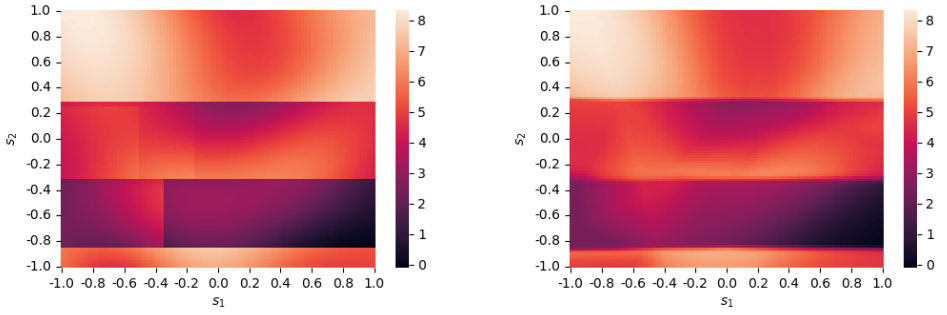

Figure 5: (left) The baseline reward function $g(S_{it})$ used in the simulation study compared to (right) the estimated baseline reward from our neural network in the nonlinear setting.

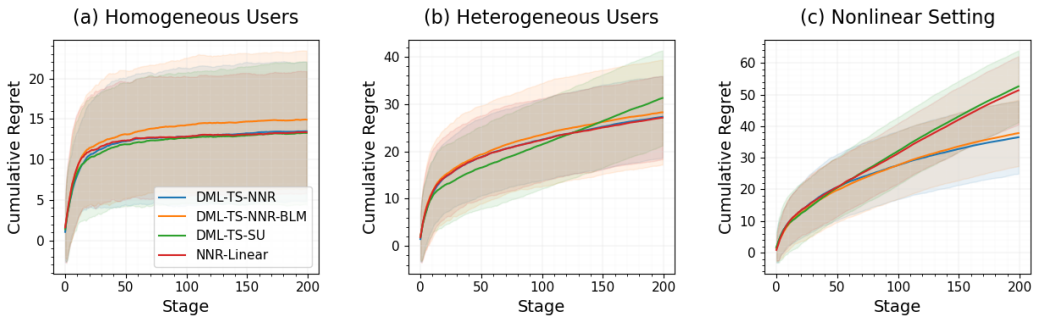

Figure 6: Cumulative regret in the (a) Homogeneous Users, (b) Heterogeneous Users, and (c) Nonlinear settings for the additional comparison methods. The fact that RoME outperforms RoME-SU demonstrates the importance of the user-specific parameters. The fact that RoME outperforms NNR-Linear shows that previous network cohesion approaches do not adequately address the (1) nonlinear nature of the baseline reward and (2) presence of time effects. The comparable performance of RoME-BLM to RoME highlights how our algorithm is robust to misspecification of the baseline reward model.

	Homogeneous Users	Heterogeneous Users	Nonlinear
RoME	345.2	373.4	362.2
RoME-BLM	45.6	49.4	52.2
RoME-SU	355.6	375.5	366.2
NNR-Linear	29.8	31.9	31.6
IntelPooling	46.3	43.9	46.8
Neural-Linear	0.7	0.7	0.9
Standard	0.2	0.2	0.3
AC	0.1	0.1	0.3

Table 2: Average computation time across all methods and settings in the main simulation.

Table 2 displays the average computation time for each method across all three settings. Each run of the simulation produces $200 * 201/2 = 20,100$ decisions. RoME and RoME-SU require the longest computation time at about six minutes. RoME-BLM requires only 45–53 seconds, indicating that the computational bottleneck for RoME is the ML model fitting. The required computation time for IntelPooling is similar to RoME-BLM at 43–47 seconds. NNR-Linear requires 29–32 seconds, and the remaining methods run in less than one second. Because most clinical mHealth studies require fewer than 1,000 decisions per day, the speed of RoME is orders of magnitude faster than is required in practice. However, large-scale commercial deployment of RoME may require computational adjustments and approximations.

A.3 Pairwise Comparisons

Table 3 shows pairwise comparisons between methods across the three settings. The individual cells indicate the percentage of repetitions (out of 50) in which the method listed in the row outperformed the method listed in the column. The asterisks indicate p-values below 0.05 from paired two-sided t-tests on the differences in final regret. The Avg column indicates the average pairwise win percentage.

RoME and RoME-BLM perform well across all three settings and the difference between them is statistically indistinguishable. These methods perform about equally well compared to IntelPooling and NNR-Linear in the first setting, but they perform better in the other two. In the Nonlinear setting in particular, RoME and RoME-BLM substantially outperform all other methods, and the pairwise differences are statistically significant at the 5% level.

Homogeneous Users

	1	2	3	4	5	6	7	8	**Avg**
1. RoME	-	58%	48%	52%	30%*	100%*	100%*	100%*	70%
2. RoME-BLM	42%	-	40%*	36%*	22%*	100%*	100%*	100%*	63%
3. RoME-SU	52%	60%*	-	46%	34%*	100%*	100%*	100%*	70%
4. NNR-Linear	48%	64%*	54%	-	32%*	100%*	100%*	100%*	71%
5. IntelPooling	70%*	78%*	66%*	68%*	-	100%*	100%*	100%*	83%
6. Neural-Linear	0%*	0%*	0%*	0%*	0%*	-	100%*	100%*	29%
7. Standard	0%*	0%*	0%*	0%*	0%*	0%*	-	100%*	14%
8. AC	0%*	0%*	0%*	0%*	0%*	0%*	0%*	-	0%

Heterogeneous Users

	1	2	3	4	5	6	7	8	**Avg**
1. RoME	-	46%	82%*	54%	100%*	100%*	100%*	100%*	83%
2. RoME-BLM	54%	-	70%*	48%	100%*	100%*	100%*	100%*	82%
3. RoME-SU	18%*	30%*	-	18%*	92%*	100%*	100%*	100%*	65%
4. NNR-Linear	46%	52%	82%*	-	100%*	100%*	100%*	100%*	83%
5. IntelPooling	0%*	0%*	8%*	0%*	-	90%*	100%*	100%*	43%
6. Neural-Linear	0%*	0%*	0%*	0%*	10%*	-	98%*	100%*	30%
7. Standard	0%*	0%*	0%*	0%*	0%*	2%*	-	100%*	15%
8. AC	0%*	0%*	0%*	0%*	0%*	0%*	0%*	-	0%

Nonlinear

	1	2	3	4	5	6	7	8	**Avg**
1. RoME	-	56%	100%*	96%*	98%*	100%*	100%*	100%*	93%
2. RoME-BLM	44%	-	94%*	98%*	100%*	100%*	100%*	100%*	91%
3. RoME-SU	0%*	6%*	-	38%	78%*	100%*	100%*	100%*	60%
4. NNR-Linear	4%*	2%*	62%	-	82%*	100%*	100%*	100%*	64%
5. IntelPooling	2%*	0%*	22%*	18%*	-	100%*	100%*	100%*	49%
6. Neural-Linear	0%*	0%*	0%*	0%*	0%*	-	98%*	100%*	28%
7. Standard	0%*	0%*	0%*	0%*	0%*	2%*	-	100%*	15%
8. AC	0%*	0%*	0%*	0%*	0%*	0%*	0%*	-	0%

Table 3: Pairwise comparisons between methods in the three settings of the main simulation. Each cell indicates the percent of repetitions (out of 50) in which the method listed in the row outperformed the method listed in the column in term of final regret. Asterisks indicate p-values below 0.05 from paired two-sided t-tests on the differences in final regret. RoME and RoME-BLM perform well in all three settings, and their final regret is statistically indistinguishable. They substantially outperform all other methods in the Nonlinear setting.

A.4 Simulation with Rectangular Data Array

The main simulation involves simulating data from a triangular data array. At the 200-th (final) stage, the algorithm has observed 200 rewards for user 1, 199 rewards for user 2, and so on.

In this section, we simulate actions and rewards under a rectangular array with 100 users and 100 time points. Although we still follow the staged recruitment regime depicted in Figure 1, at stage 100 we stop sampling actions and rewards for user 1; at stage 101 we stop sampling for user 2; and so on until we have sampled 100 time points for all 100 users. Aside from the shape of the data array, the setup is the same for this simulation as for the main simulation.

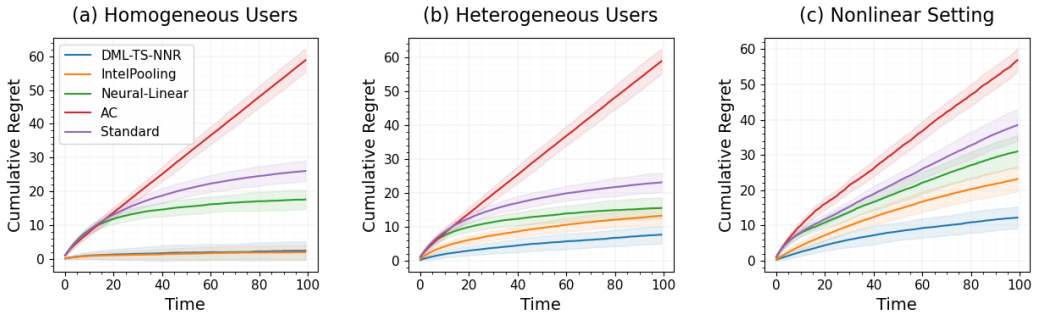

Figure 7: Cumulative regret in the (a) Homogeneous Users, (b) Heterogeneous Users, and (c) Nonlinear settings using a rectangular array of data in which we observe 100 time points for 100 users in a stagewise fashion as depicted in Figure 1. Similar to Figure 2, RoME is competitive in the first setting and substantially outperforms the competitors in the other settings.

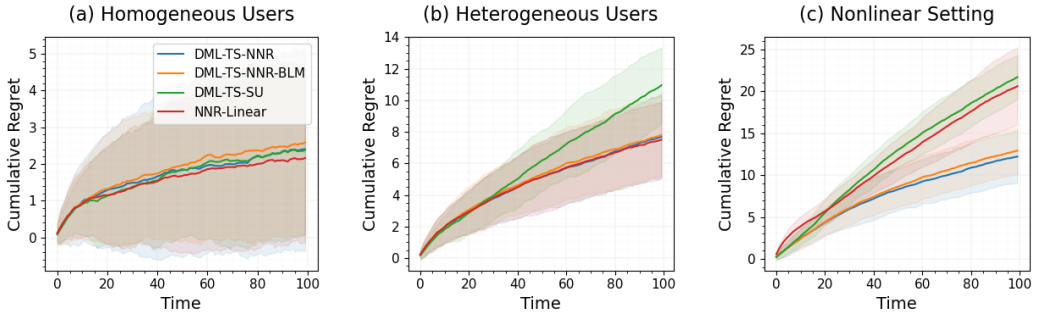

Figure 8: Cumulative regret in the (a) Homogeneous Users, (b) Heterogeneous Users, and (c) Nonlinear settings using a rectangular array of data in which we observe 100 time points for 100 users in a stagewise fashion as depicted in Figure 1. We observe the same performance ordering as in Figure 6, but here the relative differences are even larger.

The cumulative regret for these methods as a function of time—not stage—is shown in Figure 7. Qualitatively, the results are nearly identical to those from the main simulation (compare to Figure 2). The differences in final regret are even larger in this simulation than they were in the main simulation, presumably because this study exhibits greater variation in time effects due to the balance across time.

Figure 8 displays comparisons similar to those shown in Figure 6 but for the rectangular simulation. We again see that the results are qualitatively similar to the main simulation. The methods perform similarly in the Homogeneous Users setting. RoME-SU performs poorly in the Heterogeneous Users setting, but the other methods perform similarly. Finally, RoME and RoME-BLM substantially outperform the other methods in the nonlinear setting. Again, the differences in performance are even larger in this simulation than in the main simulation.

