Group-Driven Reinforcement Learning for Personalized mHealth Intervention

Feiyun Zhu[1], Jun Guo[2], Zheng Xu[1], Peng Liao[2], Liu Yang[4], and Junzhou Huang[1,3(✉)]

[1] Department of CSE, University of Texas at Arlington, Arlington, TX 76013, USA
[2] Department of Statistics, Univeristy of Michigan, Ann Arbor, MI 48109, USA
[3] Tencent AI Lab, Shenzhen 518057, China
jzhuang@uta.edu
[4] School of Software, Central South University, Changsha 410075, Hunan, China

Abstract. Due to the popularity of smartphones and wearable devices nowadays, mobile health (mHealth) technologies are promising to bring positive and wide impacts on people's health. State-of-the-art decision-making methods for mHealth rely on some ideal assumptions. Those methods either assume that the users are completely homogenous or completely heterogeneous. However, in reality, a user might be similar with some, but not all, users. In this paper, we propose a novel group-driven reinforcement learning method for the mHealth. We aim to understand how to share information among similar users to better convert the limited user information into sharper learned RL policies. Specifically, we employ the K-means clustering method to group users based on their trajectory information similarity and learn a shared RL policy for each group. Extensive experiment results have shown that our method can achieve clear gains over the state-of-the-art RL methods for mHealth.

1 Introduction

In the wake of the vast population of smart devices (smartphones and wearable devices such as the Fitbit Fuelband and Jawbone etc.) users worldwide, mobile health (mHealth) technologies become increasingly popular among the scientist communities. The goal of mHealth is to use smart devices as great platforms to collect and analyze raw data (weather, location, social activity, stress, etc.). Based on that, the aim is to provide in-time interventions to device users according to their ongoing status and changing needs, helping users to lead healthier lives, such as reducing the alcohol abuse [4] and the obesity management [11].

Formally, the tailoring of mHealth intervention is modeled as a sequential decision making (SDM) problem. It aims to learn the optimal decision rule to decide when, where and how to deliver interventions [7,10,13,17] to best serve

This work was partially supported by NSF IIS-1423056, CMMI-1434401, CNS-1405985, IIS-1718853 and the NSF CAREER grant IIS-1553687.

A. F. Frangi et al. (Eds.): MICCAI 2018, LNCS 11070, pp. 590–598, 2018.
https://doi.org/10.1007/978-3-030-00928-1_67

users. This is a brand-new research topic. Currently, there are two types of reinforcement learning (RL) methods for mHealth with distinct assumptions: (a) the off-policy, batch RL [16,17] assumes that all users in the mHealth are completely homogenous: they share all information and learn an identical RL for all the users; (b) the on-policy, online RL [7,17] assumes that all users are completely different: they share no information and run a separate RL for each user. The above assumptions are good as a start for the mHealth study. However, when mHealth are applied to more practical situations, they have the following drawbacks: (a) the off-policy, batch RL method ignore the fact that the behavior of all users may be too complicated to be modeled with an identical RL, which leads to potentially large biases in the learned policy; (b) for the on-policy, online RL method, an individual user's trajectory data is hardly enough to support a separate RL learning, which is likely to result in unstable policies that contain lots of variances [14].

A more realistic assumption lies between the above two extremes: a user may be similar to some, but not all, users and similar users tend to have similar behaviors. In this paper, we propose a novel group driven RL for the mHealth. It is in an actor-critic setting [3]. The core idea is to find the similarity (cohesion) network for the users. Specifically, we employ the clustering method to mine the group information. Taking the group information into consideration, we learn K (i.e., the number of groups) shared RLs for K groups of users respectively; each RL learning procedure makes use of all the data in that group. Such implementation balances the conflicting goals of reducing the complexity of data while enriching the number of samples for each RL learning process.

2 Preliminaries

The Markov Decision Process (MDP) provides a mathematical tool to model the dynamic system [2,3]. It is defined as a 5-tuple $\{\mathcal{S}, \mathcal{A}, P, R, \gamma\}$, where $\mathcal{S}$ is the state space and $\mathcal{A}$ is the action space. The state transition model $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \mapsto [0, 1]$ indicates the probability of transiting from one state s to another s' under a given action a. $\mathcal{R} : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ is the corresponding reward, which is assumed to be bounded over the state and action spaces. $\gamma \in [0, 1)$ is a discount factor that reduces the influence of future rewards. The stochastic policy $\pi(\cdot \mid s)$ determines how the agent acts with the system by providing each state s with a probability over all the possible actions. We consider the parameterized stochastic policy, i.e., $\pi_\theta(a \mid s)$, where θ is the unknown coefficients.

