ReBandit: Random Effects Based Online RL Algorithm for Reducing Cannabis Use

Susobhan Ghosh[1] , **Yongyi Guo**[2] , **Pei-Yao Hung**[3] , **Lara Coughlin**[4] , **Erin Bonar**[4] ,
Inbal Nahum-Shani[3] , **Maureen Walton**[4] and **Susan Murphy**[1]
[1]Department of Computer Science, Harvard University
[2]Department of Statistics, University of Wisconsin-Madison
[3]Institute for Social Research, University of Michigan
[4]Department of Psychiatry, University of Michigan
susobhan_ghosh@g.harvard.edu, guo98@wisc.edu, peiyaoh@umich.edu, laraco@med.umich.edu,
erinbona@med.umich.edu, inbal@umich.edu, waltonma@med.umich.edu, samurphy@g.harvard.edu

Abstract

The escalating prevalence of cannabis use, and associated cannabis-use disorder (CUD), poses a significant public health challenge globally. With a notably wide treatment gap, especially among emerging adults (EAs; ages 18-25), addressing cannabis use and CUD remains a pivotal objective within the 2030 United Nations Agenda for Sustainable Development Goals (SDG). In this work, we develop an online reinforcement learning (RL) algorithm called reBandit which will be utilized in a mobile health study to deliver personalized mobile health interventions aimed at reducing cannabis use among EAs. reBandit utilizes *random effects* and *informative Bayesian priors* to learn quickly and efficiently in noisy mobile health environments. Moreover, reBandit employs Empirical Bayes and optimization techniques to autonomously update its hyper-parameters online. To evaluate the performance of our algorithm, we construct a simulation testbed using data from a prior study, and compare against commonly used algorithms in mobile health studies. We show that reBandit performs equally well or better than all the baseline algorithms, and the performance gap widens as population heterogeneity increases in the simulation environment, proving its adeptness to adapt to diverse population of study participants.

1 Introduction & Motivation

Addressing at-risk substance use, including cannabis use, is a pivotal objective within the 2030 UN Agenda for Sustainable Development Goals (SDG)[1]. Within this agenda, SDG 3 focuses on ensuring healthy lives and well-being across the lifespan, yet, increasing use of cannabis, third in global prevalence after alcohol and nicotine, threatens this goal [Peacock *et al.*, 2018]. Hence, as highlighted in target 3.5 of the agenda, strengthening the prevention and treatment of cannabis use and cannabis use disorder (CUD) is crucial. Unfortunately, this coincides with a decreased public perception of the risks associated with cannabis use, likely influenced by ongoing decriminalization efforts and greater access to cannabis products [Carliner *et al.*, 2017], further worsened by one of the largest treatment gaps of any medical condition, with one study showing only 5% of those with CUD receiving treatment [Lapham *et al.*, 2019].

In the US, the prevalence of cannabis use is highest among emerging adults (EAs; age 18-25) [SAMHSA, 2023], marking it as a significant concern within the growing landscape of cannabis use. Particularly worrisome is the fact that early initiation of cannabis use links to an array of physical and mental health repercussions, as well as escalated risk for developing CUD [Volkow *et al.*, 2014; Hall, 2009; Chan *et al.*, 2021; Hasin *et al.*, 2016]. Given that cannabis use frequently commences during adolescence and peaks in emerging adulthood, this is a critical developmental period for early intervention strategies to prevent transitions into CUD.

Mobile health technologies, such as health apps and sensors, can potentially serve as support tools to help individuals manage their cannabis use. Using these tools, individuals can track their cannabis consumption, receive personalized interventions, and provide objective data for early detection of issues. These technologies enable the delivery of just-in-time adaptive interventions (JITAIs) [Nahum-Shani *et al.*, 2018], which leverage rapidly changing information about a person's state and context to decide whether and how to intervene in daily life. JITAIs have been successful for many domains of behavioral health [Jaimes *et al.*, 2015; Clarke *et al.*, 2017; Golbus *et al.*, 2021], whilst JITAIs for cannabis use among EAs are currently lacking evidence despite promising early data [Shrier *et al.*, 2018].

In this work, we develop an RL algorithm called reBandit which will be utilized in the MiWaves pilot study (Section 1.1). MiWaves focuses on developing a JITAI for reducing cannabis use among emerging adults (EAs) (ages 18-25). This JITAI leverages reBandit to determine the likelihood of delivering an intervention message.

[1] https://sdgs.un.org/2030agenda

1.1 MiWaves Pilot Study

The MiWaves pilot study focuses on developing a *personalizing* Just-In-Time Adaptive Intervention (pJITAI), namely a JITAI that integrates an RL algorithm. In this study, EAs are randomized to receive a mobile-based intervention message or no message, twice daily. The RL algorithm is designed to learn from a participant's history, and *personalize* the likelihood of intervention delivery based on a participant's current context. By combining technology, behavioral science, and data-driven decision-making, MiWaves aims to empower emerging adults with the digital tools to help reduce their cannabis use. The MiWaves pilot study has been registered on ClinicalTrials.gov (NCT05824754), and is scheduled to start in **March 2024**. Figure 1 provides a visual overview of the MiWaves pilot study.

1.2 Challenges, Contributions and Overview

Deploying RL algorithms in mHealth studies like MiWaves present a multitude of challenges that must be addressed, which include:

C1 **Limited Data**: Many sequential decision making problems in mHealth involve scarce data, forcing RL algorithms to learn and perform well under strict data constraints [Trella *et al.*, 2022].

C2 **After-study analysis and evaluation:** The RL algorithms deployed in mHealth studies need to be developed in a way to facilitate after-study analysis and off-policy evaluation.

C3 **Autonomy and Stability**: The intervention protocol in clinical studies is pre-specified. Since the RL algorithm is part of the intervention, scientists do not have the flexibility to change the RL algorithm while the study is running. RL algorithms must exhibit robustness in the face of noisy data, ensuring consistent and reliable performance throughout the study [Trella *et al.*, 2022].

C4 **Explainability**: It is imperative that RL algorithms are interpretable and comprehensible to behavioral scientists and medical professionals to enhance their ability to critique RL performance and to enhance the possibility of larger scale implementation.

C5 **Delayed Effects**: In mobile health studies, each intervention message sent to the user has a *delayed effect*. Users may perceive burden upon receiving an intervention message, which influences their future behavior.

C6 **Reproducibility**: Any algorithm used as part of the intervention in a clinical study needs to be reproducible in order for health scientists to evaluate and implement the intervention package in practice. Hence, the decisions taken by the RL algorithm must be reproducible, allowing for scrutiny and verification of their effectiveness.

To that end, we introduce reBandit, an online RL algorithm which utilizes *random effects* to address the challenges mentioned above. When used as part of an RL algorithm, random effects allow the algorithm to learn quickly and efficiently by making use of other participant's data in the population while simultaneously personalizing treatment for a given participant. Moreover, reBandit employs an informative Bayesian prior formulated from pre-existing data to act as a warm-start. Carefully designed priors incorporate previous (domain) knowledge, which help algorithms learn quickly and efficiently. Both *random effects* and informative *priors* can help reBandit to tackle challenge **C1**.