Table 4 displays results from pairwise method comparisons, similar to those shown in Table 3. Again, the results are qualitatively similar to but slightly more exaggerated than those from the main

Homogeneous Users

	1	2	3	4	5	6	7	8	**Avg**
1. RoME	-	56%	44%	42%	38%	100%*	100%*	100%*	69%
2. RoME-BLM	44%	-	44%	40%*	32%*	100%*	100%*	100%*	66%
3. RoME-SU	56%	56%	-	44%	36%	100%*	100%*	100%*	70%
4. NNR-Linear	58%	60%*	56%	-	52%	100%*	100%*	100%*	75%
5. IntelPooling	62%	68%*	64%	48%	-	100%*	100%*	100%*	77%
6. Neural-Linear	0%*	0%*	0%*	0%*	0%*	-	100%*	100%*	29%
7. Standard	0%*	0%*	0%*	0%*	0%*	0%*	-	100%*	14%
8. AC	0%*	0%*	0%*	0%*	0%*	0%*	0%*	-	0%

Heterogeneous Users

	1	2	3	4	5	6	7	8	**Avg**
1. RoME	-	60%	100%*	42%	100%*	100%*	100%*	100%*	86%
2. RoME-BLM	40%	-	100%*	34%	100%*	100%*	100%*	100%*	82%
3. RoME-SU	0%*	0%*	-	2%*	92%*	100%*	100%*	100%*	56%
4. NNR-Linear	58%	66%	98%*	-	100%*	100%*	100%*	100%*	89%
5. IntelPooling	0%*	0%*	8%*	0%*	-	98%*	100%*	100%*	44%
6. Neural-Linear	0%*	0%*	0%*	0%*	2%*	-	100%*	100%*	29%
7. Standard	0%*	0%*	0%*	0%*	0%*	0%*	-	100%*	14%
8. AC	0%*	0%*	0%*	0%*	0%*	0%*	0%*	-	0%

Nonlinear

	1	2	3	4	5	6	7	8	**Avg**
1. RoME	-	64%*	100%*	100%*	100%*	100%*	100%*	100%*	95%
2. RoME-BLM	36%*	-	100%*	100%*	100%*	100%*	100%*	100%*	91%
3. RoME-SU	0%*	0%*	-	28%*	74%*	100%*	100%*	100%*	57%
4. NNR-Linear	0%*	0%*	72%*	-	86%*	100%*	100%*	100%*	65%
5. IntelPooling	0%*	0%*	26%*	14%*	-	98%*	100%*	100%*	48%
6. Neural-Linear	0%*	0%*	0%*	0%*	2%*	-	100%*	100%*	29%
7. Standard	0%*	0%*	0%*	0%*	0%*	0%*	-	100%*	14%
8. AC	0%*	0%*	0%*	0%*	0%*	0%*	0%*	-	0%

Table 4: Pairwise comparisons between methods in the three settings of the simulation with a rectangular array of data. As in Table 3, each cell indicates the percent of repetitions (out of 50) in which the method listed in the row outperformed the method listed in the column in term of final regret. Asterisks indicate p-values below 0.05 from paired two-sided t-tests on the differences in final regret. The qualitative results are similar to those of Table 3. RoME and RoME-BLM perform well across all three settings and substantially outperform all other methods in the Nonlinear setting.

simulation. In particular, in this simulation study, RoME and RoME-BLM outperform all other methods in 100% of repetitions in the nonlinear setting.

A.5 Hyperparameter Sensitivity

In the simulation study, we chose hyperparameters for the fairest comparison possible, employing the same regularization parameters across methods. To assess robustness, we performed eight sensitivity analyses that alter hyperparameters for our methods while fixing those of the other methods; this approach gives the competing algorithms an advantage. Under these alternative hyperparameters, RoME and RoME-BLM still dramatically outperform the other methods in the Nonlinear Setting. We provide detailed results below.

Three analyses led to no meaningful changes to Figure 2 or Table 3. These involved rescaling (1) γ, (2) λ, and (3) and the bounds B, D by a factor of 10. The remaining simulations resulted in minor performance differences, especially in the Heterogeneous setting. In the following, we summarize these five analyses.

- Adding (low and medium) noise to the network: NNR-Linear slightly outperforms our methods in the Heterogeneous Setting, winning 50-60% of simulations (compared to 40-42% without noise) because NNR-Linear has access to a higher quality network.
- Increasing to 10 neighbors: RoME, RoME-BLM perform much better than RoME-SU (the Single-User ablation) in the Heterogeneous Setting, presumably because this change enforced stronger network cohesion among highly similar users.
- Rescaling σ by 10: NNR-Linear outperforms our methods in the Heterogeneous setting. NNR-Linear and IntelPooling outperform RoME-SU (but not RoME) in the Nonlinear Setting. Both occur because NNR-Linear and IntelPooling use the true value of σ.
- Setting δ to 0.05: NNR-Linear outperforms RoME-SU in the Nonlinear Setting, likely because this change led to insufficient exploration.

The results for the medium-noise sensitivity analysis are displayed in Table 5. Results for the remaining sensitivity analyses are available with our code.

B Additional Details for Valentine Study

Personalizing treatment delivery in mobile health is a common application for online learning algorithms. We focus here on the Valentine study, a prospective, randomized-controlled, remotely administered trial designed to evaluate an mHealth intervention to supplement cardiac rehabilitation for low- and moderate-risk patients (Jeganathan et al., 2022; Golbus et al., 2023). We aim to use smartwatch data (Apple Watch and Fitbit) obtained from the Valentine study to learn the optimal timing of notification delivery given the users' current context.

B.1 Data from the Valentine Study

Prior to the start of the trial, baseline data was collected from each of the participants (e.g., age, gender, baseline activity level, and health information). During the study, participants are randomized to either receive a notification ($A_t = 1$) or not ($A_t = 0$) at each of 4 daily time points (morning, lunchtime, mid-afternoon, evening), with probability 0.25. Contextual information was collected frequently (e.g., number of messages sent in prior week, step count variability in prior week, and pre-decision point step-counts).

Since the goal of the Valentine study is to increase participants' activity levels, we define the reward, R_t, as the step count for the 60 minutes following a decision point (log-transformed to eliminate skew). Our application also uses a subset of the baseline and contextual data; this subset contains the variables with the strongest association with the reward. Table 6 shows the features available to the bandit in the Valentine study data set.

For baseline variables, we use the participant's device model (Z_1, Fitbit coded as 1), the participant's step count variability in the prior week (Z_2), and a measure of the participant's pre-trial activity level based on an intake survey (Z_3, with larger values corresponding to higher activity levels).

Homogeneous Users

	1	2	3	4	5	6	7	8	**Avg**
1. RoME	-	48%	54%	52%	52%	100%*	100%*	100%*	72%
2. RoME-BLM	52%	-	58%	44%	50%	100%*	100%*	100%*	72%
3. RoME-SU	46%	42%	-	38%	46%	100%*	100%*	100%*	67%
4. NNR-Linear	48%	56%	62%	-	56%	100%*	100%*	100%*	75%
5. IntelPooling	48%	50%	54%	44%	-	100%*	100%*	100%*	71%
6. Neural-Linear	0%*	0%*	0%*	0%*	0%*	-	100%*	100%*	29%
7. Standard	0%*	0%*	0%*	0%*	0%*	0%*	-	100%*	14%
8. AC	0%*	0%*	0%*	0%*	0%*	0%*	0%*	-	0%

Heterogeneous Users

	1	2	3	4	5	6	7	8	**Avg**
1. RoME	-	50%	74%*	48%	100%*	100%*	100%*	100%*	82%
2. RoME-BLM	50%	-	82%*	40%	96%*	100%*	100%*	100%*	81%
3. RoME-SU	26%*	18%*	-	18%*	92%*	100%*	100%*	100%*	65%
4. NNR-Linear	52%	60%	82%*	-	100%*	100%*	100%*	100%*	85%
5. IntelPooling	0%*	4%*	8%*	0%*	-	94%*	100%*	100%*	44%
6. Neural-Linear	0%*	0%*	0%*	0%*	6%*	-	96%*	100%*	29%
7. Standard	0%*	0%*	0%*	0%*	0%*	4%*	-	100%*	15%
8. AC	0%*	0%*	0%*	0%*	0%*	0%*	0%*	-	0%

Nonlinear

	1	2	3	4	5	6	7	8	**Avg**
1. RoME	-	68%*	100%*	100%*	100%*	100%*	100%*	100%*	95%
2. RoME-BLM	32%*	-	98%*	98%*	100%*	100%*	100%*	100%*	90%
3. RoME-SU	0%*	2%*	-	44%	74%*	100%*	100%*	100%*	60%
4. NNR-Linear	0%*	2%*	56%	-	84%*	100%*	100%*	100%*	63%
5. IntelPooling	0%*	0%*	26%*	16%*	-	98%*	100%*	100%*	49%
6. Neural-Linear	0%*	0%*	0%*	0%*	2%*	-	100%*	100%*	29%
7. Standard	0%*	0%*	0%*	0%*	0%*	0%*	-	100%*	14%
8. AC	0%*	0%*	0%*	0%*	0%*	0%*	0%*	-	0%

Table 5: Pairwise comparisons between methods in a sensitivity analysis in which we add noise to the network. As in Table 3, each cell indicates the percent of repetitions (out of 50) in which the method listed in the row outperformed the method listed in the column in term of final regret. Asterisks indicate p-values below 0.05 from paired two-sided t-tests on the differences in final regret.

Feature	Description	Interaction	Baseline
Phase II	1 if in Phase II, 0 o.w.	$\surd$	$\surd$
Phase III	1 if in Phase II, 0 o.w.	$\surd$	$\surd$
Steps in prior 30 minutes	log transformed	$\surd$	$\surd$
Pre-trial average daily steps	log transformed	$\times$	$\surd$
Device	1 if Fitbit, 0 o.w.	$\times$	$\surd$
Prior week step count variability	SD of the rewards in the previous week	$\times$	$\surd$

Table 6: List of features available to the bandit in the Valentine study. The features available to model the action interaction (effect of sending an anti-sedentary message) and to model the baseline (reward under no action) are denoted via a "$\surd$" in the corresponding column, otherwise $\times$.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
ParticipantIdentifier	107	264	2.464	8.09	<2e-16	***
Week	25	17	0.683	2.24	4e-04	***
Residuals	2265	690	0.305			

Table 7: ANOVA analysis of the pseudo-outcomes in the Valentine study. The small p-values constitute strong evidence that the treatment effects differ by participant and over time.

At every decision point, before selecting an action, the learner sees two state variables: the participant's previous 30-minute step count (S_1, log-transformed) and the participant's phase of cardiac rehabilitation (S_2, dummy coded). The cardiac rehabilitation phase is defined based on a participant's time in the study: month 1 represents Phase I, month 2-4 represents Phase II, and month 5-6 represents Phase III.

B.2 Justification of Assumption 4

The motivation for Assumption 4 arises from an exploratory analysis we performed using data from the Valentine study. We constructed the pseudo-reward for each observation as suggested by Equation (4) and then conducted an ANOVA test. The results show clear heterogeneity between individuals and across time, motivating the adoption of user- and time-specific random effects. The results of the ANOVA test are displayed in Table 7. Figure 9 shows the shape of the heterogeneity in the treatment effects over time.

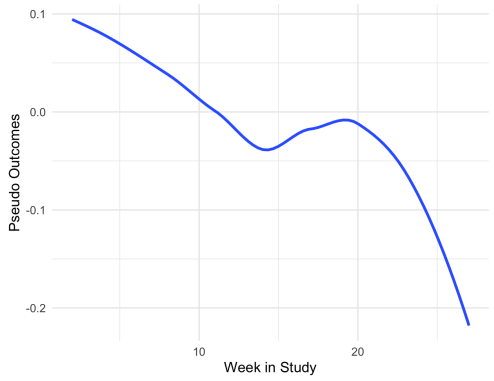

Figure 9: The time heterogeneity in the pseudo-outcomes. We calculated the pseudo-outcomes using 4, then averaged them across participants and plotted them over time. This exploratory analysis shows evidence that the causal effects vary substantially over time.

B.3 Evaluation

The Valentine study collected the sensor-based features at 4 decision points per day for each study participant. The reward for each message was defined to be $\log(0.5 + x)$, where x is the step count of the participant in the 60 minutes following the notification. As noted in the introduction, the baseline reward, i.e. the step count of a subject when no message is sent, not only depends on the state in a complex way but is likely dependent on a large number of time-varying observed variables. Both of these characteristics (complex, time-varying baseline reward function) suggest using our proposed approach.

