Push Me: Optimizing Notification Timing to Promote Physical Activity

Scott Fleming
Stanford University
scottyf@stanford.edu

Gordon Blake
Stanford University
gblake@stanford.edu

Abstract

Mobile-based health interventions are increasingly taking advantage of the ubiquity of smarphones by using apps and notifications to promote healthy behavior such as physical activity. However, these reminders are often given at rote times, without regard to factors that may influence the user's responsiveness to them. We explore optimizing notification timing to promote physical activity through an MDP-based approach. Drawing on user activity and notification data from the *MyHeart Counts* app, we modeled an MDP with states capturing the time since the user was last active and last notified and rewards based on user activity and effective notifications in a given interval. Solving the MDP via maximum likelihood estimation and value iteration produced a policy of notification timing that substantially outperformed multiple baselines. These results suggest the promise of MDP models and reinforcement learning in the realm of mobile health interventions.

Introduction

Mobile technology is a pervasive part of modern life. The average user looks at their phone 47 times a day (Wigginton et al. 2018). Mobile-based health interventions take advantage of this ubiquity by using smartphone apps and notifications to promote healthy behavior, such as reminding users to exercise. However, these reminders are often given at rote times, without regard to factors that may influence whether the recipient sees and reacts to them. Because users are often inundated with notifications, appropriately timing prompts is crucial to ensuring the success of mobile-based health interventions. We seek to better optimize when these reminders are sent in order to best promote healthy behavior.

Drawing on notification and activity data from the *MyHeart Counts* app, we modeled the problem of optimal notification timing as a Markov Decision Process (MDP) which, when solved, would yield a policy of when to send push notifications to the app user in order to maximize their physical activity. There are a number of sources of uncertainty in the data which lend this objective to an MDP-based approach. These include (1) noisy and potentially biased measurements from the accelerometer; (2) ambiguity surrounding how raw accelerometer data map to discrete physical activities (i.e. walking, standing, running, etc.); and (3) inconsistency in patterns of user response to push notifications.

We defined the states in the MDP to be based on the time since the the user was last active and last notified during a discretized time interval. The reward was based on the proportion of time that the user was active during the interval and the effectiveness of a notification, if delivered. After transforming the *MyHeart Counts* data into (state, action, reward, new state) tuples, we performed maximum likelihood estimation of the transition probabilities between states and solved the resulting MDP using value iteration, obtaining a policy recommending at which states to deliver notifications. The derived policy significantly outperformed both random and heuristic baselines in the evaluation of its expected utility.

We also analyzed the results of a similar procedure when we expanded our state space to include activity profiles as well. We found that the optimal policy for sedentary individuals tended to more aggressively notify the user compared to the optimal policy for day workers and drivers. The optimal policy derived for this expanded state space MDP representation similarly performed substantially better than heuristic and random policy baselines. These findings offer hope for powerful, personalized activity coaching.

Past Work

Recent work by A Shcherbina et al. 2018 has demonstrated the effectiveness of mobile technology as a tool for improving overall physical activity. Physical activity, in turn, has been linked to a broad swathe of important health outcomes (McConnell et al. 2017). While much work on user engagement with mobile technology has focused on adjusting push notification scheduling to optimize user engagement with the app (see Tan et al. 2016), we focused on optimal policies for maximizing the overall physical activity of app users. To that end, this paper represents a novel contribution to the current literature on optimal policies for mobile health notification schedules.

Past models of optimal push notification times have relied on feeding feature sets of user activity to classifiers such as Naive Bayes and XGBoost (Morrison et al. 2017; Pielot et al. 2017). Work in the related field of customer relationship management suggests that an MDP or POMDP may be

an appropriate model for the problem. In particular, Ma and Zhang use a notion of delayed reward similar to our objective of maximizing long-term user activity (Ma and Zhang 2008).

Approach

After obtaining data on *MyHeart Counts* user activity and notifications, we transformed the information into state, action, and reward data, performed maximum likelihood estimation to approximate transition probabilities, and obtained a policy by solving the resulting MDP via value iteration. The policy was then evaluated by comparing its expected utility against the performance of several baseline policies.

The code used for analysis is available on the project GitHub Page. The Final_Pipeline notebook contains the code for the basic state space while Final_Pipeline_Activity_Group includes code for extracting and evaluating a policy on the expanded state space, which includes activity profiles.