The most commonly used RL algorithms in mHealth settings are *bandit* algorithms. In mHealth settings, predictions of the value of next state (eg. using [Jiang *et al.*, 2015]) can be very noisy. Bandit algorithms are, thus, preferred due to their performance in such noisy environments. Moreover, they are computationally less complex, and hence are able to run stably and reliably in an online environment. Further, linear models are often considered interpretable due to their simplicity of representing the role of various factors, and can also be stably updated. reBandit utilizes both these concepts - it uses a bandit framework, along with a linear model (with random effects) to model the reward. We derive the formula to update reBandit's parameters and hyper-parameters online (Sec 4.1). We show that we are able to autonomously update these parameters and hyper-parameters within a reasonable time-limit in an online environment. Moreover, to facilitate after-study analysis, we utilize a smooth variant of posterior sampling, and clip the probabilities of taking an action (Sec 4.2). This way, reBandit is able to overcome challenges **C2**, **C3** and **C4**.

To address delayed effects (**C5**), one can use RL algorithms to model a full Markov Decision Process (MDP). However, in mobile health settings, such approaches are not feasible due to limited data and noisy outcomes [Trella *et al.*, 2023]. On the other hand, the classical bandit framework alone is also insufficient, as it is designed to optimize for immediate reward, and thus, cannot account for the delayed effects of actions. To that end, we engineer the reward used to update reBandit's parameters and hyper-parameters (Sec 4.3), to account for delayed effects of actions.

Finally, to tackle challenge **C6**, we have made our implementation of reBandit publicly available [2] To ensure reproducibility, we employ a seeded pseudo-random number generator to make every stochastic decision in reBandit. Additionally, all intermediate results and values used to make decisions are programatically stored in a database for reproducing the results obtained during any *run* of the algorithm.

2 RL Framework and Notation

This section provides a brief overview of the Reinforcement Learning (RL) [Sutton and Barto, 2018] setup used in this work, and the specifics of the RL setup with respect to the MiWaves pilot study.

We approximate the pilot study environment as a bandit environment. We represent it as a Markov Decision Process (MDP) where in the RL algorithm (eg. the mobile app) interacts with the environment (eg. the user). The MDP is specified by the tuple $\langle \mathcal{S}, \mathcal{A}, r, P, T \rangle$, where $\mathcal{S}$ is the state-space of the algorithm, $\mathcal{A}$ is the action-space, $r(s, a)$ is the reward function defined for a given state $s \in \mathcal{S}$ and action $a \in \mathcal{A}$, $P(s, a, s')$ is the transition function for a given state

[2] https://github.com/StatisticalReinforcementLearningLab/miwaves_rl_service

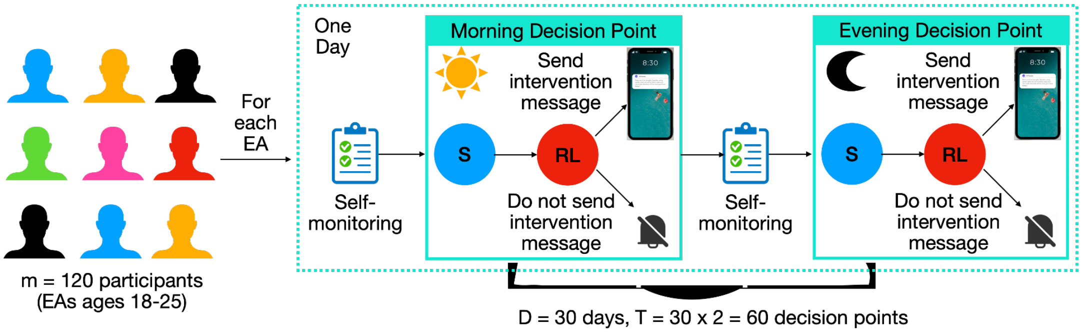
m = 120 participants
(EAs ages 18-25)
For each EA
One Day
Self-monitoring
Morning Decision Point
Send intervention message
S
RL
Do not send intervention message
Self-monitoring
Evening Decision Point
Send intervention message
S
RL
Do not send intervention message
D = 30 days, T = 30 x 2 = 60 decision points

Figure 1: Summary of the MiWaves pilot study. $m = 120$ EAs are expected to be recruited through social media ads. Each EA will be in the trial for 30 days, and will be asked to self-report twice daily - once in the morning and once in the evening. Upon completion or time expiration of the self-reporting, the RL algorithm will decide whether to send or not send an intervention message.

$s \in \mathcal{S}$, action $a \in \mathcal{A}$ and next state $s' \in \mathcal{S}$, and T is the total number of decision times. A user trajectory is given by $\mathcal{H}_i^{(T+1)} = \{S^{(t)}, a^{(t)}, R^{(t)}\}_{t=1}^{T}$, where $S^{(t)}$ denotes the state at decision time t, $a^{(t)}$ the action assigned by the RL algorithm at time t, and $R^{(t)}$ the reward collected after selection of the action. In the case of MiWaves we have:

Actions Binary action space, i.e. $\mathcal{A} = \{0, 1\}$ - to not send (0) or to send (1) an intervention message.

Decision points T decision points per user. The study is set to run for $D = 30$ days, and each day is supposed to have 2 decision points per day. Therefore, we expect to have 60 decision points per user, i.e. $T = 60$.

Reward We denote the reward for user i at decision time t by $R_i^{(t)}$. For the MiWaves pilot study, we have discrete rewards $\{0, 1, 2, 3\}$, which increase linearly with user engagement. We utilize engagement as our reward because engagement is critical to assess effectiveness of interventions after the study is over [Nahum-Shani *et al.*, 2022].

States Let us denote the state observation of the user i at decision time t as $S_i^{(t)}$. A given state $S = (S_1, S_2, S_3)$ is defined as a 3-tuple of the following binary variables (omitting the user and time index for brevity):

- S_1: Recent engagement - set to 1 if the average of past 3 observed rewards is greater than or equal to 2 (high engagement), and set to 0 otherwise (low engagement). At decision point $t = 1$, we set S_1 to 0, as there is no engagement by the user at the start of the pilot study.
- S_2: Time of day of the decision point - Morning (0) vs. Evening (1).
- S_3: Recent cannabis use - set to 0 if the participant reported using cannabis during their self-monitoring, and 1 otherwise. If the user fails to self-report, we set S_3 to be 0 because we expect the participant in the MiWaves pilot study to be using cannabis regularly (at least 3 times a week).

Overall, we represent the favorable states as 1 (not using cannabis, high engagement), and the unfavorable states as 0 (using cannabis, low engagement).

Number of users We expect $m = 120$ users to participate during the RL-powered MiWaves pilot study.

3 Related Work

Random effects (and mixed-effects) models have been well-studied in the statistical literature [Laird, 2004; Laird and Ware, 1982; Robinson, 1991], mainly in the context of batch data analysis. Mixed-effects models comprise of fixed and *random effects* - hence termed mixed effects. Laird and Ware introduce the notion of random-effects models for longitudinal data, and describe an unified approach to fitting such models using empirical Bayes and maximum likelihood estimation using EM algorithm. Our work (which is in context of streaming / real-time data) draws inspiration from Laird and Ware to extend random-effects based models to real-time decision making through RL in sequential decision making problems.