We generated 100 bootstrap samples and ran our contextual bandit on them, considering the binary action of whether or not to send a message at a given decision point based on the contextual variables S_1 and S_2. Each user is considered independently and with a cohesion network, for maximum personalization and independence of results. To guarantee that messages have a positive probability of being sent, we only sample the observations with notification randomization probability between 0.01 and 0.99. In the case of the algorithm utilizing NNR, we chose four baseline characteristics (gender, age, device, and baseline average daily steps) to establish a measure of "distance" between

users. For this analysis, the value of k representing the number of nearest neighbors was set to 5. To utilize bootstrap sampling, we train the Neural-Linear method's neural network using out-of-bag samples. The neural network architecture comprises a single hidden layer with two hidden nodes. The input contains both the baseline characteristics and the contextual variables and the activation function applied here is the *softplus* function, defined as $\text{softplus}(x) = \log(1 + \exp(x))$.

We performed an offline evaluation of the contextual bandit algorithms using an inverse propensity score (IPS) version of the method from Li et al. (2010), where the sequence of states, actions, and rewards in the data are used to form a near-unbiased estimate of the average expected reward achieved by each algorithm, averaging over all users.

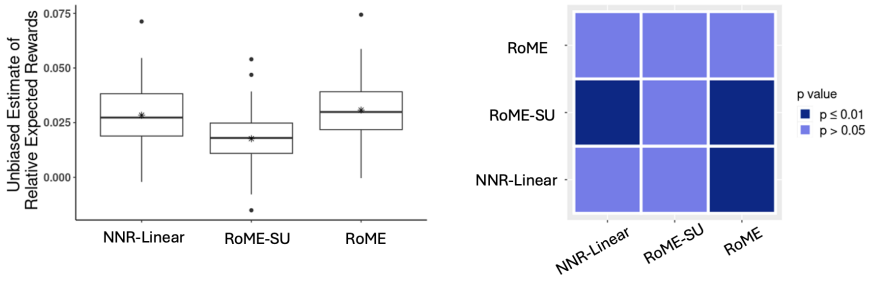

Figure 10: **(left)** Unbiased estimates of the average per-trial reward for all three ablation algorithms, relative to the reward obtained under the pre-specified Valentine Study randomization policy across 20 multiple-imputed data sets. And **(right)** p-values from the pairwise paired t-tests.

B.4 An ablation study

We further investigate the primary contributors to the algorithm's performance, we conducted a parallel ablation study using real-world data analysis. In this study, we compared RoME (as detailed in the main paper) with two additional methods for reference: RoME-SU and NNR-Linear. Each comparison focuses on specific elements of our algorithm to discern their impact on RoME's overall performance.

The differences between these methods are illustrated in Figure. 10. In this particular dataset, NNR plays a more significant role in driving the overall superior performance of our algorithm.

B.5 Inverse Propensity Score (IPS) offline evaluation

In the implemented Valentine study, the treatment was randomized with a constant probability $p_t = 0.25$ at each time t. To conduct off-policy evaluation using our proposed algorithm and the competing variations of the TS algorithm, we outline the IPS estimator for an unbiased estimate of the per-trial expected reward based on what has been studied in Li et al. (2010).

Given the logged data $\mathcal{D} = \{s_t = s_t, A_t = a_t, R_t = r_t\}_{t=1}^T$ collected under the policy $\mathbf{p} = \{p_t\}_{t=1}^T$, and the treatment policy being evaluated $\pi = \{\pi_t\}_{t=1}^T$, the objective of this offline estimator is to reweight the observed reward sequence $\{R_t\}_{t=1}^T$ to assign varying importance to actions based on the propensities of both the original and new policies in selecting them.

Lemma 1 (Unbiasedness of the IPS estimator)**.** *Assuming the positivity assumption in logging, which states that for any given s and a, if $p_t(a|s) > 0$, then we also have $\pi_t(a|s) > 0$, we can obtain an unbiased per-trial expected reward using the following IPS estimator:*

$$\hat{R}_{IPS} = \frac{1}{T} \sum_{t=1}^{T} \frac{\pi_t(a_t|s_t)}{p_t(a_t|s_t)} r_t \tag{12}$$

As mentioned in the previous section, we restrict our sampling to observations with notification randomization probabilities ranging from 0.01 to 0.99. This selection criterion ensures the satisfaction of the positivity assumption. The proof essentially follows from definition, we have:

Proof.

$$\begin{aligned}
\mathbb{E}[R_{\text{IPS}}] &= \mathbb{E}_{\mathbf{p}}\left[\frac{1}{T}\sum_{t=1}^{T}\frac{\pi_t(a_t|s_t)}{p_t(a_t|s_t)}R_t(a_t,s_t)\right] \\
&= \frac{1}{T}\sum_{t=1}^{T}\frac{\pi_t(a_t|s_t)}{p_t(a_t|s_t)}R_t(a_t,s_t)\times p_t(a_t|s_t) \\
&= \frac{1}{T}\sum_{t=1}^{T}\pi_t(a_t|s_t)R_t(a_t,s_t) \\
&= \mathbb{E}_{\pi}\left[\frac{1}{T}\sum_{t=1}^{T}R_t(a_t,s_t)\right]
\end{aligned}$$

□

To address the instability issue caused by reweighting in some cases, we use a Self-Normalized Inverse Propensity Score (SNIPS) estimator. This estimator scales the results by the empirical mean of the importance weights, and still maintains the property of unbiasedness.

$$\hat{R}_{\text{SNIPS}} = \frac{\hat{R}_{\text{IPS}}}{\frac{1}{T}\sum_{t=1}^{T}\frac{\pi_t(a_t|s_t)}{p_t(a_t|s_t)}} = \frac{\sum_{t=1}^{T}\frac{\pi_t(a_t|s_t)}{p_t(a_t|s_t)}r_t}{\sum_{t=1}^{T}\frac{\pi_t(a_t|s_t)}{p_t(a_t|s_t)}} \tag{13}$$

C Additional Details for the Intern Health Study (IHS)

To further enhance the competitive performance of our proposed RoME algorithm, we performed an additional comparative analysis using a real-world data set from the Intern Health Study (IHS) (NeCamp et al., 2020). This micro-randomized trial investigated the use of mHealth interventions aimed at improving the behavior and mental health of individuals in stressful work environments. The estimates obtained represent the improvement in average reward relative to the original constant randomization, averaging across stages (K = 30) and participants (N = 1553). The available IHS data consist of 20 multiple-imputed data sets. We apply the algorithms to each imputed data set and perform a comparative analysis of the competing algorithms. The results presented in Figure 11 shows our proposed RoME algorithm achieved significantly higher rewards than the other three competing ones and demonstrated performance comparable to the AC algorithm. These findings further support the advantages of our proposed algorithm.

C.1 Data from the IHS

Prior to the start of the trial, baseline data was collected on each of the participants (e.g., institution, specialty, gender, baseline activity level, and health information). During the study, participants are randomized to either receive a notification ($A_t = 1$) or not ($A_t = 0$) every day, with probability $3/8$. Contextual information was collected frequently (e.g., step count in prior five days, and current day in study).

We define the reward, R_t, as the step count on the following day (cubic root). Our application also uses a subset of the baseline and contextual data; this subset contains the variables with the strongest association to the reward. Table 8 shows the features available to the bandit in the IHS data set.

At every decision point, before selecting an action, the learner sees two state variables: the participant's previous 5-day average daily step count (S_1, cubic root) and the participant's day in the study (S_2, an integer from 1 to 30).

C.2 Evaluation

We run our contextual bandit on the IHS data, considering the binary action of whether or not to send a message at a given decision point based on the contextual variables S_1 and S_2. Each user is considered independently and with a cohesion network, for maximum personalization and

Feature	Description	Interaction	Baseline
Day in study	an integer from 1 to 30	$\surd$	$\surd$
Average daily steps in prior five days	cubic root	$\surd$	$\surd$
Average daily sleep in prior five days	cubic root	$\times$	$\surd$
Average daily mood in prior five days	a Likert scale from $1-10$	$\times$	$\surd$
Pre-intern average daily steps	cubic root	$\times$	$\surd$
Pre-intern average daily sleep	cubic root	$\times$	$\surd$
Pre-intern average daily mood	a Likert scale from $1-10$	$\times$	$\surd$
Sex	Gender	$\times$	$\surd$
Week category	The theme of messages in a specific week (mood, sleep, activity, or none)	$\times$	$\surd$
PHQ score	PHQ total score	$\times$	$\surd$
Early family environment	higher score indicates higher level of adverse experience	$\times$	$\surd$
Personal history of depression		$\times$	$\surd$
Neuroticism (Emotional experience)	higher score indicates higher level of neuroticism	$\times$	$\surd$

Table 8: List of features available to the bandit in the IHS. The features available to model the action interaction (effect of sending a mobile prompt) and to model the baseline (reward under no action) are denoted via a "$\surd$" in the corresponding column, otherwise $\times$.

independence of results. To guarantee that messages have a positive probability of being sent, we only sample the observations with notification randomization probability between 0.01 and 0.99. For the algorithm employing NNR, we defined participants in the same institution as their own "neighbors". This definition enables the flexibility for the value of k, representing the number of nearest neighbors, to vary for each participant based on their specific institutional context. Furthermore, in our study setting, we assume that individuals from the same institution enter the study simultaneously as a group. Due to the limited access to prior data, we are unable to build the neural linear models as in the Valentine Study.

We utilized 20 multiple-imputed data sets and performed an offline evaluation of the contextual bandit algorithms on each data set. The result is presented below in Figure 11. Similar to Section B.4, here we also compared RoME (as detailed in the main paper) with RoME-SU and NNR-Linear. The differences between these methods are illustrated in Figure. 12. In this specific dataset, both NNR and DML exhibit comparable performance individually, but their combined effect significantly enhances the overall performance of the algorithm.

D Additional Details for Algorithm 1

This appendix briefly discusses two details of Algorithm 1. The first is efficient computation of V^{-1}. Because V is a fairly large matrix, a full matrix inversion is expensive. Fortunately, however, we can dramatically reduce the necessary computational requirements because each additional (i, t) involves a rank-one perturbation to V. Consequently, we can apply the Sherman–Morrison formula to speed up computations. We leveraged this trick in our implementation of RoME for the simulation study by using efficient rank-one updates available in the SuiteSparse library (Davis & Hu, 2011).

The second detail is the requirements for the distribution, $\mathcal{D}^{TS}$:

Definition 1. *$\mathcal{D}^{TS}$ is a multivariate distribution on $\mathbb{R}^d$ absolutely continuous with respect to Lebesgue measure which satisfies: 1. (anti-concentration) that there exists a strictly positive probability p such that for any $u \in \mathbb{R}^d$ with $\|u\| = 1$, $\mathbb{P}(u^\top \eta \geq 1) \geq p$; and 2. (concentration) there exists c, c' positive constants such that $\forall \delta \in (0, 1)$, $P(\|\eta\| \leq \sqrt{cd \log(c'd/\delta)}) \geq 1 - \delta$.*

While a Gaussian prior satisfies Definition 1, this approach allows us to move beyond Bayesian posteriors to generic randomized policies. As discussed in the main paper, we recommend setting

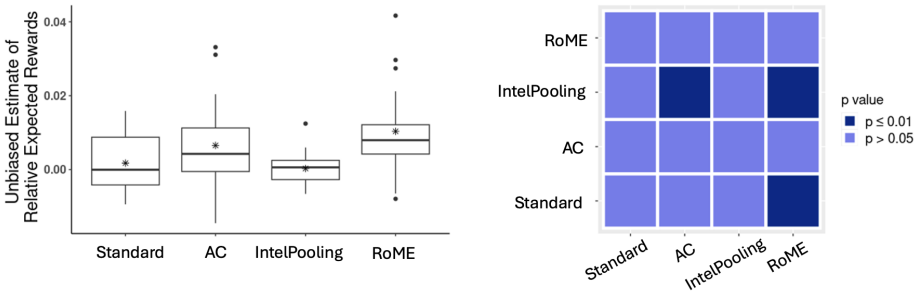

Figure 11: **(left)** Unbiased estimates of the average per-trial reward for all four competing algorithms, relative to the reward obtained under the pre-specified Intern Health Study randomization policy across 20 multiple-imputed data sets. And **(right)** p-values from the pairwise paired t-tests. The dark shade in the last column indicates that the proposed RoME algorithm achieved significantly higher rewards than the other three competing algorithms while demonstrating comparable performance to the AC algorithm.