Data

We obtained access to a large data set collected through the *MyHeart Counts* Cardiovascular Health Study. In total, 40,017 research participants agreed to share their physical activity data (as monitored by a small Core Motion coprocessor chip on their iPhones). In addition to granular physical activity information, the data included a record of the time at which certain activity coaching interventions were delivered. The intervention of interest was the "Hourly Stand Prompt" where, if an individual has been stationary (sitting) for the previous hour, they receive a notification telling them to stand up for 60 seconds. All data were collected between March 10, 2015 and October 28, 2015.

Preprocessing

The quantity of data recorded in the duration of the study was significant, but only a small subset was relevant for modeling the MDP. In order for an individual's activity to be recorded, the user had to have their phone on their person and turned on. In many cases, these requirements led to a relatively sparse record of activity and notification events over time (because a notification was only delivered if the individual's activity was being monitored and they were found to be stationary for at least 60 minutes). In order to obtain the data most relevant to the delivery of notifications and their effect on activity, only those records which had data for both were included. Days in which there was either no activity or no notification recorded were therefore ignored.

Additional preprocessing was preformed on the data to clean it for analysis, including dropping incomplete datapoints (such as those with missing column names, NA values, or malformed timestamps). Because of the unclear interpretation of comparisons between timestamps with different timezones, days in which the user changed timezones were also filtered from analysis. While the original dataset contained six activity types (cycling, walking, not available, running, stationary, and automotive), we divided these into two binary categories: active (cycling, walking, and running), and inactive (stationary and automotive). Activities marked as "not available" were categorized using last observation carried forward interpolation.

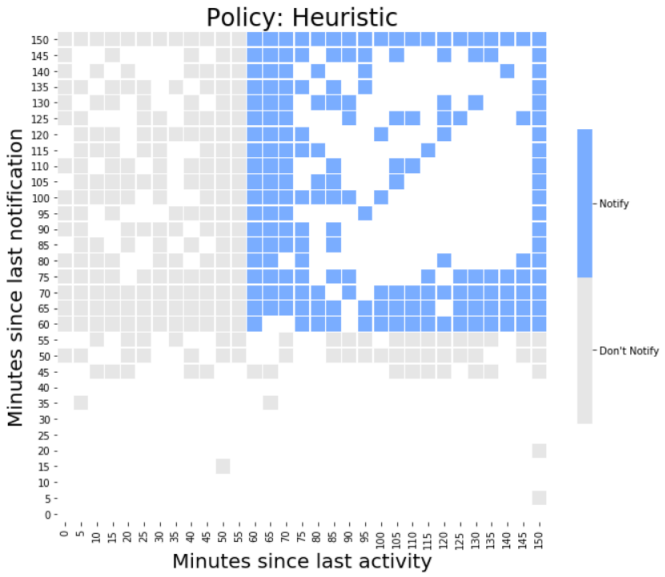

Figure 1: A heuristic policy based on notifying users who have not been active or notified for the past 60 minutes. White space denotes states for which at most one action was observed in the data and thus a policy cannot be meaningfully evaluated. Compare with Figure 6.

Model

In order to model the problem as an MDP, the data describing user activity over time was discretized into 15 minute intervals. At the end of each interval, the state was defined to be a tuple of the number of minutes since the last notification and the number of minutes since the last user activity (such as standing, running, etc.). In order to avoid a proliferation of uninformative states with large time values, these values were capped at 150 minutes.

In order to further constrain the state space, after the reward was calculated for a given interval, the time since last notification and activity were rounded to the nearest 5 minutes. Rounding reduced the number of states from 22,500 to 900, a more appropriate size for the amount of data available. We experimented with both larger and smaller rounding intervals and found that 5 minutes provided sufficient resolution without proliferation of states.

The action was a binary value indicating whether or not a notification was delivered in the interval of interest.