Formally, the quality of a policy π_θ is evaluated by a value function $Q^{\pi_\theta}(s, a) \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{A}|}$ [12]. It specifies the total amount of rewards an agent can achieve when starting from state s, first choosing action a and then following the policy π_θ. It is defined as follows [3]:

$$Q^{\pi_\theta}(s, a) = \mathbb{E}_{a_i \sim \pi_\theta, s_i \sim \mathcal{P}} \left\{ \sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, a_t) \mid s_0 = s, a_0 = a \right\}. \tag{1}$$

The goal of various RL methods is to learn an optimal policy π_{θ^*} that maximizes the Q-value for all the state-action pairs [2]. The objective is $\pi_{\theta^*} = \arg\max_\theta \widehat{J}(\theta)$ (such procedure is called the actor updating [3]), where

$$\widehat{J}(\theta) = \sum_{s\in\mathcal{S}} d_{\text{ref}}(s) \sum_{a\in\mathcal{A}} \pi_\theta(a \mid s) Q^{\pi_\theta}(s,a), \tag{2}$$

where $d_{\text{ref}}(s)$ is a reference distribution over states; $Q^{\pi_\theta}(s,a)$ is the value for the parameterized policy π_θ. It is obvious that we need the estimation of $Q^{\pi_\theta}(s,a)$ (i.e. the critic updating) to determine the objective function (2).

3 Cohesion Discovery for the RL Learning

Suppose we are given a set of N users; each user is with a trajectory of T points. Thus in total, we have $NT = N \times T$ tuples summarized in $\mathcal{D} = \{\mathcal{D}_n \mid n = 1, \cdots, N\}$ for all the N users, where $\mathcal{D}_n = \{\mathcal{U}_i \mid i = 1, \cdots, T\}$ summarizes all the T tuples for the n-th user and $\mathcal{U}_i = (s_i, a_i, r_i, s_i')$ is the i-th tuple in $\mathcal{D}_n$.

3.1 The Pooled-RL and Separate RL (Separ-RL)

The first RL method (i.e. Pooled-RL) assumes that all the N users are completely homogenous and following the same MDP; they share all information and learn an identical RL for all the users [16]. In this setting, the critic updating (with an aim of seeking for solutions to satisfy the Linear Bellman equation [2,3]) is

$$\mathbf{w} = f(\mathbf{w}) = \arg\min_{\mathbf{h}} \frac{1}{|\mathcal{D}|} \sum_{\mathcal{U}_i\in\mathcal{D}} \left\| \mathbf{x}(s_i,a_i)^\intercal \mathbf{h} - \left[r_i + \gamma \mathbf{y}(s_i';\theta)^\intercal \mathbf{w} \right] \right\|_2^2 + \zeta_c \|\mathbf{h}\|_2^2, \tag{3}$$

where $\mathbf{w} = f(\mathbf{w})$ is a fixed point problem; $|\mathcal{D}|$ represents the number of tuples in $\mathcal{D}$; $\mathbf{x}_i = \mathbf{x}(s_i,a_i)^\intercal$ is the value feature at the time point i; $\mathbf{y}_i = \mathbf{y}(s_i';\theta) = \sum_{a\in\mathcal{A}} \mathbf{x}(s_i',a)\pi_\theta(a \mid s_i')$ is the feature at the next time point; ζ_c is a tuning parameter. The least-square temporal difference for Q-value (LSTDQ) [5,6] provides a closed-form solver for (3) as follows

$$\widehat{\mathbf{w}} = \left(\zeta_c \mathbf{I} + \frac{1}{|\mathcal{D}|} \sum_{\mathcal{U}_i\in\mathcal{D}} \mathbf{x}_i (\mathbf{x}_i - \gamma \mathbf{y}_i)^\intercal \right)^{-1} \left(\frac{1}{|\mathcal{D}|} \sum_{\mathcal{U}_i\in\mathcal{D}} \mathbf{x}_i r_i \right). \tag{4}$$