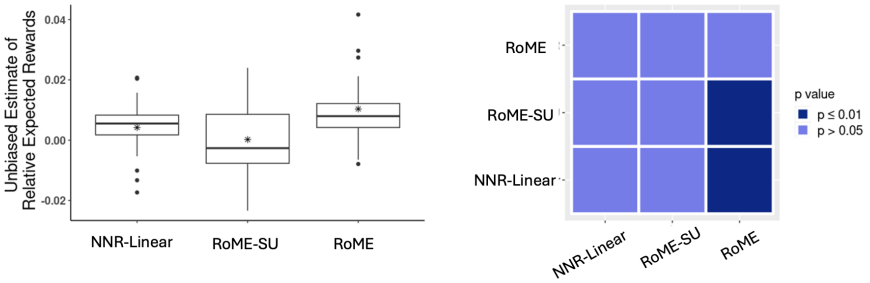

Figure 12: **(left)** Unbiased estimates of the average per-trial reward for all three ablation algorithms, relative to the reward obtained under the pre-specified Intern Health Study randomization policy across 20 multiple-imputed data sets. And **(right)** p-values from the pairwise paired t-tests.

$\mathcal{D}^{TS}$ to a multivariate Gaussian distribution or multivariate t-distribution. These choices simplify computations because the probability computation in (11) simplifies to an evaluation of the CDF of either (a) a univariate Gaussian or (b) a univariate t-distribution. The mean and variance of the corresponding univariate distribution can easily be worked out using the moments of a linear combination of random variables.

E Regret Bound

E.1 Double Robustness of Pseudo-Reward

Going forward, we use the notation $\Delta^f_{i,t}(s, \bar{a}) := f_{i,t}(s, \bar{a}) - f_{i,t}(s, 0)$ to denote the prediction of the differential reward.

Lemma 2. *If either $p_{i,t} = \pi_{i,t}$ or $f_{i,t} = r_{i,t}$, then*

$$\mathbb{E}\left[\tilde{R}^f_{i,t} | s, \bar{a}\right] = \Delta_{i,t}(s, \bar{a}).$$

That is, the pseudo-reward is an unbiased estimator of the true differential reward.

Proof. Recall that

$$\tilde{R}^f_{i,t} = \frac{R_{it} - f_{i,t}(s, A_{i,t})}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)} + \Delta^f_{i,t}(s, \bar{a})$$

Case I: π's are correctly specified

Then

$$\begin{aligned}\mathbb{E}\left[\frac{R_{it}}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}\middle| s, \bar{a}\right] &= r_{i,t}(s, \bar{a}) - r_{i,t}(s, 0) \\ &= \Delta_{i,t}(s, \bar{a})\end{aligned}$$

and

$$\begin{aligned}\mathbb{E}\left[\frac{f_{i,t}(s, A_{i,t})}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}\middle| s, \bar{a}\right] &= f_{i,t}(s, \bar{a}) - f_{i,t}(s, 0) \\ &= \Delta^f_{i,t}(s, \bar{a})\end{aligned}$$

so that

$$\begin{aligned}\mathbb{E}\left[\frac{R_{it} - f_{i,t}(s, A_{i,t})}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)} + \Delta^f_{i,t}(s, \bar{a})\middle| s, \bar{a}\right] &= \Delta_{i,t}(s, \bar{a}) - \Delta^f_{i,t}(s, \bar{a}) + \Delta^f_{i,t}(s, \bar{a}) \\ &= \Delta_{i,t}(s, \bar{a})\end{aligned}$$

Case II: f correctly specified

$$\mathbb{E}\left[\frac{R_{it}}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}\middle| s, \bar{a}\right] = \frac{1 - p_{i,t}(0|s)}{1 - \pi_{i,t}(0|s)} r_{i,t}(s, \bar{a}) - \frac{p_{i,t}(0|s)}{\pi_{i,t}(0|s)} r_{i,t}(s, 0)$$

and

$$\begin{aligned}\mathbb{E}\left[\frac{f_{i,t}(s, A_{i,t})}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}\middle| s, \bar{a}\right] &= \frac{1 - p_{i,t}(0|s)}{1 - \pi_{i,t}(0|s)} f_{i,t}(s, \bar{a}) - \frac{p_{i,t}(0|s)}{\pi_{i,t}(0|s)} f_{i,t}(s, \bar{0}) \\ &= \frac{1 - p_{i,t}(0|s)}{1 - \pi_{i,t}(0|s)} r_{i,t}(s, \bar{a}) - \frac{p_{i,t}(0|s)}{\pi_{i,t}(0|s)} r_{i,t}(s, \bar{0})\end{aligned}$$

and

$$\mathbb{E}\left[\Delta^f_{i,t}(s, \bar{a})\middle| s, \bar{a}\right] = \Delta_{i,t}(s, \bar{a})$$

$$\mathbb{E}\left[\frac{R_{it} - f_{i,t}(s, \bar{A}_{i,t})}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)} + \Delta^f_{i,t}(s, \bar{a})\middle| s, \bar{a}\right] = \Delta_{i,t}(s, \bar{a})$$

□

E.2 Preliminaries

Lemma 3. *Let X be a mean-zero sub-Gaussian random variable with variance factor v^2 and Y be a bounded random variable such that $|Y| \leq B$ for some $0 \leq B < \infty$. Then XY is sub-Gaussian with variance factor v^2B^2.*

Proof. Recall that X being mean-zero sub-Gaussian means that

$$P(|X| \geq t) \leq 2\exp\left(-\frac{t^2}{2v^2}\right).$$

Now note that

$$|XY| \leq |X|B$$

so that if $|XY| > t$, then $|X|B > t$. Thus by monotonicity

$$\begin{aligned} P(|XY| \geq t) &\leq P(|X|B \geq t) \\ &= P\left(|X| \geq \frac{t}{B}\right) \\ &\leq 2\exp\left(-\frac{t^2}{2B^2v^2}\right) \end{aligned}$$

as desired. □

Lemma 4. *If X, Y are sub-Gaussian with variance factors v_x^2, v_y^2, respectively, then $\alpha X + \beta Y$ is sub-Gaussian with variance factor $\alpha^2 v_x^2 + \beta^2 v_y^2 \forall \alpha, \beta \in \mathbb{R}$.*

Proof. Recall the equivalent definition of sub-Gaussianity that X, Y are sub-Gaussian iff for some $b, a > 0$ and all $\lambda > 0$

$$\begin{aligned} \mathbb{E}\exp\left(\lambda(X - \mathbb{E}X)\right) &\leq \exp(\lambda^2 v_x^2/2) \\ \mathbb{E}\exp\left(\lambda(Y - \mathbb{E}Y)\right) &\leq \exp(\lambda^2 v_y^2/2) \end{aligned}$$

Then

$$\begin{aligned} \mathbb{E}\exp\left(\lambda(\alpha X + \beta Y - \alpha\mathbb{E}X - \beta\mathbb{E}Y)\right) &\leq \sqrt{\mathbb{E}\exp\left(2\alpha\lambda(X - \mathbb{E}X)\right)}\sqrt{\mathbb{E}\exp\left(2\beta\lambda(Y - \mathbb{E}Y)\right)} \\ &\leq \sqrt{\exp(2\alpha^2\lambda^2 v_x^2)}\sqrt{\exp(2\beta^2\lambda^2 v_y^2)} \\ &= \exp((\alpha^2 v_x^2 + \beta^2 v_y^2)\lambda^2) \end{aligned}$$

□

The following Lemma gives the sub-Gaussianity and variance of the difference between the pseudo-reward and its expectation. We see that in the variance, all terms except those involving the inverse propensity weighted noise variance vanish as $f_{i,t}$ becomes a better estimate of $r_{i,t}$. Note that means and variances may be implicitly conditioned on the history.

Lemma 5. *If $\pi_{i,t}$ is correctly specified and $\tilde{\sigma}_{i,t}^2 \geq \frac{1}{c}$, the difference between the pseudo-reward and its expectation (taken wrt the action and noise) is mean zero sub-Gaussian with variance*

$$\begin{aligned} \textit{Var}\left(\tilde{R}_{i,t}^f(s,\bar{a})\right) = &\frac{(r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}))^2\pi_{i,t}(0|s) + \textit{Var}(\epsilon_{i,t})}{1 - \pi_{i,t}(0|s)} \\ &+ \frac{(r_{i,t}(s,0) - f_{i,t}(s,0))^2[1 - \pi_{i,t}(0|s)] + \textit{Var}(\epsilon_{i,t})}{\pi_{i,t}(0|s)} \\ &- 2(r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}))(r_{i,t}(s,0) - f_{i,t}(s,0)) \end{aligned}$$

Proof. We need to show that it is sub-Gaussian and upper bound its variance. We write the difference as

$$\begin{aligned} \tilde{R}_{i,t}^f(s,\bar{a}) - \mathbb{E}[\tilde{R}_{i,t}^f|s,\bar{a}] &= \tilde{R}_{i,t}^f(s,\bar{a}) - \Delta_{i,t}(s,\bar{a}) \\ &= \frac{R_{it} - f_{i,t}(s,A_{i,t})}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)} + \Delta_{i,t}^f(s,\bar{a}) - \Delta_{i,t}(s,\bar{a}) \\ &= \frac{r_{i,t}(s,A_{i,t}) - f_{i,t}(s,A_{i,t}) + \epsilon_{i,t}}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)} + \Delta_{i,t}^f(s,\bar{a}) - \Delta_{i,t}(s,\bar{a}) \end{aligned}$$

Note that $|r_{i,t}(s,A_{i,t})| \leq \max\left(|r_{i,t}(s,\bar{a})|, |r_{i,t}(s,0)|\right)$ and $|f_{i,t}(s,A_{i,t})| \leq \max\left(|f_{i,t}(s,\bar{a})|, |f_{i,t}(s,0)|\right)$. Thus since $\left|\frac{1}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}\right|$ is upper bounded by $c > 0$, we have that $\frac{r_{i,t}(s,A_{i,t}) - f_{i,t}(s,A_{i,t})}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}$ is bounded and thus (not necessarily mean zero) sub-Gaussian. Since $\epsilon_{i,t}$ is sub-Gaussian, its denominator is bounded, and the remaining terms are deterministic, the entire difference between the pseudo-reward and its mean is sub-Gaussian. Now

$$\begin{aligned}\text{Var}(\tilde{R}^f_{i,t}(s,\bar{a}) - \mathbb{E}[\tilde{R}^f_{i,t}|s,\bar{a}]) &= \text{Var}\left(\tilde{R}^f_{i,t}(s,\bar{a})\right) \\ &= \mathbb{E}\left[\tilde{R}^f_{i,t}(s,\bar{a})^2\right] - \Delta_{i,t}(s,\bar{a})^2 \end{aligned} \tag{14}$$

since $\mathbb{E}[\tilde{R}^f_{i,t}|s,\bar{a}]$ is not random. Now we expand the first term on the rhs.

$$\begin{aligned}\mathbb{E}\left[\tilde{R}^f_{i,t}(s,\bar{a})^2\right] &= \mathbb{E}\left[\left(\frac{r_{i,t}(s,A_{i,t}) - f_{i,t}(s,A_{i,t}) + \epsilon_{i,t}}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)} + \Delta^f_{i,t}(s,\bar{a})\right)^2\right] \\ &= \mathbb{E}\left[\left(\frac{r_{i,t}(s,A_{i,t}) - f_{i,t}(s,A_{i,t}) + \epsilon_{i,t}}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}\right)^2\right] \\ &\quad + 2\mathbb{E}\left[\frac{r_{i,t}(s,A_{i,t}) - f_{i,t}(s,A_{i,t}) + \epsilon_{i,t}}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}\right]\Delta^f_{i,t}(s,\bar{a}) + \Delta^f_{i,t}(s,\bar{a})^2 \\ &= \mathbb{E}\left[\left(\frac{r_{i,t}(s,A_{i,t}) - f_{i,t}(s,A_{i,t}) + \epsilon_{i,t}}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}\right)^2\right] \\ &\quad + 2(\Delta_{i,t}(s,\bar{a}) - \Delta^f_{i,t}(s,\bar{a}))\Delta^f_{i,t}(s,\bar{a}) + \Delta^f_{i,t}(s,\bar{a})^2 \end{aligned} \tag{15}$$