The reward was defined to be a function of user activity in the next time interval following an action. It was based on three components: (1) the proportion of that interval spent in active states (such as running instead of sitting), (2) a cost if the action was delivering a notification, and (3) an "effective notification" bonus if a notification delivery was closely

followed by user activity. Specifically:

$$R(s,a) = \sum_x (\beta_x \cdot p_x) + \mathbb{1}\{a = notify\}(\gamma + \alpha e^{-\tau}) \quad (1)$$

where p_x is the proportion of the 15-minute interval described by s that was spent in activity type x (e.g. sitting, standing, etc.), β_x is the reward associated with that activity, $\mathbb{1}\{a = notify\}$ is 1 if a notification was delivered in s and 0 otherwise. γ is the cost of delivering a notification. τ is the time between the delivered notification and the next recorded non-stationary activity, and α is a hyperparameter controlling how much these "effective notifications" are rewarded.

A follow-up analysis was modeled exactly as the first, but in addition to the time since last notification and time since last activity, we incorporated activity profiles (as calculated in McConnell et al. 2017) into the state. Specifically, users fell into one of five categories, including "Active", "Inactive", "Weekend Resters", "Weekend Warriors", and "Drivers".

Learning

Based on the criteria above, user data was processed into 187,442 sequences of (state, action, reward, new state). We then performed maximum likelihood estimation to derive the transition probabilities between states from the data, then solved the resulting MDP using value iteration.

We calculated the MLE estimates of the MDP transition probabilities T and rewards R as follows:

$$N(s,a) = \sum_{s'} N(s,a,s') \quad (2)$$

$$T(s'|s,a) = N(s,a,s')/N(s,a) \quad (3)$$

$$R(s,a) = \rho(s,a)/N(s,a) \quad (4)$$

where $N(s,a,s')$ represents the number of times we saw a particular (state s, action a, new state s') triple in the data and $\rho(s,a)$ represents the sum of the rewards overall all state-action pairs (s,a) seen in the data.

Using these MDP parameter estimates, we used an offline/sparse version of value constrained to the available data:

Algorithm 1 Sparse Value Iteration

function SparseValueIteration
$k \leftarrow 0$
$U_0(s) \leftarrow 0$ for all states s
repeat
  $U_{k+1}(s) \leftarrow \max_{\{a:N(s,a)>0\}}[R(s,a)+$
    $\gamma \sum_{\{s':N(s,a,s')>0\}} + T(s'|s,a)U_k(s')]$
  $\forall s$ s.t. $\sum_a N(s,a) > 0$
until $convergence$
return U_k

The difference here from traditional value iteration is that instead iterating over all possible states s, maximizing over all possible actions a, and summing over all possible states s', we restrict ourselves to only those states and actions for which we have associated observations in the dataset. This is obviously limiting, but it is a reasonable approximation under the finite data constraint.

While directly learning an approximation of the MDP model parameters using maximum likelihood estimation and running value iteration on that MDP yielded a policy that substantially outperformed random and heuristic baselines, the resulting policy (Figure 3) was sparse due to the small amount of data in some parts of the state space. In an effort to avoid this issue, and in an attempt to instead find a reasonable approximation of the reward for notifying or not notifying the user in all the states in the state space, we also attempted to implement a global approximation Q-learning algorithm that would learn $Q(s,a)$ for all states s and actions a in the state and action space directly. Our implementation is shown in Algorithm 2:

Algorithm 2 Global Approximation Q-learning

function GlobalApproximationQLearning
$t \leftarrow 0$
$s_0 \leftarrow$ initial state randomly chosen from observations
Initialize θ
repeat
  Choose a_t randomly from A s.t. $N(s_t, a_t) > 0$
  Observe new state s_{t+1} and reward r_t based on T and R
  $\theta \leftarrow \theta +$
    $\alpha(r_t + \gamma \max_{\{a:N(s_t,a)>0\}} \theta^T \beta(s_{t+1}, a) -$
    $\theta^T \beta(s_t, a_t)), \beta(s_t, a_t)$
until $t > t_{max}$
return θ

We ran the algorithm with $t_{max} = 1,000,000$ iterations, learning rate $\alpha = 0.01$ and discount factor $\gamma = 0.005$ (a larger discount factor tended to lead to overflow of the parameters in θ). A set of features based on the time since last notification and time since last activity (e.g. $\mathbb{1}\{t_{lastactivity} > 60\}$ and $\mathbb{1}t_{last_activity} + t_{last_notification} > 60$) were used as features in the approximation model.