As $d_{\text{ref}}(s)$ is generally unavailable, the T-trial objective for (2) is defined as

$$\hat{\theta} = \arg\max_\theta \frac{1}{|\mathcal{D}|} \sum_{\mathcal{U}_i\in\mathcal{D}} \sum_{a\in\mathcal{A}} Q(s_i,a;\widehat{\mathbf{w}}) \pi_\theta(a|s_i) - \frac{\zeta_a}{2} \|\theta\|_2^2, \tag{5}$$

where $Q(s_i,a;\widehat{\mathbf{w}}) = \mathbf{x}(s_i,a)^\intercal \widehat{\mathbf{w}}$ is the newly defined Q-value which is based on the critic updating result in (4); ζ_a is the tuning parameter to prevent overfitting.

In case of large feature spaces, one can iteratively update $\widehat{\mathbf{w}}$ via (4) and $\widehat{\theta}$ in (5) to reduce the computational cost.

The Pooled-RL works well when all the N users are very similar. However, there are great behavior discrepancies among users in the mHealth study because they have different ages, races, incomes, religions, education levels etc. Such case makes the current Pooled-RL too simple to simultaneously fit all the N different users' behaviors. It easily results in lots of biases in the learned value and policy.

The second RL method (Separ-RL), such as Lei's online contextual bandit for mHealth [7,15], assumes that all users are completely heterogeneous. They share no information and run a separate online RL for each user. The objective functions are very similar with (3), (4), (5). This method should be great when the data for each user is very large in size. However, it generally costs a lot of time and other resources to collect enough data for the Separ-RL learning. Taking the HeartSteps for example, it takes 42 days to do the trial, which only collects 210 tuples per user. What is worse, there are missing and noises in the data, which will surely reduce the effective sample size. The problem of small sample size will easily lead to some unstable policies that contain lots of variances.

3.2 Group driven RL learning (Gr-RL)

We observe that users in mHealth are generally similar with some (but not all) users in the sense that they may have some similar features, such as age, gender, race, religion, education level, income and other socioeconomic status [8]. To this end, we propose a group based RL for mHealth to understand how to share information across similar users to improve the performance. Specifically, the users are assumed to be grouped together and likely to share information with others in the same group. The main idea is to divide the N users into K groups, and learn a separate RL model for each group. The samples of users in a group are pooled together, which not only ensures the simplicity of the data for each RL learning compared with that of the Pooled-RL, but also greatly enriches the samples for the RL learning compared with that of the Separ-RL, with an average increase of $(N/K-1)\times 100\%$ on sample size (cf. Sect. 3.1).

To cluster the N users, we employ one of the most benchmark clustering method, i.e., K-means. The behavior information (i.e. states and rewards) in the trajectory is processed as the feature. Specifically, the T tuples of a user are stacked together $\mathbf{z}_n = [s_1, r_1, \cdots, s_T, r_T]^\intercal$. With this new feature, we have the objective for clustering as $J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{nk}\left\|\mathbf{z}_n - \boldsymbol{\mu}_k\right\|^2$, where $\boldsymbol{\mu}_k$ is the k-th cluster center and $r_{nk} \in \{0,1\}$ is the binary indicator variable that describes which of the K clusters the data $\mathbf{z}_n$ belongs to. After the clustering step, we have the group information $\{\mathcal{G}_k \mid k = 1, \cdots, K\}$, each of which includes a set of similar users. With the clustering results, we have the new objective for the critic updating as $\mathbf{w}_k = f(\mathbf{w}_k) = \mathbf{h}_k^*$ for $k = 1, \cdots K$, where $\mathbf{h}_k^*$ is estimated as

$$\min_{\{\mathbf{h}_k | k=1,\cdots,K\}} \sum_{k=1}^{K}\left\{\frac{1}{|\mathcal{G}_k|}\sum_{\mathcal{U}_i\in\mathcal{G}_k}\left\|\mathbf{x}_i^\intercal\mathbf{h}_k - (r_i + \gamma\mathbf{y}_i^\intercal\mathbf{w}_k)\right\|_2^2 + \zeta_c\left\|\mathbf{h}_k\right\|_2^2\right\}, \quad (6)$$

which could be solved via the LSTDQ. The objective for the actor updating is

$$\max_{\{\theta_k|k=1,\cdots,K\}} \sum_{k=1}^{K} \left\{ \frac{1}{|\mathcal{G}_k|} \sum_{\mathcal{U}_i \in \mathcal{G}_k} \sum_{a \in \mathcal{A}} Q\left(s_i, a; \widehat{\mathbf{w}}_k\right) \pi_{\theta_k}\left(a|s_i\right) - \frac{\zeta_a}{2} \left\|\theta_k\right\|_2^2 \right\}. \tag{7}$$