For the first term on the rhs of Eqn. 15,

$$\begin{aligned}&\mathbb{E}\left[\left(\frac{r_{i,t}(s,A_{i,t}) - f_{i,t}(s,A_{i,t}) + \epsilon_{i,t}}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}\right)^2\right] \\ &= \mathbb{E}\left[\left(\frac{r_{i,t}(s,A_{i,t}) - f_{i,t}(s,A_{i,t})}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}\right)^2\right] + \mathbb{E}\left[\left(\frac{\epsilon_{i,t}}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}\right)^2\right] \\ &= \frac{(r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}))^2 + \mathbb{E}[\epsilon^2_{i,t}]}{1-\pi_{i,t}(0|s)} + \frac{(r_{i,t}(s,0) - f_{i,t}(s,0))^2 + \mathbb{E}[\epsilon^2_{i,t}]}{\pi_{i,t}(0|s)}\end{aligned}$$

so that plugging this into Eqn. 15, we have

$$\begin{aligned}\mathbb{E}\left[\tilde{R}^f_{i,t}(s,\bar{a})^2\right] &= \frac{(r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}))^2 + \mathbb{E}[\epsilon^2_{i,t}]}{1-\pi_{i,t}(0|s)} + \frac{(r_{i,t}(s,0) - f_{i,t}(s,0))^2 + \mathbb{E}[\epsilon^2_{i,t}]}{\pi_{i,t}(0|s)} \\ &\quad + 2(\Delta_{i,t}(s,\bar{a}) - \Delta^f_{i,t}(s,\bar{a}))\Delta^f_{i,t}(s,\bar{a}) + \Delta^f_{i,t}(s,\bar{a})^2\end{aligned}$$

and plugging this into Eqn. 14 we obtain the variance.

$$\begin{aligned}\text{Var}\left(\tilde{R}^f_{i,t}(s,\bar{a})\right) &= \frac{(r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}))^2 + \text{Var}(\epsilon_{i,t})}{1-\pi_{i,t}(0|s)} + \frac{(r_{i,t}(s,0) - f_{i,t}(s,0))^2 + \text{Var}(\epsilon_{i,t})}{\pi_{i,t}(0|s)} \\ &\quad + 2(\Delta_{i,t}(s,\bar{a}) - \Delta^f_{i,t}(s,\bar{a}))\Delta^f_{i,t}(s,\bar{a}) + \Delta^f_{i,t}(s,\bar{a})^2 - \Delta_{i,t}(s,\bar{a})^2\end{aligned} \tag{16}$$

Note that

$$\begin{aligned}&2(\Delta_{i,t}(s,\bar{a}) - \Delta^f_{i,t}(s,\bar{a}))\Delta^f_{i,t}(s,\bar{a}) + \Delta^f_{i,t}(s,\bar{a})^2 - \Delta_{i,t}(s,\bar{a})^2 \\ &= 2(\Delta_{i,t}(s,\bar{a}) - \Delta^f_{i,t}(s,\bar{a}))\Delta^f_{i,t}(s,\bar{a}) - (\Delta_{i,t}(s,\bar{a}) - \Delta^f_{i,t}(s,\bar{a}))(\Delta_{i,t}(s,\bar{a}) + \Delta^f_{i,t}(s,\bar{a})) \\ &= (\Delta_{i,t}(s,\bar{a}) - \Delta^f_{i,t}(s,\bar{a}))(\Delta^f_{i,t}(s,\bar{a}) - \Delta_{i,t}(s,\bar{a})) \\ &= -(\Delta_{i,t}(s,\bar{a}) - \Delta^f_{i,t}(s,\bar{a}))^2\end{aligned}$$

and plugging this into Eqn. 16,

$$\begin{aligned}\text{Var}\left(\tilde{R}^f_{i,t}(s,\bar{a})\right) &= \frac{(r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}))^2 + \text{Var}(\epsilon_{i,t})}{1-\pi_{i,t}(0|s)} + \frac{(r_{i,t}(s,0) - f_{i,t}(s,0))^2 + \text{Var}(\epsilon_{i,t})}{\pi_{i,t}(0|s)} \\ &\quad - (\Delta_{i,t}(s,\bar{a}) - \Delta^f_{i,t}(s,\bar{a}))^2\end{aligned}$$

as desired. Now note that

$$\begin{aligned}\left(\Delta_{i,t}(s,\bar{a}) - \Delta^f_{i,t}(s,\bar{a})\right)^2 &= \left(r_{i,t}(s,\bar{a}) - r_{i,t}(s,0) - (f_{i,t}(s,\bar{a}) - f_{i,t}(s,0))\right)^2 \\ &= \left(r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}) - (r_{i,t}(s,0) - f_{i,t}(s,0))\right)^2 \\ &= (r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}))^2 + (r_{i,t}(s,0) - f_{i,t}(s,0))^2 \\ &\quad - 2(r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}))(r_{i,t}(s,0) - f_{i,t}(s,0))\end{aligned}$$

so that

$$\begin{aligned}\text{Var}\left(\tilde{R}^f_{i,t}(s,\bar{a})\right) = &\frac{(r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}))^2\pi_{i,t}(0|s) + \text{Var}(\epsilon_{i,t})}{1-\pi_{i,t}(0|s)} \\ &+ \frac{(r_{i,t}(s,0) - f_{i,t}(s,0))^2[1-\pi_{i,t}(0|s)] + \text{Var}(\epsilon_{i,t})}{\pi_{i,t}(0|s)} \\ &- 2(r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}))(r_{i,t}(s,0) - f_{i,t}(s,0))\end{aligned}$$

□

Corollary 1.

$$\begin{aligned}\mathbb{E}_{p(s,\bar{a})}\left[\textit{Var}\left(\tilde{R}^f_{i,t}(s,\bar{a})\right)\right] \leq &\frac{\mathbb{E}_{p(s,\bar{a})}\left[(r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}))^2\right]\pi_{i,t}(0|s) + \textit{Var}(\epsilon_{i,t})}{1-\pi_{i,t}(0|s)} \\ &+ \frac{\mathbb{E}_{p(s,\bar{a})}\left[(r_{i,t}(s,0) - f_{i,t}(s,0))^2\right][1-\pi_{i,t}(0|s)] + \textit{Var}(\epsilon_{i,t})}{\pi_{i,t}(0|s)} \\ &+ 2\sqrt{\mathbb{E}_{p(s,\bar{a})}\left[(r_{i,t}(s,\bar{a}) - f_{i,t}(s,\bar{a}))^2\right]}\sqrt{\mathbb{E}_{p(s)}\left[(r_{i,t}(s,0) - f_{i,t}(s,0))^2\right]}\end{aligned}$$

Proof. It follows immediately by taking expectations and applying Cauchy Schwartz. □

Corollary 2. $(r_{i,t}(\cdot,0) - f_{i,t}(\cdot,0))^2 = o_P(k^{-1/2})$ *and* $(r_{i,t}(\cdot,\bar{a}) - f_{i,t}(\cdot,\bar{a}))^2 = o_P(k^{-1/2})$

Proof. Note that

$$\begin{aligned}P\left((r_{i,t}(\cdot,0) - f_{i,t}(\cdot,0))^2 k^{1/2} > C\right) &\leq \frac{\mathbb{E}_{p(\mathcal{H}_{k-1})}\mathbb{E}_{p(s)}\left[(r_{i,t}(s,0) - f_{i,t}(s,0))^2\right]k^{1/2}}{C} \\ &= \frac{\mathbb{E}_{p(\mathcal{H}_{k-1})}o_P(k^{-1/2})k^{1/2}}{C} \text{ Assumption 3} \\ &= \frac{\mathbb{E}_{p(\mathcal{H}_{k-1})}o_P(1)}{C} \\ &\leq \delta \text{ for sufficiently large } k\end{aligned}$$

so that $(r_{i,t}(\cdot,0) - f_{i,t}(\cdot,0))^2 = o_P(k^{-1/2})$. In the second to last line we used that since $r_{i,t}, f_{i,t}$ are bounded, $\mathbb{E}_{p(s)}\left[(r_{i,t}(s,0) - f_{i,t}(s,0))^2\right]$ is bounded. For bounded random variables, convergence in probability implies convergence in mean. A similar result holds when using $\bar{a}$ instead of 0. □

Corollary 3.

$$\begin{aligned}\textit{Var}\left(\tilde{R}^f_{i,t}(s,\bar{a})\right) \leq &\frac{o_P(k^{-1/2})\pi_{i,t}(0|s) + \textit{Var}(\epsilon_{i,t})}{1-\pi_{i,t}(0|s)} + \frac{o_P(k^{-1/2})[1-\pi_{i,t}(0|s)] + \textit{Var}(\epsilon_{i,t}) + \textit{Var}(\epsilon_{i,t})}{\pi_{i,t}(0|s)} \\ &+ 2o_P(k^{-1/2})\end{aligned}$$

Proof. This follows immediately from the previous two Corollaries. □

Corollary 4. *For all $\delta > 0$, there exists $C > 0$ s.t. w.p. at least $1-\delta$,*

$$\begin{aligned}\textit{Var}\left(\tilde{R}^f_{i,t}(s,\bar{a})\right) &\leq \frac{Ck^{-1/2}\pi_{i,t}(0|s) + \textit{Var}(\epsilon_{i,t})}{1-\pi_{i,t}(0|s)} + \frac{Ck^{-1/2}[1-\pi_{i,t}(0|s)] + \textit{Var}(\epsilon_{i,t})}{\pi_{i,t}(0|s)} + 2Ck^{-1/2} \\ &\equiv v_k^2\end{aligned}$$

Proof. By the definition of o_P, if $(f_{i,t} - r_{i,t})^2 = o_P(k^{-1/2})$ for any $A > 0, \delta > 0$, there exists K s.t. if $K > k$, then w.p. at least $1 - \delta$,

$$(f_{i,t} - r_{i,t})^2 \leq Ak^{-1/2}.$$

Then for all $k \in \mathbb{N}$, since f, r are bounded

$$(f_{i,t} - r_{i,t})^2 \leq 16B^2 I(k \leq K) + I(k > K)Ak^{-1/2}.$$

Now choose

$$C \equiv \max\left(A, 16B^2 K^{1/2}\right).$$

□

Remark 1. *In the limit as $k \to \infty$*

$$\begin{aligned}\tilde{\sigma}_{i,t}^2 \mathit{Var}\left(\tilde{R}_{i,t}^f(s,\bar{a})\right) &\leq \pi_{i,t}(0|s)\mathit{Var}(\epsilon_{i,t}) + (1 - \pi_{i,t}(0|s))\mathit{Var}(\epsilon_{i,t}) \\ &= \mathit{Var}(\epsilon_{i,t})\end{aligned}$$

In the next remark, we show what the variance would be if we did *not* use DML and estimate $f_{i,t} \approx r_{i,t}$, but used only the inverse propensity weighted observed reward as the pseudo-reward. This was done in Greenewald et al. (2017). In this case, there are terms dependent on the mean reward that do *not* vanish as the number of stages goes to infinity.

Remark 2. *If we instead used as our pseudo-reward the inverse propensity weighted observed reward*

$$\tilde{R}_{i,t}(s,\bar{a}) = \frac{R_{it}}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}$$

this would be unbiased with variance

$$v_k^2 \equiv \frac{r_{i,t}(s,\bar{a})^2 + \mathit{Var}(\epsilon_{i,t})}{1 - \pi_{i,t}(0|s)} + \frac{r_{i,t}(s,0)^2 + \mathit{Var}(\epsilon_{i,t})}{\pi_{i,t}(0|s)} - \Delta_{i,t}(s,\bar{a})^2$$

Proof. The unbiasedness is clear from our proof of Lemma 2. For the variance,

$$\begin{aligned}\text{Var}\left(\frac{R_{it}}{\delta_{A_{i,t}=\bar{a}} - \pi_{i,t}(0|s)}\right) &= \mathbb{E}\left[\frac{(r_{i,t}(s,\bar{a}) + \epsilon_{i,t})^2}{1 - \pi_{i,t}(0|s)} + \frac{(r_{i,t}(s,0) + \epsilon_{i,t})^2}{\pi_{i,t}(0|s)}\right] - \Delta_{i,t}(s,\bar{a})^2 \\ &= \frac{r_{i,t}(s,\bar{a})^2 + \text{Var}(\epsilon_{i,t})}{1 - \pi_{i,t}(0|s)} + \frac{r_{i,t}(s,0)^2 + \text{Var}(\epsilon_{i,t})}{\pi_{i,t}(0|s)} - \Delta_{i,t}(s,\bar{a})^2\end{aligned}$$

□

Here we collect some important results. We first adapt an important concentration inequality for regularized least-squares estimates. Our proof follows the same basic strategy as Abbasi-Yadkori et al. (2011), but with modifications due to (a) the use of weighted least squares (b) the use of pseudo-rewards to estimate differential rewards (c) replacing the scaled diagonal regularization with Laplacian regularization.