There has been a myriad of works in optimizing intervention delivery in mHealth settings in recent years [Golbus *et al.*, 2021; Kramer *et al.*, 2019; Rabbi *et al.*, 2019; Trella *et al.*, 2022; Walsh and Groarke, 2019]. *Bandit* algorithms are the most commonly used RL algorithms used in such high stakes online settings [Langford and Zhang, 2007; Tewari and Murphy, 2017; Wang *et al.*, 2005] due to their simplicity and stability, and ability to perform in noisy environments. Such algorithms have mainly used one of two approaches. The first approach is person specific (a.k.a. fully personalized) [Forman *et al.*, 2019; Jaimes *et al.*, 2015; Liao *et al.*, 2019; Rabbi *et al.*, 2015] where a separate model is deployed for each user in the trial. This approach is suitable when the population of users are highly heterogeneous, but suffers greatly when data is scarce and/or noisy. Note that fully personalized approaches are not feasible for the MiWaves pilot study, due to scarce data (the study runs for only 30 days). The second approach completely pools data (a.k.a. fully pooled) across all users in the population [Clarke *et al.*, 2017; Paredes *et al.*, 2014; Trella *et al.*, 2022; Yom-Tov *et al.*, 2017; Zhou *et al.*, 2018]. Our algorithm, reBandit, strikes a balance between the two approaches - it adaptively pools data across users depending on the degree of heterogeneity in the population. Section 4.1 describes how we achieve that balance using *random effects*.

Tomkins *et al.* also use random effects in their Thompson-Sampling [Russo and Van Roy, 2014; Thompson, 1933] contextual bandit algorithm [Li *et al.*, 2010], IntelligentPooling.

IntelligentPooling updates its hyper-parameters by modeling the problem as a Gaussian Process (GP). However, IntelligentPooling fails to run autonomously and stably in an online environment (does not overcome **C3**) [3]. Here we deal with this problem by updating the hyper-parameters in reBandit using empirical Bayes (similar to [Laird and Ware, 1982]), and solve the optimization problem using projected gradient descent. reBandit runs autonomously and stably in an online environment while also having more users (12x) and more features (8x) as compared to the environment described in IntelligentPooling.

Recently, various approaches have been explored regarding the application of mixed effects models within a bandit framework [Zhu and Kveton, 2022a; Zhu and Kveton, 2022b; Aouali *et al.*, 2023]. However, these works primarily focus on utilizing mixed effects to capture the dependence and heterogeneity of rewards associated with different actions, rather than addressing the similarity and heterogeneity among multiple users. For example, [Zhu and Kveton, 2022a] consider a (non-contextual) multi-arm bandit problem, where the agent chooses one of the K arms at each time t with the goal of maximizing cumulative reward. The authors assume that the reward for the arms are correlated with each other and can be expressed using a mixed-effects model, so that pulling an arm gives some information about the reward of other arms as well. A follow up work, [Zhu and Kveton, 2022b], adds context into the reward model, and results in a linear mixed effects model for the rewards of the arms. [Aouali *et al.*, 2023] further generalizes the above works to a non-linear reward setting. Our approach diverges from these studies by utilizing mixed-effects to model user similarity and heterogeneity, while making decisions for each user at each time point.

In the broader RL literature, there has been much work on Thompson Sampling based bandit algorithms [Basu *et al.*, 2021; Hong *et al.*, 2022], especially in connection to multi-task learning and meta learning [Peleg *et al.*, 2022; Simchowitz *et al.*, 2021; Wan *et al.*, 2021; Wan *et al.*, 2023]. The multi-task learning based approaches quantify the similarity between arms and/or users from their policies - the extent to which one user's data influences or contributes to another user's policy is a function of some similarity measure. reBandit can be connected to multi-task learning, as it learns across multiple users (or tasks), and tries to maximize rewards across all users (or tasks). However, due to its unique application in mobile health, reBandit adopts a distinct set of assumptions on the structure of the similarity measures in comparison to the works mentioned above. The meta-learning based approaches exploit the underlying structure of similar tasks to improve performance on new (or unseen, but similar) tasks. While reBandit can be viewed as a form of meta-learning, where shared population parameters are learnt across users (or tasks), and user-specific parameters are learnt to personalize to tasks, reBandit does not try to improve performance on new or unseen users.

4 Bandit Algorithm: reBandit

This section details details the reBandit algorithm used in the MiWaves pilot study. Being an online RL algorithm, reBandit has two major components: (i) the online learning algorithm; and (ii) the action-selection procedure. Going forward, we describe reBandit's online learning algorithm in Section 4.1, and it's posterior sampling based action selection strategy in Section 4.2. Finally, to address delayed effects, we describe its reward engineering procedure in Section 4.3. The reBandit algorithm is summarized in Algorithm 1.

4.1 Online Learning Algorithm

This section details the online learning algorithm - specifically the algorithm's reward approximating function and its model update procedure.

Reward Approximating Function

One of the key components of the online learning algorithm is its reward approximation function, through which it models the participant's reward. Recall that the reward function is the conditional mean of the reward given state and action. We chose a Bayesian Mixed Linear Model to model the reward. Mixed models allow the RL algorithm to adaptively pool and learn across users while simultaneously personalizing actions for each user.

Let us assume that for a given user i at decision time t, the RL algorithm receives the reward $R_i^{(t)}$ after taking action $a_i^{(t)}$ Then, the reward model is written as:

$$R_i^{(t)} = g(\boldsymbol{S_i^{(t)}})^T \boldsymbol{\alpha_i} + a_i^{(t)} f(\boldsymbol{S_i^{(t)}})^T \boldsymbol{\beta_i} + \epsilon_i^{(t)} \qquad (1)$$

where $\epsilon_i^{(t)}$ is the noise, assumed to be gaussian i.e. $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \sigma_\epsilon^2 \boldsymbol{I}_{mt})$, and m is the total number of users who have been or are currently part of the study. Also $\boldsymbol{\alpha_i}$, $\boldsymbol{\beta_i}$, and $\boldsymbol{\gamma_i}$ are weights that the algorithm wants to learn. $g(S)$ and $f(S)$ are functions of the RL state defined in Section 2. To enhance robustness to misspecification of the baseline reward model when $a_i^{(t)} = 0$, $g(\boldsymbol{S_i^{(t)}})^T \boldsymbol{\alpha_i}$, we utilize action-centering [Greenewald *et al.*, 2017] to learn an over-parameterized version of the above reward model:

$$\begin{aligned} R_i^{(t)} = {} & g(\boldsymbol{S_i^{(t)}})^T \boldsymbol{\alpha_i} + (a_i^{(t)} - \pi_i^{(t)}) f(\boldsymbol{S_i^{(t)}})^T \boldsymbol{\beta_i} \\ & + (\pi_i^{(t)}) f(\boldsymbol{S_i^{(t)}})^T \boldsymbol{\gamma_i} + \epsilon_i^{(t)} \end{aligned} \qquad (2)$$

where $\pi_i^{(t)}$ is the probability of taking action $a_i^{(t)} = 1$ in state $\boldsymbol{S_i^{(t)}}$ for participant i at decision time t. We refer to the term $g(\boldsymbol{S_i^{(t)}})^T \boldsymbol{\alpha_i}$ as the baseline, and $f(\boldsymbol{S_i^{(t)}})^T \boldsymbol{\beta_i}$ as the advantage (i.e. the advantage of taking action 1 over action 0).