The objectives (6) and (7) could be solved independently for each cluster. By properly setting the value of K, we could balance the conflicting goal of reducing the discrepancy between connected users while increasing the number of samples for each RL learning: (a) a small K is suited for the case where T is small and the users are generally similar; (b) while a large K is adapted to the case where T is large and users are generally different from others. Besides, we find that the proposed method is a generalization of the conventional Pooled-RL and Separ-RL: (a) when $K = 1$, the proposed method is equivalent to the Pooled-RL; (b) when $K = N$, our method is equivalent to the Separ-RL.

4 Experiments

There are three RL methods for comparison: (a) the Pooled-RL that pools the data across all users and learn an identical policy [16,17] for all the users; (b) the Separ-RL, which learns a separate RL policy for each user by only using his or her data [7]; (c) The group driven RL (Gr-RL) is the proposed method.

The HeartSteps dataset is used in the experiment. It is a 42-days trial study where there are 50 participants. For each participant, 210 decision points are collected—five decisions per participant per day. At each time point, the set of intervention actions can be the intervention type, as well as whether or not to send interventions. The intervention is sent via smartphones, or via wearable devices like a wristband [1]. In our study, there are two choices for a policy $\{0, 1\}$: $a = 1$ indicates sending the positive intervention, while $a = 0$ means no intervention [16,17]. Specifically, the parameterized stochastic policy is assumed to be in the form $\pi_\theta\left(a \mid s\right) = \frac{\exp[-\theta^\intercal \phi(s,a)]}{\sum_{a'} \exp[-\theta^\intercal \phi(s,a)]}$, where $\theta \in \mathbb{R}^q$ is the unknown variance and $\phi\left(\cdot,\cdot\right)$ is the feature processing method for the policy, i.e., $\phi\left(s,a\right) = [as^\intercal, a]^\intercal \in \mathbb{R}^m$, which is different from the feature for the value function $\mathbf{x}\left(s,a\right)$.

4.1 Experiments Settings

For the n^{th} user, a trajectory of T tuples $\mathcal{D}_n = \{(s_i, a_i, r_i)\}_{i=1}^T$ are collected via the micro-randomized trial [7,10]. The initial state is sampled from the Gaussian distribution $S_0 \sim \mathcal{N}_p\left\{0, \Sigma\right\}$, where Σ is the $p \times p$ covariance matrix with predefined elements. The policy of selecting action $a_t = 1$ is drawn from the random policy with a probability of 0.5 to provide interventions, i.e. $\mu\left(1 \mid s_t\right) = 0.5$ for all states s_t. For $t \geq 1$, the state and immediate reward are generated as follows

$$\begin{aligned}
S_{t,1} &= \beta_1 S_{t-1,1} + \xi_{t,1}, \\
S_{t,2} &= \beta_2 S_{t-1,2} + \beta_3 A_{t-1} + \xi_{t,2}, \\
S_{t,3} &= \beta_4 S_{t-1,3} + \beta_5 S_{t-1,3} A_{t-1} + \beta_6 A_{t-1} + \xi_{t,3}, \\
S_{t,j} &= \beta_7 S_{t-1,j} + \xi_{t,j}, \qquad \text{for } j = 4, \ldots, p
\end{aligned} \tag{8}$$