Evaluation

In order to establish the benefit of the optimal policy, we implemented three simple policies as comparison baselines. The random policy took a random action at each state. The never notify policy always took the action of sending no notification. (Due to the constraint that policy evaluation could only be calculated for state-action pairs for which there were data, this policy did include a small number of notification actions at states in which only notifications were observed in the data. See the white boxes in Figure 5) The heuristic policy used the intuitive approach of always sending a notification when at least 60 minutes had elapsed since both the user's last activity and last notification (see Figure 1. Again, due to constraints on data available for evaluation, the policy was set to not notify on states for which no notification was ever observed in the data). This last approach was similar to

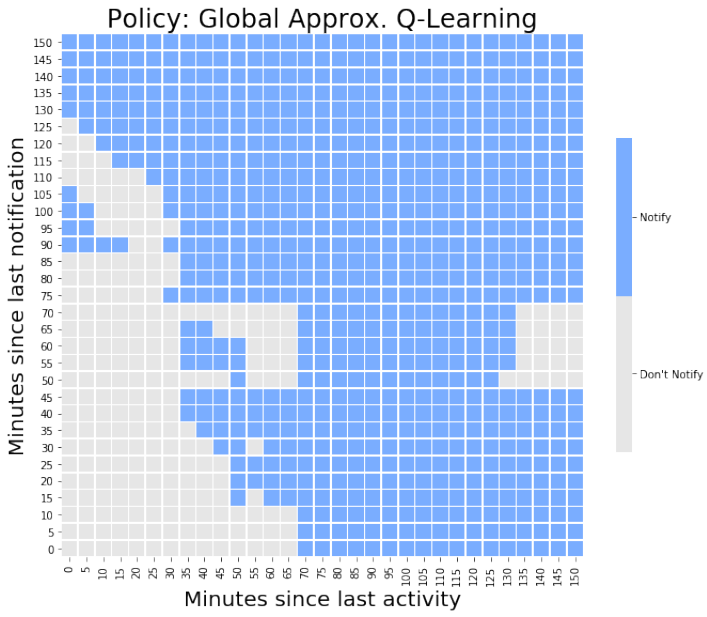

Figure 2: Policy learned via Q-Learning with Global Approximation.

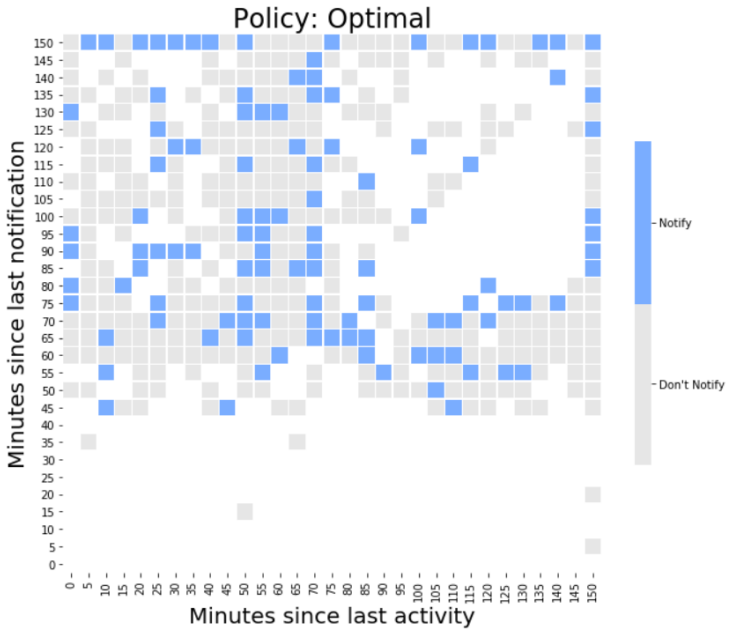

Figure 3: Notification policy learned via value iteration. Note that we could fill in these holes using a local approximation technique like nearest neighbors, but in this case we keep them to demonstrate the sparse nature of the data and by way of comparison with Figure 4.

the notification delivery logic used by the *MyHeart Counts* app.

We calculated the expected utility of each policy using iterative policy evaluation (constraining ourselves to the data available), that is, by estimating the sum of discounted future rewards from executing the policy with a discount factor of 0.99. The algorithm is shown in Algorithm 3.