We re-write the reward model as follows:

$$R_i^{(t)} = \boldsymbol{\Phi_{it}^T} \boldsymbol{\theta_i} + \epsilon_{i,t} \qquad (3)$$

where $\boldsymbol{\Phi_{it}^T} = \boldsymbol{\Phi}(\boldsymbol{S_i^{(t)}}, a_i^{(t)}, \pi_i^{(t)})^T = [g(\boldsymbol{S_i^{(t)}})^T, (a_i^{(t)} - \pi_i^{(t)}) f(\boldsymbol{S_i^{(t)}})^T, (\pi_i^{(t)}) f(\boldsymbol{S_i^{(t)}})^T]$ is the design matrix for given state and action, and $\boldsymbol{\theta_i} = [\boldsymbol{\alpha_i}, \boldsymbol{\beta_i}, \boldsymbol{\gamma_i}]^T$ is the joint weight vector that the algorithm wants to learn. We further break down the joint weight vector $\boldsymbol{\theta_i}$ into two components:

$$\boldsymbol{\theta_i} = \begin{bmatrix} \boldsymbol{\alpha}_i \\ \boldsymbol{\beta}_i \\ \boldsymbol{\gamma}_i \end{bmatrix} = \begin{bmatrix} \boldsymbol{\alpha}_{\text{pop}} + \boldsymbol{u}_{\alpha,i} \\ \boldsymbol{\beta}_{\text{pop}} + \boldsymbol{u}_{\beta,i} \\ \boldsymbol{\gamma}_{\text{pop}} + \boldsymbol{u}_{\gamma,i} \end{bmatrix} = \boldsymbol{\theta}_{\text{pop}} + \boldsymbol{u_i} \qquad (4)$$

[3] We were unable to run their code published on GitHub, and unable to parse the code or replicate it due to poor documentation

Here, $\boldsymbol{\theta}_{\text{pop}} = [\boldsymbol{\alpha}_{\text{pop}}, \boldsymbol{\beta}_{\text{pop}}, \boldsymbol{\gamma}_{\text{pop}}]^T$ is the population level term which is common across all the user's reward models and follows a normal prior distribution given by $\boldsymbol{\theta}_{\text{pop}} \sim \mathcal{N}(\boldsymbol{\mu}_{\text{prior}}, \boldsymbol{\Sigma}_{\text{prior}})$. On the other hand, $\boldsymbol{u_i} = [\boldsymbol{u}_{\alpha,i}, \boldsymbol{u}_{\beta,i}, \boldsymbol{u}_{\gamma,i}]^T$ are the individual level parameters, or the *random effects*, for any given user i. Note that the individual level parameters are assumed to be normal by definition, i.e. $\boldsymbol{u_i} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma_u})$, and independent of ϵ_i. Refer to Appendix D [4] for more details.

Online Model Update Procedure

Posterior Update: We vectorize the terms across the m users in a study, and re-write the model as:

$$\boldsymbol{R} = \boldsymbol{\Phi^T \theta} + \boldsymbol{\epsilon} \tag{5}$$

$$\boldsymbol{R} = \left[\boldsymbol{R_1^T} \dots \boldsymbol{R_m^T}\right]^T, \quad \boldsymbol{R}_i = \left[R_i^{(1)} \dots R_i^{(t)}\right]^T \tag{6}$$

$$\boldsymbol{\theta} = \left[\boldsymbol{\theta}_1^T \dots \boldsymbol{\theta}_m^T\right]^T = \boldsymbol{1}_m \otimes \boldsymbol{\theta}_{\text{pop}} + \boldsymbol{u} \tag{7}$$

$$\boldsymbol{u} = \left[\boldsymbol{u_1^T} \dots \boldsymbol{u_m^T}\right]^T \tag{8}$$

$$\boldsymbol{\epsilon} = \left[\boldsymbol{\epsilon_1^T} \dots \boldsymbol{\epsilon_m^T}\right]^T, \quad \boldsymbol{\epsilon}_i = \left[\epsilon_{i,1} \dots \epsilon_{i,t}\right]^T \tag{9}$$

$$\boldsymbol{u}_i \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_u) \tag{10}$$

$$\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \sigma_\epsilon^2 \boldsymbol{I}_{mt}) \tag{11}$$

As specified before, we assume a gaussian prior on the population level term $\boldsymbol{\theta}_{\text{pop}} \sim \mathcal{N}(\boldsymbol{\mu}_{\text{prior}}, \boldsymbol{\Sigma}_{\text{prior}})$. The hyper-parameters of the above model, given the definition above, are the noise variance σ_ϵ^2 and the random effects variance $\boldsymbol{\Sigma_u}$. Now, at a given decision point t, using estimated values of the hyper-parameters ($\sigma_{\epsilon,t}^2$ is the estimate of σ_ϵ^2 and $\boldsymbol{\Sigma_{u,t}}$ is the estimate of $\boldsymbol{\Sigma_u}$), the posterior mean and covariance matrix of the parameter $\boldsymbol{\theta}$ can be calculated as:

$$\boldsymbol{\mu}_{\text{post}}^{\boldsymbol{(t)}} = \left(\boldsymbol{\tilde{\Sigma}_{\theta,t}^{-1}} + \sigma_{\epsilon,t}^{-2}\boldsymbol{A}\right)^{-1}\left(\boldsymbol{\tilde{\Sigma}_{\theta,t}^{-1}\mu_\theta} + \sigma_{\epsilon,t}^{-2}\boldsymbol{B}\right) \tag{12}$$

$$\boldsymbol{\Sigma}_{\text{post}}^{\boldsymbol{(t)}} = \left(\boldsymbol{\tilde{\Sigma}_{\theta,t}^{-1}} + \sigma_{\epsilon,t}^{-2}\boldsymbol{A}\right)^{-1} \tag{13}$$

where

$$\boldsymbol{A} = \text{BlockDiag}\big(\boldsymbol{A_1}, \dots, \boldsymbol{A_m}\big), \quad \boldsymbol{A_i} = \sum\nolimits_{\tau=1}^{t} \boldsymbol{\Phi_{i\tau}\Phi_{i\tau}^T} \tag{14}$$

$$\boldsymbol{B^T} = [\boldsymbol{B_1^T} \ \dots \ \boldsymbol{B_m^T}], \quad \boldsymbol{B_i} = \sum\nolimits_{\tau=1}^{t} \boldsymbol{\Phi_{i\tau}} R_i^{(\tau)} \tag{15}$$

$$\boldsymbol{\mu_\theta^T} = [\boldsymbol{\mu}_{\text{prior}}{}^T \ \dots \ \boldsymbol{\mu}_{\text{prior}}{}^T] \tag{16}$$

$$\boldsymbol{\tilde{\Sigma}_{\theta,t}} = \mathbb{I}_m \otimes \boldsymbol{\Sigma_{u,t}} + \mathbb{J}_m \otimes \boldsymbol{\Sigma}_{\text{prior}} \tag{17}$$

The action-selection procedure (described in Section 4.2) uses the Gaussian posterior distribution defined by the posterior mean $\boldsymbol{\mu}_{\text{post}}^{\boldsymbol{(t)}}$ and variance $\boldsymbol{\Sigma}_{\text{post}}^{\boldsymbol{(t)}}$ to determine the action selection probability $\pi^{(t+1)}$ and the corresponding actions for the next time steps.