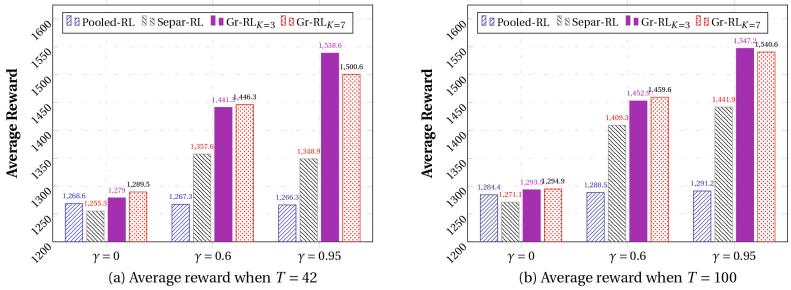

Fig. 1. Average reward of 3 RL methods: (a) Pooled-RL, (b) Separ-RL, (c) Gr-RL$_{K=3}$ and Gr-RL$_{K=7}$. The left sub-figure shows the results when the trajectory is short, i.e. $T = 42$; the right one shows the results when $T = 100$. A larger value is better.

$$R_t = \beta_{14} \times [\beta_8 + A_t \times (\beta_9 + \beta_{10} S_{t,1} + \beta_{11} S_{t,2}) + \beta_{12} S_{t,1} - \beta_{13} S_{t,3} + \varrho_t], \quad (9)$$

where $\boldsymbol{\beta} = \{\beta_i\}_{i=1}^{14}$ are the main parameters for the MDP; $\{\xi_{t,i}\}_{i=1}^{p} \sim \mathcal{N}(0, \sigma_s^2)$ is the noise in the state (9) and $\varrho_t \sim \mathcal{N}(0, \sigma_r^2)$ is the noise in the reward model (9). To mimic N users that are similar but not identical, we need N different $\boldsymbol{\beta}$s, each of which is similar with a set of others. Formally, there are two steps to obtain $\boldsymbol{\beta}$ for the i-th user: (a) select the m-th basic $\boldsymbol{\beta}$, i.e. $\boldsymbol{\beta}_m^{\text{basic}}$; it determines which group the i-th user belongs to; (b) add the noise $\boldsymbol{\beta}_i = \boldsymbol{\beta}_m^{\text{basic}} + \boldsymbol{\delta}_i$, for $i \in \{1, 2, \cdots, N_m\}$ to make each user different from others, where N_m indicates the number of users in the m-th group, $\boldsymbol{\delta}_i \sim \mathcal{N}(0, \sigma_b \mathbf{I}_{14})$ is the noise and $\mathbf{I}_{14} \in \mathbb{R}^{14 \times 14}$ is an identity matrix. The value of σ_b specifies how different the users are. Specially in our experiment, we set $M = 5$ groups (each group has $N_m = 10$ people, leading to $N = 50$ users involved in the experiment). The basic $\boldsymbol{\beta}$s for the M groups are set as follows

$$\begin{aligned}
\boldsymbol{\beta}_1^{\text{basic}} &= [0.40, 0.25, 0.35, 0.65, 0.10, 0.50, 0.22, 2.00, 0.15, 0.20, 0.32, 0.10, 0.45, 800] \\
\boldsymbol{\beta}_2^{\text{basic}} &= [0.45, 0.35, 0.40, 0.70, 0.15, 0.55, 0.30, 2.20, 0.25, 0.25, 0.40, 0.12, 0.55, 700] \\
\boldsymbol{\beta}_3^{\text{basic}} &= [0.35, 0.30, 0.30, 0.60, 0.05, 0.65, 0.28, 2.60, 0.35, 0.45, 0.45, 0.15, 0.50, 650] \\
\boldsymbol{\beta}_4^{\text{basic}} &= [0.55, 0.40, 0.25, 0.55, 0.08, 0.70, 0.26, 3.10, 0.25, 0.35, 0.30, 0.17, 0.60, 500] \\
\boldsymbol{\beta}_5^{\text{basic}} &= [0.20, 0.50, 0.20, 0.62, 0.06, 0.52, 0.27, 3.00, 0.15, 0.15, 0.50, 0.16, 0.70, 450],
\end{aligned}$$

Besides, the noises are set $\sigma_s = \sigma_r = 1$ and $\sigma_\beta = 0.01$. Other variances are $p = 3$, $q = 4$, $\zeta_a = \zeta_c = 0.01$. The feature processing for the value estimation $Q^{\pi_\theta}(s, a)$ is $\mathbf{x}(s, a) = [1, s^\intercal, a, s^\intercal a]^\intercal \in \mathbb{R}^{2p+2}$ for all the compared methods.