Lemma 6 (Adapted from Theorem 2 in Abbasi-Yadkori et al. (2011))**.** *Let $\hat{\theta}_k$ be the stage k regularized least squares (RLS) estimate from Algorithm 1 and θ^* the ground truth where we assume $\|\theta^*\| \leq S$. For any $\delta > 0$, w.p. at least $1 - \delta$ the estimates $\{\theta_k\}_{k=0}^{\infty}$ satisfies for any $\{x_k\}_{k=0}^{\infty}$,*

$$|x_k^\top(\hat{\theta}_k - \theta^*)| \leq \|x_k\|_{V_k^{-1}}\left(v_k\sqrt{2\log\left(\frac{\det(V_k)^{1/2}\det(V_0)^{-1/2}}{\delta}\right)} + \|\theta^*\|_{V_0}\right), \tag{17}$$

where v_k^2 is the variance factor for the difference between the pseudo-reward and its mean at stage k. In particular, setting $x_k = V_k(\hat{\theta}_k - \theta^*)$ implies

$$\|\hat{\theta}_k - \theta^*\|_{V_k} \leq v_k\sqrt{2\log\left(\frac{\det(V_k)^{1/2}\det(V_0)^{-1/2}}{\delta}\right)} + \|\theta^*\|_{V_0}$$

holds w.p. at least $1 - \delta$ for all $k \geq 1$.

Proof. Let $m_{i,t} = \tilde{\sigma}_{i,t}\phi(\mathbf{x}_{i,t})$ and $\rho_{i,t} = \tilde{\sigma}_{i,t}[\tilde{R}^f_{i,t}(s,\bar{a}) - \mathbb{E}[\tilde{R}^f_{i,t}|s,\bar{a}]]$. Further let

$$\begin{aligned}
\xi_k &\equiv \sum_{(i,t)\in\mathcal{O}_{k-1}} \tilde{\sigma}^2_{i,t}[\tilde{R}^f_{i,t}(s,\bar{a}) - \mathbb{E}[\tilde{R}^f_{i,t}|s,\bar{a}]]\phi(\mathbf{x}_{i,t}) \\
&= \sum_{(i,t)\in\mathcal{O}_{k-1}} \tilde{\sigma}_{i,t}[\tilde{R}^f_{i,t}(s,\bar{a}) - \mathbb{E}[\tilde{R}^f_{i,t}|s,\bar{a}]]m_{i,t} \\
&= \sum_{(i,t)\in\mathcal{O}_{k-1}} m_{i,t}\rho_{i,t}
\end{aligned}$$

Then noting that $V_k = \sum_{(i,t)\in\mathcal{O}_{k-1}} m_{i,t}m_{i,t}^\top + V_0$ and $b_k = \sum_{(i,t)\in\mathcal{O}_{k-1}} \sigma^2_{i,t}\tilde{R}^f_{i,t}\phi_{i,t}$and letting W_k be the diagonal matrix of weights $\tilde{\sigma}^2_{i,t}$, we have

$$\begin{aligned}
\hat{\theta}_k &= V_k^{-1}b_k \\
&= V_k^{-1}(\xi_k + \Phi_k^\top W_k \mathbb{E}[R_k^{f_k}|\bar{a},s]) \\
&= V_k^{-1}\xi_k + V_k^{-1}\Phi_k^\top W_k\Delta_k \text{ by Lemma 2} \\
&= V_k^{-1}\xi_k + V_k^{-1}\Phi_k^\top W_k\Phi_k\theta^* \\
&= V_k^{-1}\xi_k + V_k^{-1}(\Phi_k^\top W_k\Phi_k + V_0)\theta^* - V_k^{-1}V_0\theta^* \\
&= V_k^{-1}\xi_k + \theta^* - V_k^{-1}V_0\theta_k^*
\end{aligned}$$

and thus

$$\hat{\theta}_k - \theta^* = V_k^{-1}(\xi_k - V_0\theta^*)$$

which gives

$$|x_k^\top\hat{\theta}_k - x_k^\top\theta^*| \le \|x_k\|_{V_k^{-1}}(\|\xi_k\|_{V_k^{-1}} + \|V_0\theta^*\|_{V_k^{-1}})$$

Now since ξ_k is sub-Gaussian with variance factor v_k^2, by Theorem 1 in Abbasi-Yadkori et al. (2011), w.p. $1-\delta$,

$$\|\xi_k\|^2_{V_k^{-1}} \le 2v_k^2\log\left(\frac{\det(V_k)^{1/2}\det(V_0)^{-1/2}}{\delta}\right)$$

Further note that since $V_0 \preceq V_k$, then $V_k^{-1} \preceq V_0^{-1}$. Thus

$$\begin{aligned}
\|V_0\theta^*\|^2_{V_k^{-1}} &= \theta^{*\top}V_0^\top V_k^{-1}V_0\theta^* \\
&\le \theta^{*\top}V_0^\top V_0^{-1}V_0\theta^* \\
&= \theta^{*\top}V_0\theta^*.
\end{aligned}$$

Finally, setting $x_k = V_k(\hat{\theta}_k - \theta^*)$ implies

$$\begin{aligned}
\|\hat{\theta}_k - \theta^*\|^2_{V_k} &= (\hat{\theta}_k - \theta^*)^\top V_k(\hat{\theta}_k - \theta^*) \\
&= |x_k^\top\hat{\theta}_k - x_k^\top\theta^*| \\
&\le \|x_k\|_{V_k^{-1}}\left(v_k\sqrt{2\log\left(\frac{\det(V_k^{-1})^{1/2}\det(V_0)^{-1/2}}{\delta}\right)} + \sqrt{\theta^{*\top}V_0\theta^*}\right) \\
&= \|V_k(\hat{\theta}_k - \theta^*)\|_{V_k^{-1}}\left(v_k\sqrt{2\log\left(\frac{\det(V_k^{-1})^{1/2}\det(V_0)^{-1/2}}{\delta}\right)} + \sqrt{\theta^{*\top}V_0\theta^*}\right) \\
&= \|\hat{\theta}_k - \theta^*\|_{V_k}\left(v_k\sqrt{2\log\left(\frac{\det(V_k^{-1})^{1/2}\det(V_0)^{-1/2}}{\delta}\right)} + \sqrt{\theta^{*\top}V_0\theta^*}\right)
\end{aligned}$$

and dividing both sides by $\|\hat{\theta}_k - \theta^*\|_{V_k}$ implies

$$\|\hat{\theta}_k - \theta^*\|_{V_k} \le v_k\sqrt{2\log\left(\frac{\det(V_k)^{1/2}\det(V_0)^{-1/2}}{\delta}\right)} + \sqrt{\theta^{*\top}V_0\theta^*}$$

□

In the above inequality, we will need to bound $\|\theta^*\|_{V_0}$ in order to obtain useful confidence sets.

Lemma 7.

$$\|\theta^*\|_{V_0} \leq \sqrt{\gamma} B_{shared} + \sqrt{\lambda W K}(\sqrt{D_{user}} + \sqrt{D_{time}}) + \sqrt{\gamma W K}(B_{user} + B_{time}).$$

Proof. Recall that

$$V_0 = \text{diag}(\gamma I_p, \lambda L_{\otimes}^{\text{user}} + \gamma I_{Kp}, \lambda L_{\otimes}^{\text{time}} + \gamma I_{Kp}), \tag{18}$$

Then

$$\begin{aligned}
\theta^{*\top} V_0 \theta^* &= \gamma \|\theta_{\text{shared}}\|^2 + \lambda \text{tr}\left(\Theta_{\text{user}}^\top L^{\text{user}} \Theta_{\text{user}}\right) + \gamma \text{tr}\left(\Theta_{\text{user}}^\top I_{K_p} \Theta_{\text{user}}\right) \\
&\quad + \lambda \text{tr}\left(\Theta_{\text{time}}^\top L^{\text{time}} \Theta_{\text{time}}\right) + \gamma \text{tr}\left(\Theta_{\text{time}}^\top I_{K_p} \Theta_{\text{time}}\right) \\
&= \gamma \|\theta_{\text{shared}}\|^2 + \lambda \sum_{(i,j) \in E_{\text{user}}} \|\theta_i^{\text{user}} - \theta_j^{\text{user}}\|_2^2 + \lambda \sum_{(i,j) \in E_{\text{time}}} \|\theta_i^{\text{time}} - \theta_j^{\text{time}}\|_2^2 \\
&\quad + \gamma \|\Theta_{\text{user}}\|_F^2 + \gamma \|\Theta_{\text{time}}\|_F^2 \\
&\leq \gamma B_{\text{shared}}^2 + \lambda W K (D_{\text{user}} + D_{\text{time}}) + \gamma W K (B_{\text{user}}^2 + B_{\text{time}}^2) \text{ by Assumption 5}
\end{aligned}$$

where we assume that each user and time have at most m neighbors. Now using $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ for $a, b > 0$ we have

$$\sqrt{\theta^{*\top} V_0 \theta^*} \leq \sqrt{\gamma} B_{\text{shared}} + \sqrt{\lambda W K}(\sqrt{D_{\text{user}}} + \sqrt{D_{\text{time}}}) + \sqrt{\gamma W K}(B_{\text{user}} + B_{\text{time}}).$$

□

We will also want to bound the log determinant terms. The next two Lemmas show us how to do so.

Lemma 8.

$$det(V_{K+1}) \leq \left(\frac{\frac{3K(K+1)}{8} + \gamma(2K+1)p + 2\lambda MKp}{(2K+1)p} \right)^{(2K+1)p}$$

Proof. By AM-GM inequality,

$$\begin{aligned}
\det(V_{K+1}) &= \det\left(\sum_{(i,t) \in \mathcal{O}_K} \tilde{\sigma}_{i,t}^2 \phi(x_{i,t}) \phi(x_{i,t})^\top + V_0 \right) \\
&\leq \left(\frac{\text{tr}\left(\sum_{(i,t) \in \mathcal{O}_K} \tilde{\sigma}_{i,t}^2 \phi(x_{i,t}) \phi(x_{i,t})^\top + V_0 \right)}{(2K+1)p} \right)^{(2K+1)p}
\end{aligned}$$

Now

$$\begin{aligned}
\text{tr}\left(\sum_{(i,t) \in \mathcal{O}_K} \tilde{\sigma}_{i,t}^2 \phi(x_{i,t}) \phi(x_{i,t})^\top \right) &\leq \sum_{(i,t) \in \mathcal{O}_K} \tilde{\sigma}_{i,t}^2 \|\phi(x_{i,t})\|_2^2 \\
&\leq \frac{3}{4} \frac{K(K+1)}{2}.
\end{aligned}$$

Recall that $V_0 = \text{diag}(\gamma I_p, \lambda L_{\otimes}^{\text{user}} + \gamma I_{Kp}, \lambda L_{\otimes}^{\text{time}} + \gamma I_{Kp})$, so that

$$\begin{aligned}
\text{tr}(V_0) &= \gamma \left(\text{tr}(I_p) + \text{tr}(I_{K_p}) + \text{tr}(I_{K_p}) \right) + \lambda \left(\text{tr}(L_{\otimes}^{\text{user}}) + \text{tr}(L_{\otimes}^{\text{time}}) \right) \\
&= \gamma(2K+1)p + \lambda \left(\text{tr}(L^{\text{user}}) \text{tr}(I_p) + \text{tr}(L^{\text{time}}) \text{tr}(I_p) \right) \\
&= \gamma(2K+1)p + \lambda p (\text{tr}(L^{\text{user}}) + \text{tr}(L^{\text{time}})) \\
&\leq \gamma(2K+1)p + 2\lambda W K p
\end{aligned}$$

and thus

$$\det(V_{K+1}) \leq \left(\frac{\frac{3K(K+1)}{8} + \gamma(2K+1)p + 2\lambda MKp}{(2K+1)p} \right)^{(2K+1)p}$$