Hyper-parameter Update: The hyper-parameters in the algorithm's reward model are the noise variance σ_ϵ^2 and random effects variance $\boldsymbol{\Sigma_u}$. In order to update these variance estimates at the end of decision time t, we use Empirical Bayes [Morris, 1983] to maximize the marginal likelihood of observed rewards, marginalized over the parameters $\boldsymbol{\theta}$. So, in order to form $\boldsymbol{\Sigma_{u,t}}$ and $\sigma_{\epsilon,t}^2$, we solve the following optimization problem:

$$\boldsymbol{\Sigma_{u,t}}, \sigma_{\epsilon,t}^2 = \operatorname{argmax} l(\boldsymbol{\Sigma_{u,t}}, \sigma_{\epsilon,t}^2; \mathcal{H}_{1:m}^{(t)}) \tag{18}$$

$$\text{s.t.} \quad \boldsymbol{\Sigma_{u,t}} \succ 0, \quad \sigma_{\epsilon,t}^2 \geq 0 \tag{19}$$

where,

$$\begin{aligned} l(\boldsymbol{\Sigma_{u,t}}, \sigma_{\epsilon,t}^2; \mathcal{H}) &= \log(\det(\boldsymbol{X})) - \log(\det(\boldsymbol{X} + y\boldsymbol{A})) \\ &+ mt\log(y) - y\sum\nolimits_{\tau\in[t]}\sum\nolimits_{i\in[m]}(R_i^{(\tau)})^2 - \boldsymbol{\mu_\theta^T X \mu_\theta} \\ &+ (\boldsymbol{X\mu_\theta} + y\boldsymbol{B})^T(\boldsymbol{X} + y\boldsymbol{A})^{-1}(\boldsymbol{X\mu_\theta} + y\boldsymbol{B}) \end{aligned} \tag{20}$$

Note that, $\boldsymbol{X} = \boldsymbol{\tilde{\Sigma}_{\theta,t}^{-1}}$ (see Eq. 17) and $y = \sigma_{\epsilon,t}^{-2}$. We solve the optimization problem using gradient descent.

4.2 Action Selection Procedure

The action selection procedure utilizes a modified posterior sampling algorithm called the smooth posterior sampling algorithm. Recall from Section 4.1, our model for the reward is a Bayesian linear mixed model with action centering (refer Eq. 2) where $\pi_i^{(t)}$ is the probability that the RL algorithm selects action $a_i^{(t)} = 1$ in state $\boldsymbol{S_i^{(t)}}$ for participant i at decision point t. The RL algorithm computes the probability $\pi_i^{(t)}$ as follows:

$$\pi_i^{(t)} = \mathbb{E}_{\tilde{\beta}\sim\mathcal{N}(\mu_{\text{post},i}^{(t-1)}, \Sigma_{\text{post},i}^{(t-1)})}[\rho(f(\boldsymbol{S_i^{(t)}})^T\boldsymbol{\tilde{\beta}})|\mathcal{H}_{1:m}^{(t)}, \boldsymbol{S_i^{(t)}}] \tag{21}$$

Notice that the last expectation above is over the draw of β from the posterior distribution parameterized by $\boldsymbol{\mu}_{\text{post},\boldsymbol{i}}^{\boldsymbol{(t-1)}}$ and $\boldsymbol{\Sigma}_{\text{post},\boldsymbol{i}}^{\boldsymbol{(t-1)}}$ (see Eq. 12 and Eq. 13 for their definitions).

Classical posterior sampling sets $\rho(x) = \mathbb{I}(x > 0)$. In this case, the posterior sampling algorithm sets randomization probabilities to the posterior probability that the treatment effect is positive. However, when using a pooled algorithm, Zhang *et al.* showed that between study statistical inference is enhanced if ρ is a *smooth* i.e. continuously differentiable function. Using a smooth function ensures that the randomization probabilities formed by the algorithm concentrate. Concentration enhances the replicability of the randomization probabilities if the study is repeated. Without concentration, the randomization probabilities might fluctuate greatly between repetitions of the study [Deshpande *et al.*, 2018; Kalvit and Zeevi, 2021; Zhang *et al.*, 2022]. In MiWaves , we choose ρ to be a generalized logistic function, defined as follows (details in Appendix E [4]) :

$$\rho(x) = L_{\min} + \frac{L_{\max} - L_{\min}}{1 + c\exp(-bx)} \tag{22}$$

where $c = 5$, and $b = 21$. We set the lower and upper clipping probabilities as $L_{\min} = 0.2$ and $L_{\max} = 0.8$ (i.e., $0.2 \leq \pi_i^{(t)} \leq 0.8$). The probabilities are clipped to facilitate after-study analysis and off-policy evaluation [Zhang *et al.*, 2022].

[4] https://arxiv.org/abs/2402.17739

Algorithm 1: reBandit

```
Input  : $m$, $D$, $\boldsymbol{\mu}^{(0)}_{\text{post}} = \boldsymbol{\mu}_{\text{prior}}$, $\boldsymbol{\Sigma}^{(0)}_{\text{post}} = \boldsymbol{\Sigma}_{\text{prior}}$, $\boldsymbol{\Sigma}_{\boldsymbol{u},\boldsymbol{0}}$, $\sigma^2_{\epsilon,0}$,
         $\rho(x)$
for $d = 1$ to $D$ do
    for $j = 0$ to $1$ do
        Compute timestep $\tau = ((d-1) \times 2) + j$
        for $i = 1$ to $m$ do
            Observe state $\boldsymbol{S}^{(\tau)}_i$;
            Get posteriors $\boldsymbol{\mu}^{(d-1)}_{\text{post},i}$ and $\boldsymbol{\Sigma}^{(d-1)}_{\text{post},i}$ for user $i$
              from $\boldsymbol{\mu}^{(d-1)}_{\text{post}}$ and $\boldsymbol{\Sigma}^{(d-1)}_{\text{post}}$;
            Compute action selection probability $\pi^{(\tau)}_i$
              using Eq. 21
            Sample action $a^{(t)}_i = \text{Bern}(\pi^{(\tau)}_i)$
            Collect reward $R^{(\tau)}_i$
        end
    end
    Update $\boldsymbol{\Sigma}_{\boldsymbol{u},\boldsymbol{d}}$ and $\sigma^2_{\epsilon,d}$ using Eq. 18, with engineered
      rewards from Eq. 23;
    Update posteriors $\boldsymbol{\mu}^{(d)}_{\text{post}}$ and $\boldsymbol{\Sigma}^{(d)}_{\text{post}}$ using Eq. 12 and Eq.
      13, with engineered rewards from Eq. 23
end
```

4.3 Reward Engineering

To account for delayed effects in the bandit framework, we engineer the reward for the RL algorithm. Note that this engineered reward is only utilized to update the RL algorithm's parameters and hyper-parameters. We are still interested in maximizing the reward defined in Sec. 2, and use it to evaluate the algorithm's performance. The engineered reward $\hat{R}^{(t)}_i$ for user i at decision time t is defined as:

$$\hat{R}^{(t)}_i = R^{(t)}_i - a^{(t)}_i \text{cost}(a^{(t)}_i) \tag{23}$$

$$\text{cost}(a^{(t)}_i) = \lambda \cdot \sigma_{i,\text{obs}} \tag{24}$$

where $\sigma_{i,\text{obs}}$ is the standard deviation of the observed rewards for a given user i, and λ is a tuned non-negative hyper-parameter. Note that the reward is not penalized when $a^{(t)}_i = 0$. Intuitively, the cost function is designed to allow the RL algorithm to optimize for user engagement, while simultaneously accounting for the delayed effect of sending an intervention message, i.e. $a^{(t)}_i = 1$.