4.2 Evaluation Metric and Results

In the experiments, the expectation of long run average reward (ElrAR) $\mathbb{E}[\eta^{\pi_{\hat{\theta}}}]$ is proposed to evaluate the quality of a learned policy $\pi_{\hat{\theta}}$ [9,10]. Intuitively in

Table 1. The average reward of three RL methods when the discount factor γ changes from 0 to 0.95: (a) Pooled-RL, (b) Separ-RL, (c) Gr-RL$_{K=3}$ and Gr-RL$_{K=7}$. A larger value is better. The **bold value** is the best and the *blue italic value* is the 2nd best.

γ	Average reward ($T = 42$)			
	Pooled-RL	Separ-RL	Gr-RL$_K = 3$	Gr-RL$_K = 7$
0	1268.6 ± 68.2	1255.3 ± 62.3	*1279.0± 66.6*	**1289.5 ± 64.5**
0:2	1268.1 ± 68.3	1287.6 ± 76.8	*1318.3± 62.5*	**1337.3 ± 56.7**
0:4	1267.6 ± 68.4	1347.0 ± 54.1	*1368.8± 57.6*	**1389.7 ± 50.7**
0:6	1267.3 ± 68.5	1357.6 ± 57.9	*1441.3± 48.2*	**1446.3 ± 46.7**
0:8	1266.8 ± 68.7	1369.4 ± 51.6	**1513.9 ± 38.8**	*1484.0± 44.5*
0:95	1266.3 ± 68.7	1348.9 ± 53.4	**1538.6 ± 34.3**	*1500.6± 42.8*
Avg	1267.4	1327.6	**1410.0**	*1407.9*
γ	*Average reward ($T = 100$)*			
0	1284.4 ± 64.1	1271.1 ± 70.7	*1293.5± 62.1*	**1294.9 ± 63.7**
0:2	1285.8 ± 63.9	1301.2 ± 65.6	*1329.6± 58.5*	**1332.9 ± 58.7**
0:4	1287.1 ± 63.8	1370.1 ± 49.1	*1385.5± 52.1*	**1393.0 ± 49.2**
0:6	1288.5 ± 63.6	1409.3 ± 42.2	*1452.9± 44.3*	**1459.6 ± 40.9**
0:8	1289.9 ± 63.4	1435.0 ± 37.6	**1519.0 ± 39.5**	*1518.0± 38.5*
0:95	1291.2 ± 63.2	1441.9 ± 35.9	**1547.2 ± 37.2**	*1540.6± 38.1*
Avg	1287.8	1371.4	*1421.3*	**1423.2**

The value of γ specifies different RL methods: (a) $\gamma = 0$ means the contextual bandit [7]*,(b) $0 < \gamma < 1$ indicates the discounted reward RL.*

the HeartSteps application, ElrAR measures the average step a user could take each day when he or she is provided by the intervention via the learned policy $\pi_{\hat{\theta}}$. Specifically, there are two steps to achieve the ElrAR [10]: (a) get the $\eta^{\pi_{\hat{\theta}}}$ for each user by averaging the rewards over the last $4,000$ elements in the long run trajectory with a total number of $5,000$ tuples; (b) ElrAR $\mathbb{E}\left[\eta^{\pi_{\hat{\theta}}}\right]$ is achieved by averaging over the $\eta^{\pi_{\hat{\theta}}}$'s of all users.

The experiment results are summarized in Table 1 and Fig. 1, where there are three RL methods: (a) Pooled-RL, (b) Separ-RL, (c) Gr-RL$_{K=3}$ and Gr-RL$_{K=7}$. $K = 3, 7$ is the number of cluster centers in our algorithm, which is set different from the true number of groups $M = 5$. Such setting is to show that Gr-RL does not require the true value of M. There are two sub-tables in Table 1. The top sub-table summarizes the experiment results of three RL methods under six γ settings (i.e. the discount reward) when the trajectory is short, i.e. $T = 42$. While the bottom one displays the results when the trajectory is long, i.e. $T = 100$. Each row shows the results under one discount factor, $\gamma = 0, \cdots, 0.95$; the last row shows the average performance over all the six γ settings.