Algorithm 3 Sparse Iterative Policy Evaluation

function SparseIterativePolicyEvaluation
$U_0^\pi(s) \leftarrow 0$ for all s s.t. $\sum_a N(s,a) > 0$
for $t \leftarrow 1$ **to** n **do**
    begin
    $U_t(s) \leftarrow R(s, \pi(s)) +$
    $\gamma \sum_{\{s' : N(s,a,s') > 0\}} T(s'|s, \pi(s)) U_{t-1}^\pi(s')$
    for all s s.t. $\sum_a N(s,a) > 0$
    end
return U_n

Results and Discussion

We found that the optimal policy learned via value iteration outperformed all other baselines, achieving approximately 3.6 times the expected utility of either the never notify policy or the heuristic policy, which performed comparably to each other. The optimal policy also outperformed the random policy by a factor of 10. Thus, the learned policy shows improvement over both the policy of taking no action and an arbitrary heuristic.

When examining the optimal policy in Figure 3, we observe some interesting trends. First, it appears that the optimal policy tends to overlap with both our heuristic policy (Figure 1) and the policy used to generate the data (Figure 5) by tending to notify more frequently in the bands where (1) the time since the last activity is between 50 and 70 minutes and the time since last notification is greater than 60 minutes, and (2) the time since the last notification is between 50 and 70 minutes and the time since last activity is greater than 60 minutes. This aligns with an intuitive sense of when the optimal time to deliver a notification might be: when the user has been sedentary for some time and has not received a notification for a while. Note, however, that the bands for our optimal policy are wider than the arbitrary heuristic and hard threshold under which the notifications were delivered (where the user was directed to stand if he/she has been stationary for more than 60 minutes and the last notification was delivered more than 60 minutes ago). This also aligns with our intuition; there is nothing unique about 60 minutes as a threshold, so to the extent that it is generally better to send a notification to an individual after time has passed since the last notification and activity, our policy captures that.

Additionally, the optimal policy suggested to notify the user more often than we might have expected in the states for which the user had been active within the last 30 minutes. This may be more a product of the way in which we constructed our reward function - where notifications

were rewarded for being "effective" insofar as physical activity shortly followed the notification delivery - rather than a reflection of optimal behavior in a high-level sense. If we imagine that a user is more likely to be active in the near future given that he was recently active, then our optimal policy would frequently acquire "effective" notification bonuses when the fact that physical activity followed a notification was due to chance rather than causality. Future work will need to address this more fully.

Interestingly, the optimal policy derived under our global approximation Q-learning approach (see Figure 2) demonstrated a similar pattern in which, even if the time since the last notification was within the previous 10 minutes, our derived optimal policy would repeatedly notify the user so long as he/she has been sedentary for more than about 40 minutes. Once the user becomes active - even if for just a moment - the notifications would stop under this learned policy. At first glance, this seems like an undesirable behavior for a push notification schedule (incessant and repeated notifications could quickly become irritable rather than productive), but in another sense if the ultimate objective of the app is to increase user's physical activity and repeated notifications in a short amount of time induce that effect, our learned policy could be quite effective. There are questions about sustained user engagement that would need to be considered in order to actually implement a policy like this, but that will be addressed in a future work.

The same patterns mentioned above tended to hold even when we incorporated users' activity profiles into the state space, but there were subtle variations based on these profiles (see Figure 7 and Figure 8 for examples). As reported in McConnell et al. 2017, "Active" users tended to have a high proportion of time spent in a physically active state throughout both the workweek and workday while "Inactive" users had a low proportion across all days. "Weekend Warriors" tended to be sedentary on weekdays while active during weekends. "Weekend Resters" displayed the opposite pattern, with consistent weekday activity that tapered off during weekends. "Drivers" showed comparable weekend and weekday activity but were less active overall than the "Active" group.

Interestingly, we found that the optimal notification policy for "Inactive" physical activity types tended to notify more frequently, on average, relative to other activity types across all states. Optimal policies for other activity profiles, especially "Weekend Warriors" and "Drivers" tended to more sparsely notify the user. In the case of "Drivers" this makes sense - those who are frequently driving cannot spontaneously become physically active while behind the wheel. The reason why "Weekend Warriors" are less frequently notified under the optimal policy is less clear. Perhaps because these individuals are physically active on the weekends, they feel less of a need to be physically active during the week and justify not responding to physical activity coaching accordingly. The data for these other activity types were more limited, however, making such interpretations somewhat tenuous. Answering these questions more definitively will require more work in the future.