□

Lemma 9.

$$\log \frac{det(V_K)}{det(V_0)} \leq (2K+1)p \log\left(\frac{3K(K+1)}{\gamma 8(2K+1)p} + 1 + \frac{2\lambda MK}{\gamma(2K+1)}\right)$$

Proof. Note that

$$\det(V_0) \geq \gamma^{(2K+1)p},$$

so that

$$\frac{\det(V_K)}{\det(V_0)} \leq \left(\frac{\frac{3K(K+1)}{8} + \gamma(2K+1)p + 2\lambda MKp}{\gamma(2K+1)p}\right)^{(2K+1)p}$$

Taking the $\log$, we have

$$\begin{aligned}\log \frac{\det(V_K)}{\det(V_0)} &\leq \log\left(\frac{\frac{3K(K+1)}{8} + \gamma(2K+1)p + 2\lambda MKp}{\gamma(2K+1)p}\right)^{(2K+1)p} \\ &\leq (2K+1)p \log\left(\frac{3K(K+1)}{\gamma 8(2K+1)p} + 1 + \frac{2\lambda MK}{\gamma(2K+1)}\right)\end{aligned}$$

□

Corollary 5. *If Assumption 3, then for any $\delta > 0$, there exists $C > 0$ s.t. w.p. at least $1-\delta$ the estimates $\{\hat{\theta}_k\}_{k=0}^{\infty}$ in Algorithm 2 satisfies for any $\{x_k\}_{k=0}^{\infty}$,*

$$|x_k^\top(\hat{\theta}_k - \theta^*)| \leq \|x_k\|_{V_k^{-1}} \left(\left(\frac{C}{k^{1/2}} + \sigma^2 c^2\right)\sqrt{2\log\left(\frac{\det(V_k)^{1/2}\det(V_0)^{-1/2}}{\delta/2}\right)} + \|\theta^*\|_{V_0}\right), \tag{19}$$

In particular, setting $x_k = V_{k-1}(\hat{\theta}_{k-1} - \theta^*)$ implies

$$\|\hat{\theta}_k - \theta^*\|_{V_k} \leq \left(\frac{C}{k^{1/2}} + \sigma^2 c^2\right)\sqrt{2\log\left(\frac{\det(V_k)^{1/2}\det(V_0)^{-1/2}}{\delta/2}\right)} + \|\theta^*\|_{V_0}$$

holds w.p. at least $1-\delta$ for all $k \geq 1$.

Proof. Use Corollary 3 and Lemma 6, each with $\delta/2$. Then w.p. at least $1-\delta$ the result holds. □

We next state a slightly modified form of a standard result of RLS (Lemma 11 in Abbasi-Yadkori et al. (2011)) that helps to guarantee that the prediction error is cumulatively small. This bounds the sum of quadratic forms where the matrix is the inverse Gram matrix and the arguments are the feature vectors. We use such terms to construct a martingale in the regret bound so that we can bound such terms and the martingale.

Proposition 1. *Let $\lambda \geq 1$ and $\gamma \geq 1$. For any arbitrary sequence $(x_{i,t})_{(i,t)\in\mathcal{O}_k}$, let*

$$V_{k+1} \equiv \sum_{(i,t)\in\mathcal{O}_k} \tilde{\sigma}_{i,t}^2 \phi(x_{i,t})\phi(x_{i,t})^\top + V_0,$$

be the regularized Gram matrix. Then

$$\sum_{k=1}^{K} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \|\phi(x_{i,t})\|_{V_k^{-1}}^2 \leq 2c \log\left(\frac{\det(V_{K+1})}{\det(V_0)}\right).$$

where c is a constant such that $0 < \frac{1}{c} < \tilde{\sigma}_{i,t}^2 \forall i, t \in \mathbb{N}$.

Proof. By Lemma 11 in Abbasi-Yadkori et al. (2011), we have

$$\sum_{k=1}^{K} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \tilde{\sigma}_{i,t}^2 \|\phi(x_{i,t})\|^2_{V_k^{-1}} \leq 2\log\left(\frac{\det(V_{K+1})}{\det(V_0)}\right).$$

The lower bound on the weights implies

$$\sum_{k=1}^{K} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \|\phi(x_{i,t})\|^2_{V_k^{-1}} \leq 2c\log\left(\frac{\det(V_{K+1})}{\det(V_0)}\right)$$

as desired. □

Finally, we state Azuma's concentration inequality which describes concentration of super-martingales with bounded differences and is useful in controlling the regret due to the randomization of Thompson sampling.

Proposition 2 (Azuma's concentration inequality)**.** *If a super-martingale $(Y_t)_{t\geq 0}$ corresponding to a filtration $\mathcal{F}_t$ satisfies $|Y_t - Y_{t-1}| < c_t$ some constant c_t for all $t = 1, \ldots, T$ then for any $\alpha > 0$:*

$$P\left(Y_T - Y_0 \geq \alpha\right) \leq \exp\left(-\frac{\alpha^2}{2\sum_{t=1}^{T} c_t^2}\right).$$

E.3 Proof of Theorem 1

We first decompose the regret bound

$$\begin{aligned}
&\sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \left[\pi^\star_{i,t} x(S_{i,t}, a^*_{i,t})^\top \theta^*_{i,t} - \pi_{i,t} x(S_{i,t}, A_{i,t})^\top \theta^*_{i,t}\right] \\
&= \sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \Big[(\pi^\star_{i,t} - \pi_{i,t}) x(S_{i,t}, A_{i,t})^\top \theta^*_{i,t} \\
&\qquad\qquad + \pi^\star_{i,t} \left\{x(S_{i,t}, a^*_{i,t})^\top \theta^*_{i,t} - x(S_{i,t}, A_{i,t})^\top \theta^*_{i,t}\right\}\Big] \\
&\leq \sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \left[(\pi^\star_{i,t} - \pi_{i,t}) x(S_{i,t}, A_{i,t})^\top \theta^*_{i,t}\right] \\
&\qquad\qquad + \sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \left[x(S_{i,t}, a^*_{i,t})^\top \theta^*_{i,t} - x(S_{i,t}, A_{i,t})^\top \theta^*_{i,t}\right]
\end{aligned}$$

For the first term, $\sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \left[(\pi^\star_{i,t} - \pi_{i,t}) x(S_{i,t}, A_{i,t})^\top \theta^*_{i,t}\right]$, one can apply a nearly identical proof to that of Greenewald et al. (2017). For the second term we need a novel strategy, and thus focus on that term. The proof follows closely from Abeille & Lazaric (2017) with several adjustments. Assumption 6 implies that we only need to consider the unit ball $\mathcal{X} = \{\|x\| \leq 1\}$. Then the second term of the regret can be decomposed into

$$\underbrace{\sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \left((\phi(x^\star_{i,t}))^\top \theta^\star - \phi(x_{i,t})^\top \tilde{\theta}_k\right)}_{R^{TS}(K)} + \underbrace{\sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \left(\phi(x_{i,t})^\top \tilde{\theta}_k - \phi(x_{i,t})^\top \theta^\star\right)}_{R^{RLS}(K)}$$

where $\phi(x^\star_{i,t})$ is the context vector under the optimal action and $\theta^\star$ is the true parameter value. The first term is the regret due to the random deviations caused by sampling $\tilde{\theta}_k$ and whether it provides sufficient useful information about the true parameter $\theta^\star_k$. The second term is the concentration of the sampled term around the true linear model for the advantage function.

Definition 2. *We define the filtration $\mathcal{F}_k$ as the information accumulated up to stage k before the sampling procedure, that is, $\mathcal{F}_k = (\mathcal{F}_1, \sigma(x_1, r_2, x_2, \ldots, x_{k-1}, r_{k-1}))$, and filtration $\mathcal{F}^x_k$ as the information accumulated up to stage k and including the sampled context, that is, $\mathcal{F}_t = (\mathcal{F}_1, \sigma(x_1, r_2, x_2, \ldots, x_{k-1}, r_{k-1}, x_k))$.*

Bounding $R^{RLS}(T)$. We decompose the second term into the variation of the point estimate and the variation of the random sample around the point estimate:

$$\sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \left(\phi(x_{i,t})^\top \tilde{\theta}_k - \phi(x_{i,t})^\top \hat{\theta}_k\right) + \sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \left(\phi(x_{i,t})^\top \hat{\theta}_k - \phi(x_{i,t})^\top \theta_k^\star\right)$$

The first term describes the deviation of the TS linear predictor from the RLS one, while the second term describes the deviation of the RLS linear predictor from the true linear predictor. The first term is controlled by the construction of the sampling distribution D^{TS}, while the second term is controlled by the RLS estimate being a minimizer of the regularized cumulative squared error in (8). In particular, the first term will be small when the TS estimate concentrates around the RLS one, while the second will be small when the RLS estimate concentrates around the true parameter vector. The next proposition gives a lower bound on the probability that, for all stages, both the RLS parameter vector concentrates around the true parameter vector and the TS parameter vector concentrates around the RLS one.

Recall that

$$\beta_k(\delta) = v_k \left[2\log\left(\frac{\det(V_k)^{1/2}}{\det(V_0)^{1/2}\delta/2}\right)\right]^{1/2} + B$$

where $B = \sqrt{\gamma}B_{\text{shared}} + \sqrt{\lambda MK}(\sqrt{D_{\text{user}}} + \sqrt{D_{\text{time}}}) + \sqrt{\gamma MK}(B_{\text{user}} + B_{\text{time}})$. This is, from Lemma 6 and Corollary 7, an upper bound on a $1-\delta$ confidence set on the RLS estimator.

Proposition 3. *Let $\hat{E}_k$ denote the event that $\hat{\theta}_k$ concentrates around the true parameter for all $l \le k$, i.e., $\hat{E}_k = \{\forall l \le k, \|\hat{\theta}_l - \theta_l^\star\|_{V_l} \le \beta_l(\delta')\}$. Let $\gamma_k(\delta) \equiv \beta_k(\delta')\sqrt{cd\log\frac{c'd}{\delta}}$ Let $\tilde{E}_k$ denote the event that $\tilde{\theta}_l$ concentrates around the estimated parameter for all $l \le k$, i.e., $\tilde{E}_k = \{\forall l \le k, \|\tilde{\theta}_l - \hat{\theta}_l\|_{V_l} \le \gamma_l(\delta')\}$. Let $E_k = \hat{E}_k \cap \tilde{E}_k$. Then $P(E_k) \ge 1 - \delta/2$.*

Proof. Let $\delta' = \delta/4K$, then Lemma 6 and a union bound give us

$$\begin{aligned} P(\hat{E}_K) &= P(\cap_{k=1}^K \{\|\hat{\theta}_k - \theta^\star\|_{V_k} \le \beta_k(\delta')\}) \\ &= 1 - \sum_{k=1}^K P(\|\hat{\theta}_k - \theta^\star\|_{V_k} > \beta_k(\delta')) \\ &= 1 - \sum_{k=1}^K \delta' = 1 - \delta' K = 1 - \delta/4. \end{aligned}$$

Applying the TS sampling distribution and $\tilde{\theta}_k = \hat{\theta}_k + \beta_k(\delta')V_k^{-1/2}\eta_k$ where η_t is drawn i.i.d. from D^{TS} we have

$$P\left(\|\tilde{\theta}_k - \hat{\theta}_k\|_{V_k} \le \beta_k(\delta')\sqrt{cd\log\left(\frac{c'd}{\delta'}\right)}\right) = P\left(\|\eta_k\| \le \sqrt{cd\log\left(\frac{c'd}{\delta'}\right)}\right) \ge 1 - \delta'.$$

by Definition 1. A union-bound argument yields the conclusion. □

We can then bound $R^{RLS}(K)$ by leveraging Lemma 6 and decomposing the error via

$$\begin{aligned} R^{RLS}(K) \le &\sum_{k=1}^K \frac{1[E_K]}{k} \left[\sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} |\phi(x_{i,t})^\top(\tilde{\theta}_k - \hat{\theta}_k)|\right] \\ &+ \sum_{k=1}^K \frac{1[E_K]}{k} \left[\sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} |\phi(x_{i,t})^\top(\hat{\theta}_k - \theta^*)|\right] \end{aligned}$$