5 Experimental Results

In this section, we detail the design of a simulation testbed (Sec. 5.1) to help evaluate the performance of our algorithm. Our experimental setup and the corresponding results are discussed in Sec 5.2.

5.1 Simulation Testbed Design

We leverage data from the SARA [Rabbi *et al.*, 2018] study, which trialed an mHealth app aimed at sustaining engagement of substance use data collection from participants. Since the SARA study focused on a similar demographic of EAs as the MiWaves pilot study, it appears ideal for constructing a simulation testbed. However, note that this data is impoverished. SARA had only 1 decision point per day, as compared to 2 per day in MiWaves . The goal of the messages sent to the participants in SARA was to increase survey completion in order to collect substance use data. In contrast, the goal of sending intervention messages in MiWaves pilot study is to reduce the participant's cannabis use through self-monitoring and mobile health engagement. Moreover, the daily cannabis use data in SARA was collected retro-actively at the end of each week, which often resulted in participant's noisy recollection of their cannabis use, and had missing cannabis use data if the participant chose to not respond. In contrast, participants in MiWaves are asked to self-report twice daily, which reduces the amount of missing data if they fail to self-report once. We construct a *base dataset* of 42 users after cleaning and imputing the SARA data (please refer to appendix A.1[4] for more details).

Base Model For the base model of the environment, we fit Multinomial Logistic Regression (MLR) models on each of the 42 users in the base dataset. The learnt weights include weights for the baseline (when action is 0), and the advantage (added to the baseline when action is 1). These user models are overfit to learn the user behavior as well capture the noise in the environment. We choose MLR for our user models, as it is interpretable, and performs similar in comparison to a generic neural network (see Appendix A.6 [4]) .

Varying Treatment Effects (TE) The effect of the intervention message on a particular user is measured by their unique treatment effect size. Given that the intervention messages in SARA had minimal treatment effect [Nahum-Shani *et al.*, 2021], we introduce higher levels of treatment effect into the user models by augmenting their weights. Higher levels of treatment effect increase the likelihood of obtaining higher rewards when taking action 1. To that end, we construct TE = *low* and TE = *high* treatment effect variants for each MLR user model. Refer to Appendix A.9 [4] for more details.

Modeling Habituation (HB) To account for delayed effects in the environment, we introduce user habituation to repeated stimuli (multiple intervention messages sent to the user in a short span of time) by adding a negative effect in the baseline weights of the MLR user models. To that end, we define *dosage* for each user at each decision point as the weighted average of the number of intervention messages received in the previous six decision points. The weights are decreased with each past decision point, reflecting a diminishing impact of older intervention messages received by the user. Next, we impute baseline weights for dosage in the MLR user models in a way that higher dosage (more messages received) leads to higher likelihood of generating low rewards, and vice-versa. Note that this procedure simulates how the user may experience habituation; if the RL algorithm does not send many interventions to a user experiencing habituation, the user may dis-habituate and recover their baseline behavior. We construct two environment variants - HB = *low* and HB = *high* habituation effect - by varying the baseline weights for dosage. Additionally, we simulate the proportion of users who can experience habituation within a population - set at either $P = 50\%$ or $P = 100\%$. Please refer to Appendix A.9 [4] for more details.

Alg.	Minimal Treatment Effect					Low Treatment Effect				
	HB=None	HB=Low		HB=High		HB=None	HB=Low		HB=High	
		P=50%	P=100%	P=50%	P=100%		P=50%	P=100%	P=50%	P=100%
reBandit	128.54±0.18	127.23±0.18	126.01±0.18	123.22±0.19	119.55±0.20	129.44±0.17	128.11±0.17	126.80±0.18	123.74±0.19	120.12±0.20
BLR	127.78±0.16	126.60±0.18	125.78±0.18	123.23±0.19	119.60±0.20	129.10±0.17	127.85±0.17	126.53±0.18	123.75±0.19	120.16±0.20
random	127.83±0.16	126.52±0.18	125.22±0.18	119.03±0.21	110.29±0.23	128.97±0.17	127.70±0.17	126.45±0.18	120.49±0.20	112.06±0.22

Table 1: Average total reward per user per simulated trial, averaged across 500 simulated trials and 120 users per trial, along with their 95% confidence intervals (CI) for minimal and low treatment effect settings. HB refers to the level of habituation in the environment, while P is used to denote the proportion of the population who can experience habituation.

Alg.	HB=None	HB=Low		HB=High	
		P=50%	P=100%	P=50%	P=100%
reBandit	132.25±0.16	130.95±0.17	129.71±0.17	124.70±0.19	121.19±0.20
BLR	132.21±0.16	130.94±0.17	129.63±0.17	124.70±0.19	121.22±0.19
random	131.05±0.17	129.88±0.17	128.71±0.17	123.19±0.20	115.39±0.22

Table 2: Average total reward per user per simulated trial along with their 95% CIs for the high treatment effect settings.

5.2 Simulation Results

We construct 15 simulation environment variants using a combination of techniques described in Sec. 5.1. For each environment, we simulate 500 studies with $m = 120$ users each, over a period of $D = 30$ days ($T = 60$). The $m = 120$ users are drawn with replacement from the 42 MLR user models learnt using SARA data.

We compare the performance of our algorithm to two common approaches in mobile health studies. First, is a full pooling algorithm called BLR. BLR utilizes Bayesian Linear Regression [Liao *et al.*, 2019] to pool data and learn a single model across all the users in a study, and select actions according to the action selection procedure mentioned in Sec. 4.2. We use engineered rewards (Sec. 4.3) to update BLR's parameters and hyper-parameters. We also update BLR hyper-parameters using Empirical Bayes, similar to the approach described in Sec. 4.1, for a fair comparison. For both reBandit and BLR, we update the posteriors at the end of each simulated day (every 2 decision points), and the hyper-parameters at the end of each week (every 14 decision points). We refer the reader to Appendix B[4] for more details about BLR's implementation. In addition to BLR, we also compare against the random algorithm, which utilizes an action selection probability of $\pi_i^{(t)} = 0.5$.

For each algorithm and simulation environment pair, we calculate the average total reward per user per simulated trial, averaged across the 500 simulated trials and 120 users in each trial. We also compute their 95% confidence intervals (CIs). We summarize our findings in Table 1 and 2. In all the simulation environments, reBandit performs no worse than the baseline algorithms. We highlight the environments where reBandit significantly outperforms other algorithms (CIs do not overlap) in *green*. In the environments where the CIs for the average total reward overlap for reBandit and BLR, we individually compare each of the 500 seeded simulations, and count the number of times reBandit achieved an average total reward per user as compared to BLR. If this number is greater than 50% of the simulations, i.e. greater than 250, we highlight those environments in *yellow*, otherwise they are highlighted in *blue*.