As we shall see, Gr-RL$_{K=3}$ and Gr-RL$_{K=7}$ generally perform similarly and are always among the best. Such results demonstrate that our method doesn't require the true value of groups and is robust to the value of K. In average, the

proposed method improves the ElrAR by 82.4 and 80.3 steps when $T = 42$ as well as 49.8 and 51.7 steps when $T = 100$, compared with the best result of the state-of-the-art methods, i.e. Separ-RL. There are two interesting observations: (1) the improvement of our method decreases as the trajectory length T increases; (2) when the trajectory is short, i.e. $T = 42$, it is better to set small Ks, which emphasizes the enriching of dataset; while the trajectory is long, i.e. $T = 100$, it is better to set large Ks to simplify the data for each RL learning.

5 Conclusions and Discussion

In this paper, we propose a novel group driven RL method for the mHealth. Compared with the state-of-the-art RL methods for mHealth, it is based on a more practical assumption that admits the discrepancies between users and assumes that a user should be similar with some (but not all) users. The proposed method is able to balance the conflicting goal of reducing the discrepancy between pooled users while increasing the number of samples for each RL learning. Extensive experiment results verify that our method gains obvious advantages over the state-of-the-art RL methods in the mHealth.

References

1. Dempsey, W., Liao, P., Klasnja, P., Nahum-Shani, I., Murphy, S.A.: Randomised trials for the fitbit generation. Significance **12**(6), 20–23 (2016)
2. Geist, M., Pietquin, O.: Algorithmic survey of parametric value function approximation. IEEE TNNLS **24**(6), 845–867 (2013)
3. Grondman, I., Busoniu, L., Lopes, G.A.D., Babuska, R.: A survey of actor-critic reinforcement learning: standard and natural policy gradients. IEEE Trans. Syst. Man Cybern. **42**(6), 1291–1307 (2012)
4. Gustafson, D.: A smartphone application to support recovery from alcoholism: a randomized clinical trial. JAMA Psychiatry **71**(5), 566–572 (2014)
5. Kolter, J.Z., Ng, A.Y.: Regularization and feature selection in least-squares temporal difference learning. In: International Conference on Machine Learning, pp. 521–528 (2009)
6. Lagoudakis, M.G., Parr, R.: Least-squares policy iteration. J. Mach. Learn. Res. **4**, 1107–1149 (2003)
7. Lei, H., Tewari, A., Murphy, S.: An actor-critic contextual bandit algorithm for personalized interventions using mobile devices. In: NIPS 2014 Workshop: Personalization: Methods and Applications, pp. 1–9 (2014)
8. Li, T., Levina, E., Zhu, J.: Prediction models for network-linked data. CoRR abs/1602.01192, February 2016
9. Liao, P., Tewari, A., Murphy, S.: Constructing just-in-time adaptive interventions. Ph.D. Section Proposal, pp. 1–49 (2015)
10. Murphy, S.A., Deng, Y., Laber, E.B., Maei, H.R., Sutton, R.S., Witkiewitz, K.: A batch, off-policy, actor-critic algorithm for optimizing the average reward. CoRR abs/1607.05047 (2016)
11. Patrick, K., Raab, F., Adams, M., Dillon, L., Zabinski, M., Rock, C., Griswold, W., Norman, G.: A text message-based intervention for weight loss: randomized controlled trial. J. Med. Internet Res. **11**(1), e1 (2009)

12. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction, 2nd edn. MIT Press, Cambridge (2012)
13. Xu, Z., Li, Y., Axel, L., Huang, J.: Efficient preconditioning in joint total variation regularized parallel MRI reconstruction. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015, Part II. LNCS, vol. 9350, pp. 563–570. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24571-3_67
14. Xu, Z., Wang, S., Zhu, F., Huang, J.: Seq2seq fingerprint: An unsupervised deep molecular embedding for drug discovery. In: ACM Conference on Bioinformatics, Computational Biology, and Health Informatics (2017)
15. Zhu, F., Guo, J., Li, R., Huang, J.: Robust actor-critic contextual bandit for mobile health (mhealth) interventions. arXiv preprint arXiv:1802.09714 (2018)
16. Zhu, F., Liao, P.: Effective warm start for the online actor-critic reinforcement learning based mhealth intervention. In: The Multi-disciplinary Conference on Reinforcement Learning and Decision Making, pp. 6–10 (2017)
17. Zhu, F., Liao, P., Zhu, X., Yao, Y., Huang, J.: Cohesion-driven online actor-critic reinforcement learning for mhealth intervention. arXiv:1703.10039 (2017)