One troubling finding from our Value Iteration formulation is that many of the states had only one recorded action available in the dataset from which we could derive an optimal policy. Those states for which at most one action (either notify or not notify) was recorded are marked in white in Figure 3. By comparison, Figure 6 only shades in white those states for which we had no observed activity at all (i.e. the states were not observed at all in our dataset). The states in Figure 6 for which there are no record at all in the dataset are in a reasonable space: under the policy in which the notifications were originally delivered, every time a person had more than 60 minutes since the last activity and more than 60 minutes since the last notification, a notification would have been fired, moving the individual to 0 minutes since last notification. Thus there should be little to no data in the corner representing time since last notification > 60 minutes and time since last activity > 60 minutes. (The fact that there are any data at all in that space reflect subtle differences in the way in which we derived times since last activity and notification compared to the way in which the original app derived these times). While Q-learning with global approximation addressed this issue to some extent by approximating $Q(s, a)$ for more abstract features of the state space, the limited data and subsequent variability of policies learned under the Q-learning method made finding a globally optimal policy difficult.

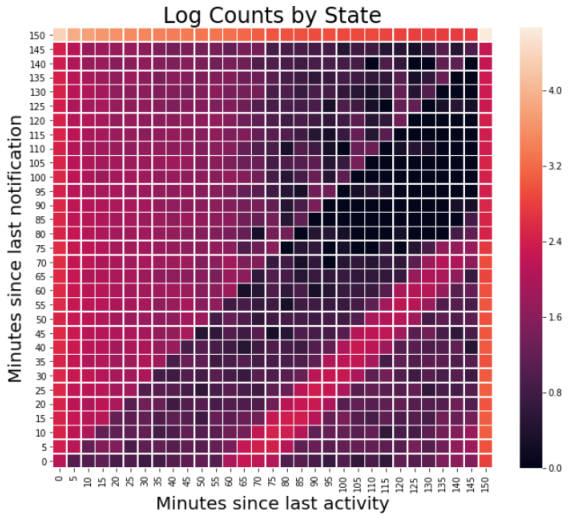

Figure 4: Log counts of the number of instances of each state in the training data. A large proportion of data appears at the outer edge of the state space.

Limitations

Several limitations to the MDP-based approach arose from the data collection and preprocessing methods used. Despite the large number of (state, action, reward, new state) sequences derived from the data, states were not evenly distributed across the state space. Rather, as can be seen in Figure 4, a large number of states were concentrated on the "edges" of the space, with the minutes since last notification or minutes since last activity reaching the maximum. This

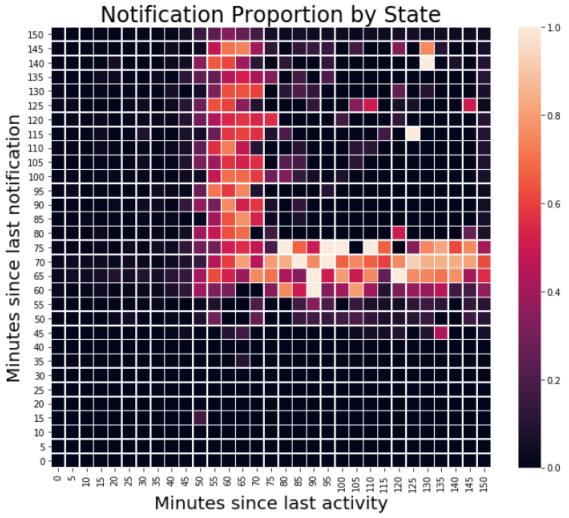

Figure 5: The proportion of notification actions by state in the training data. A small band of states show a high proportion of notifications in the data while most other states have few to no examples of a notification delivered from that state.

distribution resulted in a lack of examples for certain areas of the state space, with some states having 10 or fewer instances in the data.