The primary takeaway from our simulation results in Tables 1 and 2 is that reBandit is impactful - it performs better than BLR in most environments, and even in the blue highlighted environments where it performs slightly worse than BLR, the performance is still comparable. It is important to note that our procedures to artificially introduce treatment effects and user habituation into the user models reduces the heterogeneity among the user models. This is due to the fact that our procedure to artificially inject treatment effect establishes a non-negative effect of taking an action across all the users in the user models. The same applies to the procedure for introducing user habituation, as it establishes a clear negative effect with respect to dosage across all users in the user models. However, in practice, higher levels of treatment effect or user habituation effect may lead to more heterogeneity in the population. Given that limitation, it is easy to observe that in our simulations, as levels of treatment effect or user habituation effect are increased, the performance gap between reBandit and BLR decreases. In simulation environments characterized by more pronounced heterogeneity due to lower levels of treatment and/or habituation effects, reBandit excels by adeptly identifying and leveraging the heterogeneity within the user population to personalize the likelihood of intervention message delivery and accrues greater rewards.

6 Conclusion

In this paper, we introduced reBandit, an online RL algorithm which will be a part of the upcoming mobile health study named MiWaves aimed at reducing cannabis use among emerging adults. We addressed the unique challenges inherent in mobile health studies, including limited data, and requirement for algorithmic autonomy and stability, while designing reBandit. We showed that reBandit utilizes *random-effects* and *informative Bayesian priors* to learn quickly and efficiently in noisy environments which are common in mobile health studies. The introduction of random effects allows reBandit to leverage the heterogeneity in the study population and deliver personalized interventions. To benchmark our algorithm, we detailed the design of a simulation testbed using prior data, and showed that reBandit performs equally well or better than two common approaches used in mobile health studies. In the future, we aim to analyze the effectiveness of the interventions in the MiWaves pilot study. In addition, we aim to investigate the contribution of an individual's data and the study population data towards learning the individual's parameters in the random effects model (see Appendix F[4]).

Acknowledgements

Research reported in this paper was supported by NIH/NIDA P50DA054039, and NIH/NIBIB and OD P41EB028242. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

References

[Aouali *et al.*, 2023] Imad Aouali, Branislav Kveton, and Sumeet Katariya. Mixed-effect thompson sampling. In *International Conference on Artificial Intelligence and Statistics*, pages 2087–2115. PMLR, 2023.

[Basu *et al.*, 2021] Soumya Basu, Branislav Kveton, Manzil Zaheer, and Csaba Szepesvári. No regrets for learning the prior in bandits. *Advances in neural information processing systems*, 34:28029–28041, 2021.

[Carliner *et al.*, 2017] Hannah Carliner, Qiana L Brown, Aaron L Sarvet, and Deborah S Hasin. Cannabis use, attitudes, and legal status in the us: A review. *Preventive medicine*, 104:13–23, 2017.

[Chan *et al.*, 2021] Gary CK Chan, Denise Becker, Peter Butterworth, Lindsey Hines, Carolyn Coffey, Wayne Hall, and George Patton. Young-adult compared to adolescent onset of regular cannabis use: A 20-year prospective cohort study of later consequences. *Drug and Alcohol Review*, 40(4):627–636, 2021.

[Clarke *et al.*, 2017] Shanice Clarke, Luis G Jaimes, and Miguel A Labrador. mstress: A mobile recommender system for just-in-time interventions for stress. In *2017 14th IEEE annual consumer communications & networking conference (CCNC)*, pages 1–5. IEEE, 2017.

[Deshpande *et al.*, 2018] Yash Deshpande, Lester Mackey, Vasilis Syrgkanis, and Matt Taddy. Accurate inference for adaptive linear models. In *International Conference on Machine Learning*, pages 1194–1203. PMLR, 2018.

[Forman *et al.*, 2019] Evan M Forman, Stephanie G Kerrigan, Meghan L Butryn, Adrienne S Juarascio, Stephanie M Manasse, Santiago Ontañón, Diane H Dallal, Rebecca J Crochiere, and Danielle Moskow. Can the artificial intelligence technique of reinforcement learning use continuously-monitored digital data to optimize treatment for weight loss? *Journal of behavioral medicine*, 42:276–290, 2019.

[Golbus *et al.*, 2021] Jessica R Golbus, Walter Dempsey, Elizabeth A Jackson, Brahmajee K Nallamothu, and Predrag Klasnja. Microrandomized trial design for evaluating just-in-time adaptive interventions through mobile health technologies for cardiovascular disease. *Circulation: Cardiovascular Quality and Outcomes*, 14(2):e006760, 2021.

[Greenewald *et al.*, 2017] Kristjan Greenewald, Ambuj Tewari, Susan Murphy, and Predag Klasnja. Action centered contextual bandits. *Advances in neural information processing systems*, 30, 2017.

[Hall, 2009] Wayne Hall. The adverse health effects of cannabis use: what are they, and what are their implications for policy? *International Journal of drug policy*, 20(6):458–466, 2009.

[Hasin *et al.*, 2016] Deborah Hasin, Bradley Kerridge, Tulshi Saha, Boji Huang, Roger Pickering, Sharon Smith, Jeesun Jung, Haitao Zhang, and Bridget Grant. Prevalence and correlates of dsm-5 cannabis use disorder, 2012-2013: Findings from the national epidemiologic survey on alcohol and related conditions–iii. *American Journal of Psychiatry*, 173(6):588–599, 2016.

[Hong *et al.*, 2022] Joey Hong, Branislav Kveton, Manzil Zaheer, and Mohammad Ghavamzadeh. Hierarchical bayesian bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 7724–7741. PMLR, 2022.

[Jaimes *et al.*, 2015] Luis G Jaimes, Martin Llofriu, and Andrew Raij. Preventer, a selection mechanism for just-in-time preventive interventions. *IEEE Transactions on Affective Computing*, 7(3):243–257, 2015.

[Jiang *et al.*, 2015] Nan Jiang, Alex Kulesza, Satinder Singh, and Richard Lewis. The dependence of effective planning horizon on model accuracy. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 1181–1189, 2015.

[Kalvit and Zeevi, 2021] Anand Kalvit and Assaf Zeevi. A closer look at the worst-case behavior of multi-armed bandit algorithms. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 8807–8819. Curran Associates, Inc., 2021.

[Kramer *et al.*, 2019] Jan-Niklas Kramer, Florian Künzler, Varun Mishra, Bastien Presset, David Kotz, Shawna Smith, Urte Scholz, Tobias Kowatsch, et al. Investigating intervention components and exploring states of receptivity for a smartphone app to promote physical activity: protocol of a microrandomized trial. *JMIR research protocols*, 8(1):e11540, 2019.

[Laird and Ware, 1982] Nan Laird and James Ware. Random-effects models for longitudinal data. *Biometrics*, pages 963–974, 1982.

[Laird, 2004] Nan Laird. Random effects and the linear mixed model. In *Analysis of Longitudinal and Cluster-Correlated Data*, volume 8, pages 79–96. Institute of Mathematical Statistics, 2004.

[Langford and Zhang, 2007] John Langford and Tong Zhang. The epoch-greedy algorithm for contextual multi-armed bandits. *Advances in neural information processing systems*, 20(1):96–1, 2007.

[Lapham *et al.*, 2019] Gwen T Lapham, Cynthia I Campbell, Bobbi Jo H Yarborough, Rulin C Hechter, Brian K Ahmedani, Irina V Haller, Andrea H Kline-Simon, Derek D Satre, Amy M Loree, Constance Weisner, et al. The prevalence of healthcare effectiveness data and information set (hedis) initiation and engagement in treatment among patients with cannabis use disorders in 7 us health systems. *Substance Abuse*, 40(3):268–277, 2019.

[Li *et al.*, 2010] Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, 2010.