By definition of the event E_K, we have

$$|\phi(x_{i,t})^\top(\tilde{\theta}_k - \hat{\theta}_k)|1[E_k] \le \|\phi(x_{i,t})\|_{V_k^{-1}}\gamma_k(\delta'), \quad |\phi(x_{i,t})^\top(\hat{\theta}_k - \theta^\star)|1[E_k] \le \|\phi(x_{i,t})\|_{V_k^{-1}}\beta_k(\delta')$$

so from Proposition 1, we have

$$\begin{aligned}
&\sum_{k=1}^{K} \frac{1[E_K]}{k} \left[\sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} |\phi(x_{i,t})^\top(\tilde{\theta}_k - \hat{\theta}_k)| \right] \\
&\leq \gamma_K(\delta') \sum_{k=1}^{K} \left[\sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \frac{1}{k} \|\phi(x_{i,t})\|_{V_k^{-1}} \right] \\
&\leq \gamma_K(\delta') \sqrt{\sum_{k=1}^{K} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \frac{1}{k^2}} \sqrt{\sum_{k=1}^{K} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \|\phi(x_{i,t})\|^2_{V_k^{-1}}} \\
&\leq \gamma_K(\delta') \sqrt{\sum_{k=1}^{K} \frac{1}{k}} \sqrt{\sum_{k=1}^{K} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \|\phi(x_{i,t})\|^2_{V_k^{-1}}} \\
&\leq \gamma_K(\delta') \sqrt{H_K} \sqrt{\sum_{(i,t)\in\mathcal{O}_k} \|\phi(x_{i,t})\|^2_{V_k^{-1}}} \\
&\leq \gamma_K(\delta') \sqrt{H_K} \sqrt{2c \log\left(\frac{\det(V_{K+1})}{\det(V_0)}\right)}.
\end{aligned}$$

Using a similar derivation for the $\beta_k(\delta')$ case, we obtain

$$\begin{aligned}
R^{RLS}(K) &\leq (\beta_K(\delta') + \gamma_K(\delta')) \sqrt{\sum_{k=1}^{K} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \frac{1}{k^2}} \sqrt{2c \log\left(\frac{\det(V_{K+1})}{\det(V_0)}\right)} \\
&\leq (\beta_K(\delta') + \gamma_K(\delta')) \sqrt{\sum_{k=1}^{K} \frac{1}{k}} \sqrt{2c \log\left(\frac{\det(V_{K+1})}{\det(V_0)}\right)} \\
&\leq (\beta_K(\delta') + \gamma_K(\delta')) \sqrt{H_K} \sqrt{2c \left[(2K+1)p \log\left(\frac{3K(K+1)}{\gamma 8(2K+1)p} + 1 + \frac{2\lambda MK}{\gamma(2K+1)}\right)\right]}
\end{aligned}$$

with probability at least $1 - \delta/2$ by Proposition 3, where H_K is the harmonic number. Note that $H_K \sim \log(K)$ for large K.

Bounding $R^{TS}(T)$. Leveraging Abeille & Lazaric (2017), Definition 1 lets us bound $R^{TS}(K)$ under the event E_k. Let $\phi(x^\star_{i,t})(\theta) = \arg\max_{x_{i,t}\in\mathcal{X}} \theta^\top \phi(x_{i,t})$. Then

$$R^{TS}(K) \leq \sum_{k=1}^{K} \frac{1}{k} R_k^{TS} 1[E_k] \leq \frac{4\gamma_K(\delta')}{d} \sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \mathbb{E}\left[\|\phi(x^\star_{i,t})(\tilde{\theta})\|_{V_k^{-1}} | \mathcal{F}_k\right] \tag{20}$$

We re-write the sum in (20) as:

$$\sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \|\phi(x_{i,t})\|_{V_k^{-1}} + \underbrace{\sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \left(\mathbb{E}\left[\|\phi(x^\star_{i,t})(\tilde{\theta})\|_{V_k^{-1}} | \mathcal{F}_k\right] - \|\phi(x_{i,t})\|_{V_k^{-1}}\right)}_{R_2^{TS}}$$

The first term is bounded by Proposition 1:

$$\sum_{k=1}^{K} \frac{1}{k} \sum_{(i,t)\in\mathcal{O}_k\setminus\mathcal{O}_{k-1}} \|\phi(x_{i,t})\|_{V_k^{-1}} \leq \sqrt{2cH_K \log\left(\frac{\det(V_{K+1})}{\det(V_0)}\right)}$$

The second term is a martingale by construction and so we can apply Azuma's inequality. Under Assumption 6, so since $V_k \leq \frac{1}{\lambda} I$ we have

$$\mathbb{E}\left[\|\phi(x_{i,t})^\star(\tilde{\theta})\|_{V_k^{-1}} | \mathcal{F}_t\right] - \|\phi(x_{i,t})\|_{V_k^{-1}} \leq \frac{2}{\sqrt{\lambda}}, \quad a.s.$$

This provides the upper-bound

$$R^{TS}(K) \leq \frac{4\gamma_K(\delta')}{d}\left(\sqrt{\frac{8K}{\lambda}\log\left(\frac{4}{\delta}\right)} + \sqrt{2cH_K(2K+1)p\log\left(\frac{3K(K+1)}{\gamma 8(2K+1)p} + 1 + \frac{2\lambda MK}{\gamma(2K+1)}\right)}\right).$$

Overall bound. Putting together the two bounds under a union bound argument yields the upper bound in Theorem 1; specifically, we have

$$\left(\beta_K(\delta') + \gamma_K(\delta')\left[1+\frac{4}{d}\right]\right)\sqrt{2cH_K(2K+1)p\log\left(\frac{3K(K+1)}{\gamma 8(2K+1)p} + 1 + \frac{2\lambda MK}{\gamma(2K+1)}\right)}$$
$$+\frac{4\gamma_K(\delta')}{p}\sqrt{\frac{8K}{\lambda}\log\left(\frac{4}{\delta}\right)}$$

F Notation Guide

For convenience, we summarize below some key notation used in the main paper.

- $a = 0$: control action
- q: number of non-baseline treatment arms
- $i = 1, 2, \ldots$: index for individuals (later in the paper, we consider N individuals)
- $t = 1, 2, \ldots$: index for decision points (later in the paper, we consider T decision points)
- $S_{i,t} \in \mathcal{S}$: context vector observed for individual i at decision point t
- $A_{i,t} \in \{0, \ldots, q\}$: action chosen for individual i at decision point t
- $R_{i,t} \in \mathbb{R}$: reward observed for individual i at decision point t
- $r_{i,t}(s,a) := \mathbb{E}\left[R_{i,t}|S_{i,t} = s, A_{i,t} = a\right]$: conditional model for the observed reward given the state and context
- $x(s,a) \in \mathbb{R}^{p\times 1}$: feature vector containing the state and action
- $\theta_{i,t} \in \mathbb{R}^p$: vector of parameters that may depend on the individual i and decision point t. Later $\theta_{i,t}$ is written as $\theta_{i,t} = \theta + \theta_i^{user} + \theta_t^{time}$, where θ_i^{user} is the individual-specific but time-invariant term and θ_t^{time} is a shared time-specific term
- $\delta_{a>0}$: indicator function that takes the value 1 if $a > 0$ and 0 otherwise
- $g_t(s)$: baseline reward function that is observed when individuals are randomized to receive no treatment
- $\Delta_{i,t}(s,a) := r_{i,t}(s,a) - r_{i,t}(s,0)$: linear differential reward for any action $a > 0$ and state s
- $\mathcal{H}_{i,t}$: history up to decision point t for individual i
- $\pi_{i,t}(a|s)$: probability of action $a \in [K]$ given current context $s \in \mathcal{S}$ for a fixed (implicit) history
- $\bar{a}_{i,t} \in [K]$: potential non-baseline arm that may be chosen if the baseline arm is not chosen
- $f_{i,t}(s,a)$: working model for the true conditional mean $r_{i,t}(s,a)$
- $\tilde{R}^f_{i,t}(s,\bar{a})$: pseudo-reward given state $S_{i,t} = s$ and potential arm $\bar{a}$, which has the same expectation as the differential reward, $\Delta_{i,t}(s_{i,t}, \bar{a}_{i,t})$; this term is written as $\tilde{R}^f_{i,t}$ in some points in the main paper, with the state and action implied
- $\Delta^f_{i,t}(s,\bar{a}) := f_{i,t}(s,\bar{a}) - f_{i,t}(s,0)$: doubly robust estimator for the differential reward
- $I_m(t) \subseteq \{1, ..., t\}$: the m-th fold, of M total folds, as assigned up to time t. $I^{\complement}_m(t)$ denotes its complement
- λ: parameter for ridge penalization, used in $\lambda||\theta||_2^2$

- $\tilde{\sigma}^2_{i,t} = \pi_{i,t}(0|s_{i,t}) \cdot (1 - \pi_{i,t}(0|s_{i,t}))$: weights used in the penalized regression estimation, which are inversely proportional to $var(\tilde{R}^f_{i,t})$
- $\left\{d(i,j) := ||\theta_i - \theta_j||_2^2\right\}_{i \neq j}$: L_2-distances for network construction
- $G = (V, E)$: graph with nodes V and edges E; each node corresponds to an individual, $V := [N]$, and $(i, j) \in E$ for the smallest $M << N$ distances
- Q: incidence matrix where the element $Q_{v,e}$ corresponds to the v-th vertex (individual) and e-th edge
- L: Laplacian
- M: max number of neighbors per participant and time step
- $\mathcal{O}_k = \{(i,t) : i \leq k \,\&\, t \leq k + 1 - i\}$: the set of observed time points across all individuals at stage k of the sequential requirement setting
- G_{user}: nearest-neighbor graph characterizing proximity in the user domain
- G_{time}: nearest-neighbor graph characterizing proximity in the time domain item $\Theta_k = \text{vec}[(\theta, \theta_1^{\text{user}}, \ldots, \theta_k^{\text{user}}, \theta_1^{\text{time}}, \ldots, \theta_k^{\text{time}})] \in \mathbb{R}^{p(2k+1)}$: the set of all parameters, including the individual-specific time-invariant parameters and the shared time-specific parameters, at stage k of the sequential recruitment setting
- D_{user}, D_{time}, B_{user}, B_{time}, M: terms for bound on parameters and mean estimate

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract offers a brief summary of the paper's motivation and contribution, while the introduction delves deeper into background details and highlights the key components of our proposed algorithm.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In the Discussion section (Section 7), we reflect on limitations related to computation, modeling assumptions, hyper-parameter tuning, and practical implementation.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We present all assumptions clearly, each numbered for reference, and we provide the full set of assumptions in Theorem 1. Additionally, the proof is available in the Appendix, and within the main text, we indicate where the corresponding proof can be found.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: In the manuscript, we provided a clear description of the proposed algorithm in Algorithm 2. Additionally, we elaborated on the key components of the algorithm in Section 4. Furthermore, in Section 6.1, we outlined the simulation setup, with detailed information available in Appendix A.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility.

In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The primary simulation results can be entirely reproduced using the provided code on Github. However, the Valentine study data, used for our real-world analysis, is not publicly accessible.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Algorithm 2 requires data splits, and we devoted two paragraphs on Page 5 to discussing options for this split. Additionally, we clearly listed all of the hyperparameters involved in forming objective functions and provided guidance on how to make suitable choices for them.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: In Section 6.2, the experiment results include hypothesis testing, and we have clearly indicated the p-values in the Figure legend to denote statistical significance.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: At the start of Section 6, we offer a paragraph outlining the computer resources required to reproduce the experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The author has obtained permission to access both the Valentine study and the Intern Health Study dataset. It's important to note that the experiments conducted in this paper are purely numerical simulations and offline evaluation of existing data, which do not involve any human research subjects or participants.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: This paper is driven by the current challenges in mobile health policy learning, aiming to enhance policy learning to maximize the benefits of mobile interventions. As a result, we emphasize the societal impacts, focusing on how the algorithm has the potential to improve public health.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We have cited the associated papers that produced the code package and dataset in Section 6.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The Appendix includes a link to our GitHub repository that contains the code to reproduce our simulation study.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.