Furthermore, learning the effectiveness of notification timing from data was impeded by the fact that notifications were not delivered according to a random schedule in the training data. In the original app, notifications were delivered only after the user had been inactive for the past hour and after at least an hour had passed since the last notification. This led to a high proportion of training examples including notifications in a narrow band of the state space and a scarcity of notifications elsewhere (see Figure 5). Large areas of the state space only contained examples of a single action taken from that state, rather than both notify and non-notify actions, as can be observed in the white space in Figures 1 and 3.

Finally, it must be acknowledged that strict data filtering protocols may have neglected certain classes of users. For example, dropping training examples that required cross-time zone comparison may have systematically eliminated "Drivers" from the training data.

Future work

While relatively parsimonious, a state space based solely on the amount of time since the user last moved and was last notified is a weak representation of the user's true state. Intuitively, the state should capture all the information required to decide whether or not the user is likely to respond to an activity notification in that moment. A richer state space including such variables as the time of day, the day of the week, and the user's own historical movement profile (whether they tend to be active in the morning or evening, on weekdays or weekends) would produce a more accurate notification model.

More fundamentally, the user's true state of "responsiveness" to a notification cannot be directly observed, suggesting that the problem may lend itself to the application of Partially Observable Markov Decision Processes (POMDPs). In this formulation, the belief state of how likely a user is to be responsive would be informed by observations of his or her recent activity in addition to a model of how past notifications affect future responsiveness.

Both a richer MDP state and a POMDP model would benefit from additional data, particularly when derived using a schedule that distributes notifications across the state space.

Conclusion

Our investigation has revealed the potential of Markov-based models to improve the timing of activity reminder notifications. Even relatively simple state representations have been shown to improve markedly over the naive, rote delivery schedules often used in such interventions. While our conclusions are limited by the constraints of the *MyHeart Counts* data, they suggest that further work, including a richer state space, POMDP models, and more complete data may produce even more effective, personalized notification schedules, promoting the success of mobile-based health interventions.

Contributions

Both team members collaborated to produce the writeups. Individual contributions are listed below.
Scott:

- extract intervention timestamps for each user from data (Preprocessing)
- reward calculation in state extraction function (Model)
- value iteration on extracted data (Learning)
- policy evaluation as described (Evaluation)
- expanded state space incorporating user activity group (Model)
- Q-Learning with Global Approximation (Learning)
- Scott works in the Ashley lab. His preprocessing work, as well as consultations with the *MyHeart Counts* team, were performed as part of his research rotation.

Gordon:

- extract user motion data and clean, preprocess, and unify motion and notification data (Preprocessing)
- extract (state, action, reward, new state) sequences from user data (with the exception of the reward calculation, implemented by Scotty) (Model)
- baseline comparison policies (Evaluation)
- policy and state space visualizations (Figures)
- analysis of data availability in state space (Limitations)

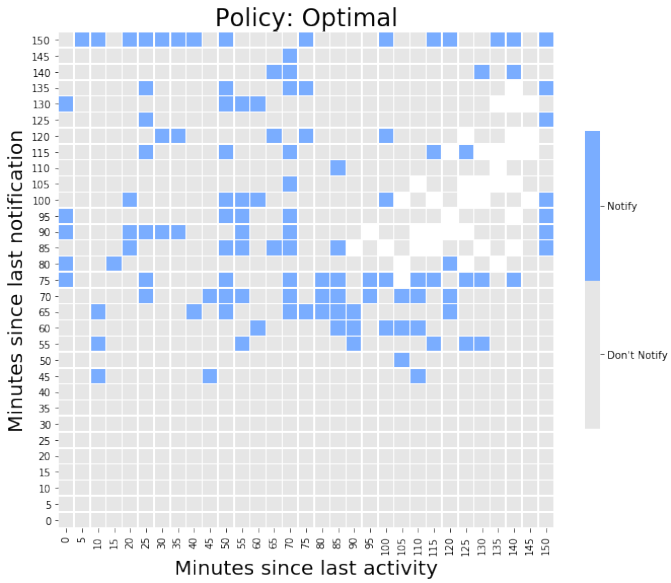

Figure 6: Notification policy learned via value iteration. White spaces in this case reflect parts of the state in which no actions recorded in the data. Compare with Figure 3.