[Liao *et al.*, 2019] Peng Liao, Kristjan H. Greenewald, Predrag V. Klasnja, and Susan A. Murphy. Personalized heartsteps: A reinforcement learning algorithm for optimizing physical activity. *CoRR*, abs/1909.03539, 2019.

[Morris, 1983] Carl N Morris. Parametric empirical bayes inference: theory and applications. *Journal of the American statistical Association*, 78(381):47–55, 1983.

[Nahum-Shani *et al.*, 2018] Inbal Nahum-Shani, Shawna N Smith, Bonnie J Spring, Linda M Collins, Katie Witkiewitz, Ambuj Tewari, and Susan A Murphy. Just-in-time adaptive interventions (jitais) in mobile health: key components and design principles for ongoing health behavior support. *Annals of Behavioral Medicine*, pages 1–17, 2018.

[Nahum-Shani *et al.*, 2021] Inbal Nahum-Shani, Mashfiqui Rabbi, Jamie Yap, Meredith L Philyaw-Kotov, Predrag Klasnja, Erin E Bonar, Rebecca M Cunningham, Susan A Murphy, and Maureen A Walton. Translating strategies for promoting engagement in mobile health: A proof-of-concept microrandomized trial. *Health Psychology*, 40(12):974, 2021.

[Nahum-Shani *et al.*, 2022] Inbal Nahum-Shani, Steven D Shaw, Stephanie M Carpenter, Susan A Murphy, and Carolyn Yoon. Engagement in digital interventions. *American Psychologist*, 2022.

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

[Peacock *et al.*, 2018] Amy Peacock, Janni Leung, Sarah Larney, Samantha Colledge, Matthew Hickman, Jürgen Rehm, Gary A Giovino, Robert West, Wayne Hall, Paul Griffiths, et al. Global statistics on alcohol, tobacco and illicit drug use: 2017 status report. *Addiction*, 113(10):1905–1926, 2018.

[Peleg *et al.*, 2022] Amit Peleg, Naama Pearl, and Ron Meir. Metalearning linear bandits by prior update. In *International Conference on Artificial Intelligence and Statistics*, pages 2885–2926. PMLR, 2022.

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

[Rabbi *et al.*, 2018] Mashfiqui Rabbi, Meredith Philyaw Kotov, Rebecca Cunningham, Erin E Bonar, Inbal Nahum-Shani, Predrag Klasnja, Maureen Walton, Susan Murphy, et al. Toward increasing engagement in substance use data collection: development of the substance abuse research assistant app and protocol for a microrandomized trial using adolescents and emerging adults. *JMIR research protocols*, 7(7):e9850, 2018.

[Rabbi *et al.*, 2019] Mashfiqui Rabbi, Predrag Klasnja, Tanzeem Choudhury, Ambuj Tewari, and Susan Murphy. Optimizing mhealth interventions with a bandit. *Digital Phenotyping and Mobile Sensing: New Developments in Psychoinformatics*, pages 277–291, 2019.

[Robinson, 1991] George K Robinson. That blup is a good thing: the estimation of random effects. *Statistical science*, pages 15–32, 1991.

[Russo and Van Roy, 2014] Daniel Russo and Benjamin Van Roy. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39(4):1221–1243, 2014.

[SAMHSA, 2023] SAMHSA. 2021 NSDUH Detailed Tables. https://www.samhsa.gov/data/report/2021-nsduh-detailed-tables, 2023. Accessed: 2024-02-14.

[Shrier *et al.*, 2018] Lydia A Shrier, Pamela J Burke, Meredith Kells, Emily A Scherer, Vishnudas Sarda, Cassandra Jonestrask, Ziming Xuan, and Sion Kim Harris. Pilot randomized trial of moment, a motivational counseling-plus-ecological momentary intervention to reduce marijuana use in youth. *Mhealth*, 4, 2018.

[Simchowitz *et al.*, 2021] Max Simchowitz, Christopher Tosh, Akshay Krishnamurthy, Daniel Hsu, Thodoris Lykouris, Miro Dudik, and Robert Schapire. Bayesian decision-making under misspecified priors with applications to meta-learning. *Advances in Neural Information Processing Systems*, 34:26382–26394, 2021.

[Sutton and Barto, 2018] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[Tewari and Murphy, 2017] Ambuj Tewari and Susan A Murphy. From ads to interventions: Contextual bandits in mobile health. *Mobile health: sensors, analytic methods, and applications*, pages 495–517, 2017.

[Thompson, 1933] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285–294, 1933.

[Tomkins *et al.*, 2021] Sabina Tomkins, Peng Liao, Predrag Klasnja, and Susan Murphy. Intelligentpooling: Practical thompson sampling for mhealth. *Machine learning*, 110(9):2685–2727, 2021.

[Trella *et al.*, 2022] Anna Trella, Kelly Zhang, Inbal Nahum-Shani, Vivek Shetty, Finale Doshi-Velez, and Susan Murphy. Designing reinforcement learning algorithms for digital interventions: pre-implementation guidelines. *Algorithms*, 15(8):255, 2022.

[Trella *et al.*, 2023] Anna L. Trella, Kelly W. Zhang, Inbal Nahum-Shani, Vivek Shetty, Finale Doshi-Velez, and Susan A. Murphy. Reward design for an online reinforcement learning algorithm supporting oral self-care. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(13):15724–15730, Sep. 2023.

[Volkow *et al.*, 2014] Nora D Volkow, Ruben D Baler, Wilson M Compton, and Susan RB Weiss. Adverse health effects of marijuana use. *New England Journal of Medicine*, 370(23):2219–2227, 2014.

[Walsh and Groarke, 2019] Jane C Walsh and Jenny M Groarke. Integrating behavioral science with mobile (mhealth) technology to optimize health behavior change interventions. *European Psychologist*, 2019.

[Wan *et al.*, 2021] Runzhe Wan, Lin Ge, and Rui Song. Metadata-based multi-task bandits with bayesian hierarchical models. *Advances in Neural Information Processing Systems*, 34:29655–29668, 2021.

[Wan *et al.*, 2023] Runzhe Wan, Lin Ge, and Rui Song. Towards scalable and robust structured bandits: A meta-learning framework. In *International Conference on Artificial Intelligence and Statistics*, pages 1144–1173. PMLR, 2023.

[Wang *et al.*, 2005] Chih-Chun Wang, Sanjeev R Kulkarni, and H Vincent Poor. Bandit problems with side observations. *IEEE Transactions on Automatic Control*, 50(3):338–355, 2005.

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

[Zhang *et al.*, 2022] Kelly W Zhang, Lucas Janson, and Susan A Murphy. Statistical inference after adaptive sampling for longitudinal data. *arXiv preprint arXiv:2202.07098*, 2022.

[Zhou *et al.*, 2018] Mo Zhou, Yonatan Mintz, Yoshimi Fukuoka, Ken Goldberg, Elena Flowers, Philip Kaminsky, Alejandro Castillejo, and Anil Aswani. Personalizing mobile fitness apps using reinforcement learning. In *CEUR workshop proceedings*, volume 2068. NIH Public Access, 2018.

[Zhu and Kveton, 2022a] Rong Zhu and Branislav Kveton. Random effect bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 3091–3107. PMLR, 2022.

[Zhu and Kveton, 2022b] Rong Zhu and Branislav Kveton. Robust contextual linear bandits. *arXiv preprint arXiv:2210.14483*, 2022.