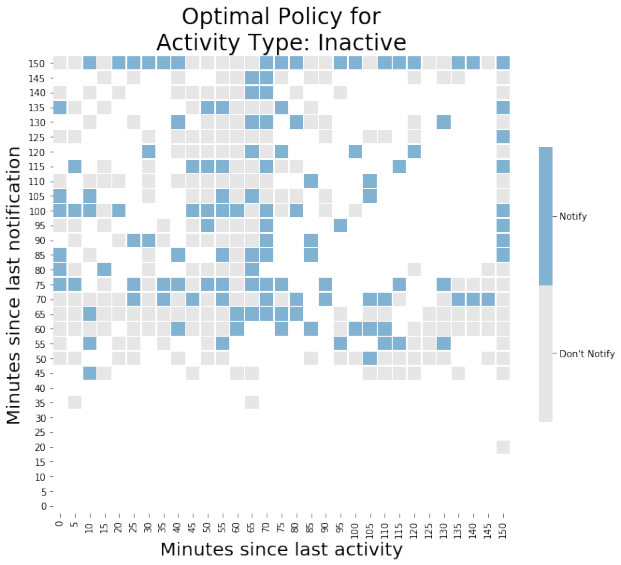

Figure 7: Optimal policy derived for users with inactive activity profiles. The notification frequency for these users under the optimal policy reflects the optimal policy when activity profile is not considered as part of the state. This is perhaps not surprising as the inactive activity profile was the predominant activity type in the dataset.

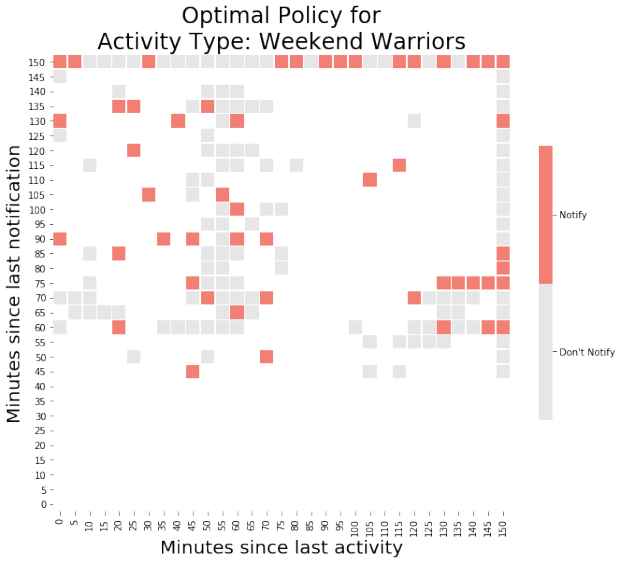

Figure 8: Optimal policy derived for users with weekend warrior activity profiles. The notification frequency for these users under the optimal policy was slightly more sparse than for inactive types, though limited data make direct comparisons difficult.

Works Cited

Ma, Shaohui and Hao Zhang. "A Dynamic CRM Model Based on POMDP". *Fifth International Conference on Fuzzy Systems and Knowledge Discovery*. 2008. Print.

McConnell, Michael V, et al. "Feasibility of obtaining measures of lifestyle from a smartphone app: the MyHeart Counts Cardiovascular Health Study". *JAMA cardiology* 2.1 (2017): 67–76. Print.

Morrison, Leanne G., et al. "The Effect of Timing and Frequency of Push Notifications on Usage of a Smartphone-Based Stress Management Intervention: An Exploratory Trial". *PLoS ONE* 12.1. DOI: https://doi.org/10.1371/journal.pone.0169162 (2017). Print.

Pielot, Martin, et al. "Beyond Interruptibility: Predicting Opportune Moments to Engage Mobile Phone Users". *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*. 2017. Print.

Shcherbina, A, et al. "Digital randomized controlled trial of physical activity intervention". *Under Review* (2018). Print.

Tan, Luchen, et al. "An Exploration of Evaluation Metrics for Mobile Push Notifications". *Proceedings of, the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 2016. Print.

Wigginton, Craig, et al. "Global mobile consumer survey: US edition". 2018. (accessed: 10.04.2018). Web. <https://www2.deloitte.com/us/en/pages/technology-media-and-telecommunications/articles/global-mobile-consumer-survey-us-edition.html#>.